On Central Primitives for Quantum Cryptography with Classical Communication

Kai-Min Chung[1], Eli Goldin[2], and Matthew Gray[3]

[1]Academia Sinica (kmchung@iis.sinica.edu.tw)
[2]New York University (eli.goldin@nyu.edu)
[3]University of Oxford (matthew.gray@cs.ox.ac.uk)

February 27, 2024

Abstract

Recent work has introduced the "Quantum-Computation Classical-Communication" (QCCC) (Chung et. al.) setting for cryptography. There has been some evidence that One Way Puzzles (OWPuzz) are the natural central cryptographic primitive for this setting (Khurana and Tomer). For a primitive to be considered central it should have several characteristics. It should be well behaved (which for this paper we will think of as having amplification, combiners, and universal constructions); it should be implied by a wide variety of other primitives; and it should be equivalent to some class of useful primitives. We present combiners, correctness and security amplification, and a universal construction for OWPuzz. Our proof of security amplification uses a new and cleaner version construction of EFI from OWPuzz (in comparison to the result of Khurana and Tomer) that generalizes to weak OWPuzz and is the most technically involved section of the paper. It was previously known that OWPuzz are implied by other primitives of interest including commitments, symmetric key encryption, one way state generators (OWSG), and therefore pseudorandom states (PRS). However we are able to rule out OWPuzz's equivalence to many of these primitives by showing a black box separation between general OWPuzz and a restricted class of OWPuzz (those with efficient verification, which we call $\mathsf{EV}-\mathsf{OWPuzz}$). We then show that $\mathsf{EV}-\mathsf{OWPuzz}$ are also implied by most of these primitives, which separates them from OWPuzz as well. This separation also separates extending PRS from highly compressing PRS answering an open question of Ananth et. al.

Contents

1 Introduction

In the realm of cryptography, there is perhaps no primitive more important than the one-way function. A one-way function is an efficiently computable deterministic function which is easy to compute, but hard to invert. Although at first glance the definition seems simple, one-way functions are special for several reasons. First and foremost, one-way functions are "minimal." If modern cryptography exists in any form, then one-way functions must also exist [HILL99, IL89, Imp95]. Furthermore, pretty much all of these constructions are obvious. Second, one-way functions are "useful." There is a large class of cryptographic primitives (known as Minicrypt) which can all be built from and are equivalent to one-way functions [Imp95]. Included in Minicrypt are symmetric key encryption, pseudorandom generators, and commitment schemes [HILL99, GGM86, Nao91]. Finally, one-way functions are "well-behaved." There satisfy several natural properties [Lev87], and are equivalent to most of their variants [Yao82, IL89]. Due to these three characteristics of one-way functions, one of the most useful things to do when trying to understand a new classical cryptographic primitive is to compare it to a one-way function.

This centrality of one-way functions no longer holds once quantum computation enters the picture. In particular, in the quantum setting, it seems that one-way functions are no longer minimal [Kre21]. In particular, there exists a quantum oracle relative to which one-way functions do not exist, but quantum cryptography (in the form of pseudorandom state generators, quantum bit commitments, and many other primitives) is still possible. Recently, there has been strong evidence in support of a new simple primitive, the EFI pair, being minimal [KT24, BCQ22]. An EFI pair is a pair of efficiently samplable quantum mixed states which are indistinguishable yet statistically far. Furthermore, EFI pairs are also useful. They can be used to build a large number of quantum

cryptographic primitives, from quantum bit commitments to secure multiparty computation [BCQ22, AQY22]. Finally, EFI pairs are fairly well-behaved. The security of EFI pairs can be amplified [BQSY23], there exists combiners and universal constructions for EFI pairs [HKNY23], and EFI pairs are also equivalent to some of their variants [HMY23].

In the classical setting, it appears that one-way functions serve as an effective minimal primitive. In the quantum output setting, EFI pairs are a promising candidate for our minimal primitive. A number of recent works have also considered a hybrid setting, primitives where the cryptographic algorithms are quantum, but all communication and outputs are classical [ACC+22, ALY23a] [CLM23, KT24]. In the style of [ACC+22], we will refer to this as the quantum computation classical communication (QCCC) setting. An immediate and natural question about this setting is "what is a good central primitive?"

Just like in the fully quantum setting, it is unlikely that one-way functions can be a minimal primitive in the QCCC setting. In particular, there is a barrier to building one-way functions from a QCCC primitive known as the one-way puzzle [KT24, Kre21]. A one-way puzzle consists of an efficient quantum sampler which produces keys and puzzles along with a (possibly inefficient) verification procedure. The one-wayness corresponds to the idea that given a puzzle, it should be hard to find a matching key. Although one-way functions cannot serve as a central QCCC primitive, at first glance one-way puzzles make a fairly good candidate. In particular, one-way puzzles are minimal in the sense that almost all QCCC primitives can be used to build one-way puzzles [KT24].

On the other hand, their well-behavedness and usefulness are less clear. It is known that one-way puzzles can be used to build EFI pairs (and thus everything which follows from EFI pairs) [KT24]. However, as far as the authors are aware, there are no existing constructions of QCCC style primitives from one-way puzzles. The well-behavedness of one-way puzzles is similarly unstudied.

Our results In this work, we seek to investigate what primitives can be built from one-way puzzles, as well what useful properties one-way puzzles may or may not satisfy. Whether or not one-way puzzles are adopted as a central primitive in the same manner as one-way functions or EFI pairs is a community matter, but we hope that our results help shed light onto the question. To summarize our results, we show that

1. There exists a robust combiner for one-way puzzles. That is, given two candidate one-way puzzles, there is a way to combine the candidates to get a construction which is secure as long as one of the candidates is secure.

2. There exists a universal construction of a one-way puzzle. That is, a construction which is secure as long as one-way puzzles exist.

3. There exist amplification theorems for one-way puzzles. That is, there is a method to take a one-way puzzle with weakened correctness or security guarantees and transform it into a full one-way puzzle.

4. We show that one-way puzzles can be built from EFID pairs (the QCCC version of EFI pairs).

5. We show that one-way puzzles are equivalent to one-way puzzles whose key is generated uniformly at random, answering an open question of [KT24].

We also consider in detail an important variant of one-way puzzles, a restricted version of which was first introduced under the name "hard quantum planted problem for QCMA," [KNY23], but which we will refer to as efficiently verifiable one-way puzzles. We show the following results about this variant

1. There exists combiners, a universal construction, and amplification theorems for efficiently verifiable one-way puzzles.

2. Most QCCC primitives which can be used to build one-way puzzles can also be used to build efficiently verifiable one-way puzzles, with the notable exception of interactive commitment schemes. In particular, we show explicitly that pseudodeterministic PRGs and non-interactive commitments imply efficiently verifiable one-way puzzles.

3. There exists a quantum oracle relative to which one-way puzzles exist but efficiently verifiable one-way puzzles do not.

The last two points here together provide a barrier to building most QCCC primitives from one-way puzzles. Perhaps this means that efficiently verifiable one-way puzzles make a better candidate for centrality. However, if QCCC commitments can be built from one-way puzzles, then it may make sense to treat one-way puzzles as a central primitive on a lower level than efficiently verifiable one-way puzzles. We compare the relationship to the separation between one-way functions and one-way permutations.

Note that our separation in fact separates efficiently verifiable one-way puzzles from pseudorandom state generators with linear output. But it is not hard to show that pseudorandom state generators with logarithmic output can be used to build efficiently verifiable one-way puzzles. This, our final separation also provides a barrier to length reeduction for pseudorandom state generators, answering an open question of [ALY23b].

A summary of known relationships between QCCC primitives is included in Figure 1.

A better construction of EFI pairs from one-way puzzles The most technically demanding of our results is, surprisingly, the amplification theorem for one-way puzzles. It turns out that due to the inefficient nature of verification, most natural techniques fail. The techniques we use to achieve amplification for one-way puzzles can be also be used to construct EFI pairs from one-way puzzles, recreating a result from [KT24]. In addition, our construction has several advantages over the existing construction in the literature.

First, the proof of security for our construction is significantly more straightforward than the existing argument. In particular, the argument does not rely on techniques dealing explicitly with a preimage space (such as leftover hash lemma or Goldreich-Levin), and so more naturally fits with the quantum nature of the primitive. Second, our construction produces an EFI pair even when instantiated with a one-way puzzle with weakened security guarantees. This is the essential reason that this technique is useful for proving amplification.

2 Technical Overview

2.1 A cleaner construction of EFI pairs from any one-way puzzle

In a recent work by Khurana and Tomer [KT24], it was shown that there is a black-box construction of an EFI pair from any one-way puzzle. Since one-way puzzles can be built from one-way state generators, this then shows that if one-way state generators exist, so do EFI pairs (and thus quantum bit commitments).

Since EFI pairs intuitively are a "pseudorandom" primitive while one-way puzzles are a "one-way" primitive, the argument presented in [KT24] is heavily inspired by the classical construction of a pseudorandom generator from any one-way function, first shown in [HILL99].

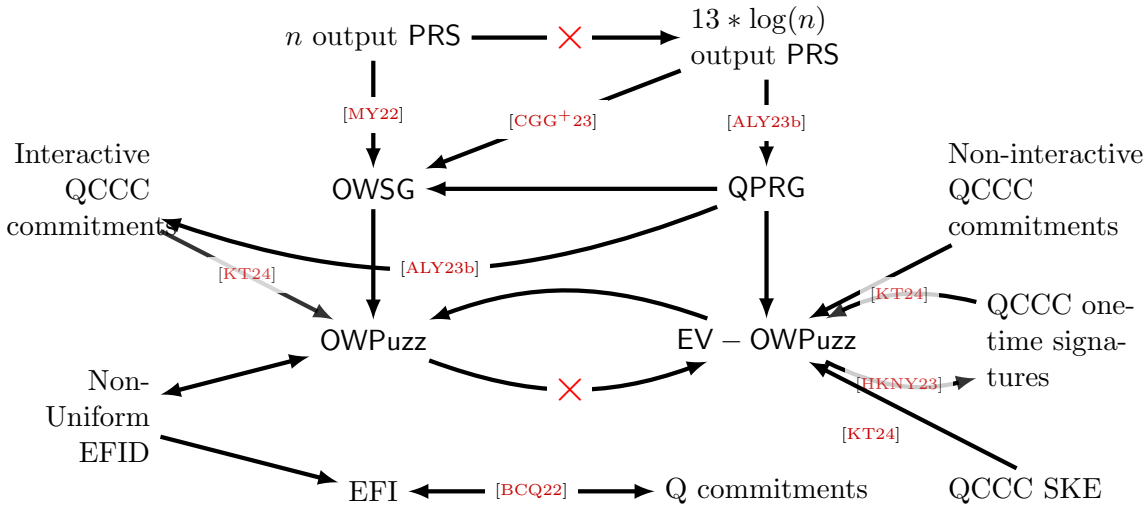

Figure 1: All implications known about one-way puzzles (OWPuzz) and efficiently verifiable one-way puzzles (EV − OWPuzz).

The key idea behind [HILL99] is to first use the one-way function to construct something called a pseudoentropy generator. A pseudoentropy generator is simply a samplable distribution which is indistinguishable from another (not necessarily samplable) distribution with greater entropy. Then, the pseudoentropy generator is used to construct a non-uniform PRG. That is, a PRG where the construction takes in a short advice string of length $O(\log \lambda)$ depending on the security parameter. This gives out a different PRG candidate for each possible value of the advice string. Applying a PRG combiner to all of these candidates then gives a standard PRG.

[KT24] follows the same overall structure to build an EFI pair from any one-way puzzle. In particular, they show how to build a pseudoentropy generator from any one-way puzzle, and then show how to use a one-way puzzle to build something they refer to as an imbalanced EFID pair. An EFID pair is classical version of an EFI pair. We recall that a non-uniform EFID pair is an EFID pair that takes in a short advice string. An imbalanced EFID pair is a stronger primitive than a non-uniform EFID pair, where there are additional requirements on hiding and/or binding when the primitive is instantiated with incorrect advice. Finally, they show how to use an imbalanced EFID pair to build a standard EFI pair, although this technique requires switching to quantum output.

In this work, we present an alternate construction of EFI pairs from one-way puzzles, with several advantages. The foremost advantage, which is useful for our other results, is that our construction works even when instantiated with weak one-way puzzles. In addition to this, the proof of our construction is significantly simpler, and relies almost entirely on standard classical techniques.

Theorem 1 (Informal version of Corollary 15)**.** *If there exists a weak one-way puzzle, then there exists an EFI pair.*

The overall approach. While [KT24] relies on the techniques of [HILL99] to realize their construction, there have been a number of follow-up works succeeding [HILL99] providing more efficient constructions of PRGs from OWFs [HRV10] [VZ12, MP23]. In particular, we observe that the techniques of [VZ12] are particularly "quantum-friendly," much more so than the techniques of [HILL99]. Furthermore, we make the (as far as we are

aware) novel observation that the construction of [VZ12] gives a pseudorandom generator even when instantiated with a weak one-way function.

The failure of Goldreich-Levin for weak one-way puzzles. One key idea underlying [HILL99], as well as most other constructions of PRGs from one-way functions [VZ12, MP23], is to extract $H_{min}(x|f(x)) + O(\log n)$ bits of pseudoentropy from x given $f(x)$. The leftover hash lemma gives the ability to extract $H_{min}(x|f(x)) - O(\log n)$ bits of entropy from x, and Goldreich-Levin provides an extra $O(\log n)$ bits of pseudoentropy [GL89], so these two techniques together can extract a pseudorandom string of length $H_{min}(x|f(x)) + O(\log n)$ from x given $f(x)$.

In particular, the Goldreich-Levin theorem shows that if there is an algorithm distinguishing $\mathsf{Ext}(x)$ from uniform given $f(x)$ with advantage ϵ, then there is an algorithm computing x from $f(x)$ with probability $\mathsf{poly}(\epsilon)$ [GL89]. Since ϵ^2 is negligible for a strong one-way function, so is ϵ, and so these distributions are indistinguishable. However, if f is only a weak one-way function, then we only get a constant bound on the distinguishing advantage, and so the approaches of [HILL99, VZ12, MP23] all break down.

A similar approach, with some technically involved adjustments to handle quantum sampling, is done in [KT24] by using a quantum version of the Goldre-ich-Levin theorem [AC01]. In particular, [KT24] also relies on using Goldreich-Levin to extract $O(\log n)$ from the key k given the puzzle s. But for the same reason as before, this approach does not hold when the sampler is only weakly one-way.

Furthermore, there is a lot of technical care needed when using the leftover hash lemma and Goldreich-Levin on puzzles sampled using quantum randomness [KT24]. This is because the pre-image space of a puzzle is now a distribution over keys instead of a set, and so hashing techniques become significantly more complicated. Luckily, [VZ12] demonstrates a way to construct PRGs from one-way functions without relying on either of these techniques, providing an approach that is both quantum-friendly and applies even with weak security. We adapt their techniques to give a construction of EFI pairs from weak one-way puzzles illustrated in Figure 2.

The construction of [VZ12]. To build a PRG from a one-way function f, [VZ12] makes the observation that the distribution $(f(x), x)$ satisfies a property which they call KL-hard to sample. In particular, this means that for any sampler $\mathcal{S}$ (which in this case can be thought of as a distributional inverter),

$$KL(f(x), x||f(x), \mathcal{S}(f(x))) \geq \delta$$

for some value of $\delta \geq \frac{1}{\mathsf{poly}(\lambda)}$. Here KL refers to Kullback-Leibler divergence, or "relative entropy." They then adapt the techniques of [HRV10] to build a PRG from a distribution which is KL-hard to sample. Note that this construction requires knowledge of the entropy of the KL-hard to sample distribution. However, for a one-way function, $H(f(x), x) = |x|$ the input length of the one-way function.

For a ϵ one-way function, the KL-hardness parameter is $\delta = -\log \epsilon$. Thus, for a standard one-way function, $\delta = \omega(\log \lambda)$. But the techniques of [HRV10] apply whenever $\delta = \frac{1}{\mathsf{poly}(\lambda)}$, and so the techniques of [VZ12] work just as well for weak one-way functions. Thus, the same construction gives a PRG from any weak one-way functions.

Building a KL-hard to sample distribution from a one-way puzzle The key observation underlying [VZ12] is that KL divergence can only decrease from computation.

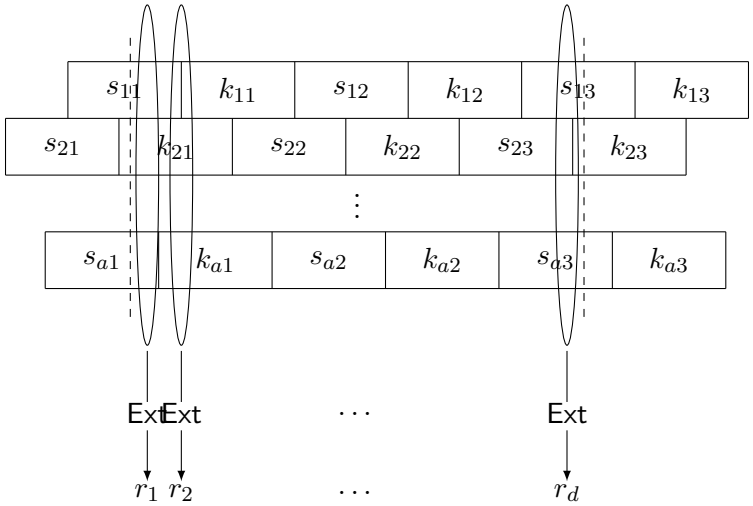

Figure 2: The construction of [VZ12, HRV10] applied to a one-way puzzle $\mathsf{Samp} \to (k, s)$. The idea is that many samples are taken and arranged in a grid. Then, each row is given a random offset, with both sides truncated. Finally, some number of random bits are extracted from each column using a pairwise-independent hash Ext. This produces a pseudorandom string with less than full entropy, and we can repeat to get a non-uniform EFID pair.

That is, for any function F,

$$KL(F(X)||F(Y)) \leq KL(X||Y)$$

But the boolean function $F(y, x) = 1$ if and only if $f(x) = y$ is well-defined. So if $\mathcal{S}$ is any sampler,

$$KL(f(x), x||f(x), \mathcal{S}(f(x))) \geq KL(F(f(x), x)||F(f(x), \mathcal{S}(f(x))))$$

$$= KL(1||Bern(p)) = -\log p$$

where p is the advantage of $\mathcal{S}$ in the one-way function security game. This immediately gives that the distribution $(f(x), x)$ is KL-hard to sample.

We observe that the same exact technique also works for one-way puzzles. In particular, let $(\mathsf{Samp}, \mathsf{Ver})$ be a one-way puzzle and let $\mathsf{Samp} \to (k, s)$. The equivalent of checking if $f(x) = y$ is simply to run verification. And so

$$KL(s, k||s, \mathcal{S}(k)) \geq KL(\mathsf{Ver}(s, k)||\mathsf{Ver}(s, \mathcal{S}(k))) = KL(Bern(q)||Bern(p))$$

where p is the success probability of $\mathcal{S}$ in the one-way puzzle game and q is the correctness parameter of the one-way puzzle. Although we do not have $KL(Bern(q)||Bern(p)) = -\log p$, when Samp is a weak one-way puzzle, we can still lower bound $KL(s, k||s, \mathcal{S}(k))$ by $\frac{1}{\mathsf{poly}(\lambda)}$. And so (s, k) is KL-hard to sample.

Building a non-uniform EFID pair from a KL-hard to sample distribution Note that the techniques [VZ12] uses to build a PRG from a KL-hard to sample distribution are entirely black box, and so also work in the quantum setting. Thus, applying the same construction to (s, k) produces a pseudorandom distribution D with length $d = |D|$ depending on $H(k, s)$. When building a PRG, the approach [HRV10, VZ12] take is to

argue that D can be sampled by applying some function G to a uniformly random string of length $d' < d$, and so G is a PRG. Here, the randomness of the distribution is quantum, and so this idea will not apply directly. But similar reasoning can be used to show a upper bound on the entropy of D. In particular, we produce such an argument directly and show that $H(D) < d - \mathsf{poly}(\lambda)$. For a visualization of the construction of D, see Figure 2.

We then observe that any distribution with sufficiently less entropy than its length must have some statistical distance from the uniform distribution. Thus, the [VZ12] construction applied to a one-way puzzle produces a distribution D which is indistinguishable from uniform but has noticeable statistical distance from uniform. We then use parallel repetition to boost the statistical distance to $1 - \mathsf{negl}(\lambda)$, and so the pair $(\mathcal{U}^t, D^t)$ forms a EFID pair.

Unfortunately, this construction has a number of pseudorandom bits dependent on $H(k, s)$. Thus, the EFID pair construction has to have knowledge of the entropy of the one-way puzzle sampler output. This can be done by giving the construction $\Theta(\log \lambda)$ bits of advice, and so instead of a full EFID pair, we get a non-uniform EFID pair.

From non-uniform EFID pairs to EFI pairs To recap, [KT24] built imbalanced EFID (a stronger version of non-uniform EFID) from one-way puzzles, while our technique only builds non-uniform EFID from one-way puzzles. Note that this is not a fundamental difference, upon observation it is clear that our construction also satisfies the requirements of imbalanced EFID.

However, the reason [KT24] required this stronger notion of non-uniform EFID was because, at the time that work was published, it was unknown how to build combiners for EFI pairs. Recent work (interestingly using similar techniques to [KT24]) has shown how to combine EFI pairs [HKNY23], and so using these techniques EFI pairs follow directly from non-uniform EFID.

2.2 Combiners and universal constructions

One major property satisfied by one-way functions is the existence of a universal construction [Lev87]. By this, we mean that there exists a specific construction of a one-way function which is secure if any one-way functions exist.

As shown originally by Levin [Lev87] and formalized in [HKN+05], this useful fact is essentially a corollary of the fact that there exists *robust combiners* for one-way functions. That is, given any two one-way function candidates f and g, there is a construction $h^{f,g}$ such that h is one-way as long as one of f or g is one-way.

The universal one-way function is then defined as follows. Take the first $\log \lambda$ Turing machines and treat them as one-way function candidates. Running the combiner on all these candidates results in a universal one-way function f_U. As long as one-way functions exist, there is a Turing machine with some constant length which acts as a good one-way function. Thus, for all sufficiently large λ, f_U will also be a one-way function.

Since both combiners and universal constructions are highly desirable properties, we would like to investigate whether robust combiners also exist for one-way puzzles. We thus prove the following theorem

Theorem 2 (Informal version of Corollary 11)**.** *There exists a robust combiner for one-way puzzles.*

with the following corollary

Corollary 1 (Informal version of Theorem 26)**.** *There exists a pair of algorithms* $(\mathsf{Samp}_U, \mathsf{Ver}_U)$ *such that as long as one-way puzzles exist,* $(\mathsf{Samp}_U, \mathsf{Ver}_U)$ *is a one-way puzzle.*

Note that it has been shown that combiners and universal constructions exist for quantum primitives which both imply one-way puzzles and are implied by one-way puzzles, namely one-way state generators and EFI pairs respectively [HKNY23]. Thus, this result should not be particularly surprising. However, none of the arguments for constructing combiners for one-way state generators, EFI pairs, or one-way functions translate directly into building combiners for one-way puzzles.

Note that if we know that both candidate one-way puzzles satisfy correctness, then it is easy to construct a combiner. In particular, running both candidate samplers in parallel and having the verification algorithm accept if and only if both candidate verification algorithms accept is enough to ensure that the combined construction satisfy both correctness and security.

However, if we omit the correctness requirement, then it is possible that the "bad" verification algorithm always rejects. In this case, the combiner we defined previously will also not satisfy correctness.

To resolve this issue, we follow the template of [HKNY23] and show that there is a "correctness guaranteeing" procedure for any one-way puzzle. Namely

Theorem 3 (Informal version of Corollary 9)**.** *Let* $(\mathsf{Samp}, \mathsf{Ver})$ *be a one-way puzzle candidate. There exists a construction* $(\mathsf{Samp}', \mathsf{Ver}')$ *where* $\mathsf{Samp}', \mathsf{Ver}'$ *depend on* $(\mathsf{Samp}, \mathsf{Ver})$ *satisfying the following*

1. *If* $(\mathsf{Samp}, \mathsf{Ver})$ *is a one-way puzzle, then so is* $(\mathsf{Samp}', \mathsf{Ver}')$.
2. *Regardless of whether* $(\mathsf{Samp}, \mathsf{Ver})$ *is a one-way puzzle,* $(\mathsf{Samp}', \mathsf{Ver}')$ *satisfies one-way puzzle correctness.*

If we apply this correctness amplification procedure to the candidate one-way puzzles and then apply the security combiner described earlier, we achieve a robust combiner for one-way puzzles.

The main question remaining is how to actually do this correctness amplification. The natural approach to correctness guaranteeing (which is analogous to the approach used by [HKNY23]) is to have the sampler check whether verification passes on its produced key-puzzle pair. If not, the sampler will output a special symbol $\perp$, on which the verifier will always accept. However, this approach requires that the sampler be able to run the verifier. But for one-way puzzles, the verification algorithm may not be efficient.

Our solution is to defer the checking step to the verification algorithm itself. In particular, we will say that a puzzle is good if the probability that verification passes when it is naturally generated is high. Since our verification algorithm is inefficient, it has the computational resources to check if a given puzzle is good. The key idea, then is that we modify verification to automatically accept any good puzzle.

If the scheme originally satisfied correctness, then all but a negligible fraction of puzzles will be good and so we do not compromise security. Furthermore, since the probability that verification fails on a good puzzle is by definition low, the probability that the modified verification fails will also be low. Note that this style of correctness guaranteeing will only give a guarantee that the correctness error is below some constant (say $1/2$). We can then boost to full correctness through parallel repetition.

A note on the definition of one-way puzzles The original definition of one-way puzzles introduced required that the verification procedure be represented by a Turing machine which is guaranteed to halt (i.e. a decider) [KT24]. When defining a robust combiner for use in a universal construction, it is necessary that the combiner work even if

one of the candidate verification algorithms does not halt. This makes building a combiner seemingly as difficult as solving the halting problem.

We instead define one-way puzzles so that verification can be any arbitrary function. Note that, because Ver is never actually run, all known constructions using one-way puzzles go through when using this weakened definition. In addition, under our definition, combiners and universal constructions exist. Thus, we believe that this generalized definition is the "right" definition of a one-way puzzle, and that the restriction of verification to halting Turing machines used by [KT24] is unnecessarily restrictive.

2.3 Amplification of one-way puzzles

A second desirable property for a central primitive to have is an amplification theorem. In particular, given a one-way function with a weaker security guarantee, it is possible to build a normal one-way function. This makes it significantly easier to construct one-way functions from other primitives as well as produce candidate one-way functions.

Thus, one may wonder whether the same is true for one-way puzzles. That is, given a one-way puzzle with a weakened security guarantee, is it possible to build a normal one-way puzzle? We can also ask the same question of correctness. Given a one-way puzzle with a weakened correctness guarantee, is it possible to build a normal one-way puzzle?

In particular, we define (α, β) one-way puzzles, where α is the correctness error and β the security error. Observe that standard one-way puzzles are simply $(\mathsf{negl}(\lambda), \mathsf{negl}(\lambda))$ one-way puzzles. We show the following

Theorem 4 (Restatement of Theorem 20)**.** *If there exists a* $(1 - 1/\mathsf{poly}(\lambda), \mathsf{negl}(\lambda))$ *one-way puzzle, then there exists a* $(\mathsf{negl}(\lambda), \mathsf{negl}(\lambda))$ *one-way puzzle.*

Theorem 5 (Restatement of Theorem 22)**.** *If there exists a* $(\mathsf{negl}(\lambda), 1 - 1/\mathsf{poly}(\lambda))$ *one-way puzzle, then there exists a* $(\mathsf{negl}(\lambda), \mathsf{negl}(\lambda))$ *one-way puzzle.*

Amplifying Security For the purposes of this section, we will refer to a $(\mathsf{negl}(\lambda), 1 - 1/\mathsf{poly}(\lambda))$ one-way puzzle as a weak one-way puzzle. We will also refer to the standard notion of a one-way puzzle as a strong one-way puzzle. The question of security amplification can then be rephrased as "can we build a strong one-way puzzle from any weak one-way puzzle?"

Recently, [BQSY23] showed that parallel repetition amplifies soundness guarantees for any 3 round quantum interactive protocol. At first glace, one might think that this result immediately gives a security amplification theorem for one-way puzzles.

Upon observation, it turns out that the argument of [BQSY23] relies on the assumption that the security game itself can be run efficiently. But the one-way puzzle security game requires running the verification algorithm, which has no guarantees on efficiency. And so the obvious approach to amplifying security falls short in this setting.

But what can we do? Our key observation is that strong one-way puzzles can be built from EFID pairs (which we recall is the classical version of an EFI pair). In addition, [KT24] shows that one-way puzzles can be used to build EFI pairs, and along the way they show that strong one-way puzzles can be used to build a variant of EFID pairs, which we here call a non-uniform EFID pair. Unfortunately, their techniques do not work for weak one-way puzzles, an issue we remedy in Section 2.1.

The outline of our argument is to use our improved construction of EFI pairs from one-way puzzles from Section 2.1, which shows that we can build non-uniform EFID pairs from weak one-way puzzles as well. We then show how to build strong one-way puzzles from non-uniform EFID pairs.

Building strong one-way puzzles from non-uniform EFID Recall, a non-uniform cryptographic primitive is a cryptographic primitive where the construction takes in a short advice string of length $O(\log \lambda)$. Our construction of strong one-way puzzles from EFID pairs then allows us to build a non-uniform strong one-way puzzle from a non-uniform EFID pair. For each possible advice string s, we can instantiate the non-uniform strong one-way puzzle with s to get a new strong one-way puzzle candidate. For each security parameter, one of these candidates is a strong one-way puzzle. Thus, using a combiner on all of these candidates simultaneously produces a strong one-way puzzle which does not need any advice string.

We remark that in general, if we have a robust combiner for a primitive then we can turn any non-uniform construction of that primitive into its full version.

Amplifying Correctness To amplify correctness, we observe that our correctness guaranteer will always increase correctness at some cost to security. By carefully tracking this cost and interleaving with security amplification, it is possible to boost to full correctness without hurting security.

2.4 Relationships with other QCCC primitives

It has been shown in [KT24] that one-way puzzles can be built from almost all QCCC style primitives. In particular, they show how to build a one-way puzzle from a digital signature, a symmetric encryption protocol, or a commitment scheme. Just as a one-way function can be built from any useful classical primitive, a one-way puzzle can be built from any useful QCCC primitive.

Note that minimality of a primitive is not very hard to achieve. As an example, any primitive which can be built unconditionally is minimal in the same sense as a one-way function. But, importantly, one-way functions are also *useful*. One-way functions can be used to construct a large class of important cryptographic primitives, often referred to as symmetric key primitives or Minicrypt. In particular, one-way functions can be used to build (classical) digital signatures, symmetric encryption protocols, and commitment schemes. Thus, if we want to treat one-way puzzles as a central primitive for QCCC cryptography, it seems important that one-way puzzles imply at least one more directly useful QCCC primitive.

Unfortunately, existing results in this direction are noticeably weaker. In particular, as far as the authors are aware, the only QCCC primitive for which a construction is known from one-way puzzles is non-uniform EFID pairs [KT24]. On the other hand, there are some *quantum output* implications of one-way puzzles. In particular, it is known how to build commitments with quantum output (and all equivalent primitives) from one-way puzzles [KT24].

2.5 Efficiently verifiable one-way puzzles

The key challenge to using one-way puzzles to build other primitives is that the constructions may not make use of the verification scheme in a black-box manner since the verification scheme is not itself efficient. Thus, to make the problem easier, we consider a variant of one-way puzzles with efficient verification.

A very similar primitive, termed "hard quantum planted problems for QCMA," has been studied before in the context of publicly verifiable deletion [KNY23]. A hard quantum planted problem is essentially an efficiently verifiable one-way puzzle with perfect correctness. Direct observation shows that perfect correctness is unnecessary for any of

the applications of hard quantum planted problems for QCMA, and so their results hold for efficiently verifiable one-way puzzles as well. In particular, they show that

Theorem 6 (Theorem 6.2 from [KNY23])**.** *If there exists an efficiently verifiable one-way puzzle and quantum*

$$Z \in \{SKE, COM, PKE, ABE, QFHE, TRE, WE\},$$

then there exists Z with publicly verifiable deletion.

Their construction requires building a stronger variant of a one-time signature scheme from efficiently verifiable one-way puzzles. In particular

Theorem 7 (Theorem 3.2 from [KNY23])**.** *If there exists an efficiently verifiable one-way puzzle, then there exists a QCCC one-time signature scheme.*

The construction is essentially just a Lamport signature [Lam79]. As this theorem is not presented with full proof details in [KNY23], for completeness we restate this claim as Theorem 14 and give a full proof.

Since efficiently verifiable one-way puzzles seem more useful than normal one-way puzzles, we might wonder whether they are also minimal. Fortunately, most of the constructions of one-way puzzles from QCCC primitives have efficient verification algorithms. The two notable exceptions are EFID pairs and commitment schemes.

Theorem 8 (Theorems A.4 and A.6 from [KT24] and Theorems 15 and 16 in this paper)**.** *If there exists a QCCC signature scheme, secret key encryption scheme, non-interactive commitment scheme, or pseudodeterministic PRG, then there exists an efficiently verifiable one-way puzzle.*

Applying this theorem to the results of [KNY23] then gives the following two interesting corollaries.

Corollary 2. *If there exists QCCC*

$$Z \in \{SKE, PKE, ABE, QFHE, TRE\}$$

then there exists Z with publicly verifiable deletion.

Corollary 3. *There exists an efficiently verifiable one-way puzzle if and only if there exists a QCCC one-time signature scheme.*

Amplification and combiners for efficiently verifiable one-way puzzles. Since efficiently verifiable one-way puzzles seem to be about as minimal for QCCC as one-way puzzles, but have much more powerful applications, we may consider whether efficiently verifiable one-way puzzles should instead be considered a "central" primitive for QCCC cryptography. We then may hope that there exists an amplification theorem and a universal construction for efficiently verifiable one-way puzzles. We show that this is indeed the case.

Theorem 9 (Restatement of Theorem 20)**.** *If there exists a $(1 - 1/\mathsf{poly}(\lambda), \mathsf{negl}(\lambda))$ efficiently verifiable one-way puzzle, then there exists a $(\mathsf{negl}(\lambda), \mathsf{negl}(\lambda))$ efficiently verifiable one-way puzzle.*

Theorem 10 (Restatement of Theorem 24)**.** *If there exists a $(\mathsf{negl}(\lambda), 1 - 1/\mathsf{poly}(\lambda))$ efficiently verifiable one-way puzzle, then there exists a $(\mathsf{negl}(\lambda), \mathsf{negl}(\lambda))$ efficiently verifiable one-way puzzle.*

Theorem 11 (Informal version of Corollary 11)**.** *There exists a robust combiner for efficiently verifiable one-way puzzles.*

Corollary 4 (Informal version of Theorem 25)**.** *There exists a pair of algorithms* $(\mathsf{Samp}_U, \mathsf{Ver}_U)$ *such that as long as efficiently verifiable one-way puzzles exist,* $(\mathsf{Samp}_U, \mathsf{Ver}_U)$ *is an efficiently verifiable one-way puzzle.*

Note that most of the barriers to these results go away when the verification algorithm is required to be efficient. Thus, the "naive" constructions described earlier are provably secure for efficiently verifiable one-way puzzles.

Are one-way puzzles equivalent to efficiently verifiable one-way puzzles? Although the advantage of treating efficiently verifiable one-way puzzles as a "central" QCCC primitive is that it has actual applications in the QCCC setting, this does come at a cost to its "minimality". It is not clear how to build efficiently verifiable one-way puzzles from every primitive known to imply OWPuzz. In particular, constructions are lacking from EFID pairs and commitments.

Thus, we may consider whether or not it even matters whether verification is efficient. Ideally, we would be able to build an efficiently verifiable one-way puzzle from any one-way puzzle. In fact, if we restrict the sampling algorithm to being a classical randomized algorithm, such a claim holds true. Given a classical one-way puzzle, we can build an efficiently verifiable one-way puzzle by replacing the key with the random coins of the sampler. Then, the verifier can simply check whether running the sampler on the randomness given produces the given puzzle.

However, as this approach directly uses the randomness of sampling, it is inherently non-quantum. In fact, it turns out that in the quantum setting, there is a black-box separation

Theorem 12 (Informal version of Theorem 19)**.** *There exists a quantum oracle* $\mathcal{O}$ *relative to which one-way puzzles exist but efficiently verifiable one-way puzzles do not exist.*

This theorem follows from a simple observation. A search-to-decision argument shows that any efficiently verifiable one-way puzzle can be broken using a QCMA oracle. But there exists an oracle relative to which pseudorandom states exist and BQP=QCMA [Kre21]. As pseudorandom states can be used to build one-way puzzles [KT24, MY22], Theorem 12 follows.

A barrier against length shrinking for pseudorandom states An open question in the literature is whether pseudorandom states with output length $n(\lambda)$ can be built from pseudorandom states with output length $n'(\lambda)$ for any values of n, n' such that $n \neq n' \geq \log n$. However, pseudorandom states with output length $O(\log \lambda)$ can be used to build QCCC pseudodeterministic PRGs, and thus efficiently verifiable one-way puzzles. But our argument gives a black-box separation between efficiently verifiable one-way puzzles and pseudorandom states with output length λ. Thus, we get the following corollary

Corollary 5 (Informal version of Corollary 8)**.** *There exists a quantum oracle* $\mathcal{O}$ *relative to which PRSs with output length* λ *exist but PRSs with output length* $c \log \lambda$ *(for* $c > 12$*) do not.*

Note that this observation at its core comes from the simple observation that pseudodeterministic PRGs can be broken with a QCMA oracle, and so this observation is

little more than a corollary of the results of [ALY23b, Kre21], and is known in folklore. However, we provide a full formal proof of this statement as a contribution towards the systemization of knowledge in quantum cryptography.

2.6 Random Input OWPuzz

Another natural variant of one-way puzzles we might consider is a one-way puzzle where the key must be sampled uniformly at random, and then the puzzle is sampled from the key. We will call this a random input one-way puzzle. This more closely aligns with the classical notion of one-way functions, and in fact the construction of one-way puzzles from one-way state generators produces a random input one-way puzzle (assuming the key generation for the one-way state generator is uniform) [KT24].

[KT24] left as an open question whether random input one-way puzzles can be built from arbitrary one-way puzzles. Note that this statement does hold classically, since both are equivalent to one-way functions.

We show that these two notions are indeed equivalent

Theorem 13 (Restatement of Theorem 31)**.** *If there exists a one-way puzzle, then there exists a random input one-way puzzle. If there exists an efficiently verifiable one-way puzzle, then there exists a random input efficiently verifiable one-way puzzle.*

The idea is fairly natural. We simply treat the random input as a one-time pad, and apply it to the original key. We then include the one-time padded key with the original puzzle in the final output.

Note that our amplification lemma for one-way puzzles also produces a random input one-way puzzle, and so also gives an indirect proof of this theorem, although this approach does not hold for efficiently verifiable one-way puzzles.

3 Open Questions

Although we are aware of a few implications, the landscape of QCCC reductions, even those relating to one-way puzzles/efficiently verifiable one-way puzzles, is still fairly unexplored. We list a few interesting questions in this space related to our work

1. Is it possible to build efficiently verifiable one-way puzzles from a QCCC commitment scheme? QCCC commitments and EFID pairs are the two QCCC primitives for which the obvious construction of one-way puzzles does not have an efficient verifier. If the answer to this question is no, then it may be possible to build QCCC commitments from standard one-way puzzles.

2. Are there any useful cryptographic primitives we can construct from one-way puzzles without efficient verification? Due to the black-box separation between one-way puzzles and efficiently verifiable one-way puzzles, it seems like the answer may be no. However, a few primitives (such as EFID pairs and QCCC commitments) fall outside of this separation, and so there is still hope for a construction.

3. Is there a combiner for QCCC EFID pairs? If so, by using the construction of non-uniform EFID from one-way puzzles, we would be able to construct standard EFID from one-way puzzles. Interestingly, there does exist a combiner for both the quantum version and the classical version of this primitive [HKNY23, Lev87, Gol90].

4. Can we build any QCCC primitives besides one-time signatures from efficiently verifiable one-way puzzles, for example secret key encryption or pseudodeterministic PRGs? What about many-time signatures? Although we observe that the known construction of many time signatures from one-way functions uses a pseudorandom function in order to be stateless, so this may necessitate building a QCCC style pseudorandom function [GMR87].

4 Preliminaries

4.1 Definitions of QCCC primitives

As discussed previously this definition of OWPuzz slightly generalizes the notion given in [KT24].

Definition 1. *An (α, β) one way puzzle (OWPuzz) is a pair of a sampling algorithm and a verification function $(\mathsf{Samp}, \mathsf{Ver})$ with the following syntax:*

1. $\mathsf{Samp}(1^\lambda) \to (k, s)$ *is a uniform QPT algorithm which outputs a pair of classical strings (k, s). We refer to s as the puzzle and k as the key. Without loss of generality, we can assume $k \in \{0,1\}^\lambda$.*

2. $\mathsf{Ver}(k, s) \to b$ *is some (possibly uncomputable) function which takes in a key and puzzle and outputs a bit $b \in \{0, 1\}$.*

satisfying the following properties:

1. *Correctness: Outputs of the sampler pass verification with overwhelming probability*

$$\Pr_{\mathsf{Samp}(1^\lambda)\to(k,s)}[\mathsf{Ver}(k, s) \to 1] \geq 1 - \alpha$$

2. *Security: Given a puzzle s, it is computationally infeasible to find a key s which verifies. That is, for all non-uniform QPT algorithms $\mathcal{A}$,*

$$\Pr_{\mathsf{Samp}(1^\lambda)\to(k,s)}[\mathsf{Ver}(\mathcal{A}(s), s) \to 1] \leq \beta$$

If for all c, $(\mathsf{Samp}, \mathsf{Ver})$ is a $(\lambda^{-c}, \lambda^{-c})$ one way puzzle, then we say that $(\mathsf{Samp}, \mathsf{Ver})$ is a strong OWPuzz and omit the constants. When unambigious, we will simply say that $(\mathsf{Samp}, \mathsf{Ver})$ is a OWPuzz.

Definition 2. *A one-time signature scheme is a set of QPT algorithms $(KeyGen, S, V)$ with the following syntax*

1. $KeyGen(1^\lambda) \to (vk, sk)$ *takes the security parameter as input and ouptuts a signing key sk and a verification key vk*

2. $S(sk, m) \to \sigma$ *takes in the signing key and a message as input, and outputs a signature σ*

3. $V(vk, m, \sigma) \to 0/1$ *takes in a verification key vk, a message m, and a signature σ, and outputs a single bit*

satisfying the following security properties

1. *Correctness: For all m in the message space,*

$$\Pr_{KeyGen(1^\lambda)\to(vk,sk)}[V(vk,m,S(sk,m))\to 1]\geq 1-\mathsf{negl}(\lambda)$$

2. *One-time Security: An adversary with the ability to make one signature query can not forge a signature for a different message. More formally, for all $m_0\neq m_1$ in the message space and for all PPT $\mathcal{A}$,*

$$\Pr_{KeyGen(1^\lambda)\to(vk,sk)}[V(vk,m_1,\mathcal{A}(vk,S(sk,m_0)))\to 1]\leq \mathsf{negl}(\lambda)$$

Definition 3 (Pseudodeterministic Quantum Pseudorandom Generator [ALY23a])**.** *A pseudodeterministic quantum pseudorandom generator* (QPRG) *is a uniform QPT algorithm G that on input a classical seed $k\in\{0,1\}^{n(\lambda)}$ outputs a string of length $\ell(\lambda)$ with the following guarantees:*

1. *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*

$$\Pr_{\{0,1\}^\lambda\to k}[k\in\mathcal{K}_\lambda]\geq 1-\mu(\lambda)$$

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

2. *Stretch: $\ell(\lambda)>n(\lambda)$*

3. *Security: For every non-uniform QPT algorithm $\mathcal{A}$,*

$$\left|\Pr_{\{0,1\}^\lambda\to k}[\mathcal{A}(G(k))\to 1]-\Pr_{\{0,1\}^{\ell(\lambda)}\to y}[\mathcal{A}(y)\to 1]\right|\leq\mathsf{negl}(\lambda)$$

Definition 4 (Commitment scheme from [KT24])**.** *A commitment scheme is an efficient two-party protocol between a committer Com and a receiver Rec consisting of a commit stage and an opening stage operating on a private input m described as follows*

1. *Commit stage: both parties receive a unary security parameter 1^λ. The committer Com receives a private input m. It interacts with the receiver Rec using only classical messages, and together they produce a transcript z. At the end of the stage, both parties hold a private quantum state ρ_{Com} and ρ_{Rec} respectively.*

2. *Opening stage: both parties receive the transcript z as well as their private quantum states ρ_{Com} and ρ_{Rec} respectively. They then interact using only classical messages. At the end of the stage, the receiver either outputs a message or the reject symbol $\perp$.*

satisfying the following two properties

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

2. *(Computational) hiding: For all $m\neq m'$ and for all QPT adversarial receivers Rec', the transcript of the interaction between the adversarial receiver and the committer with input m is indistinguishable from the transcript of the interaction between the adversarial receiver and the committer with input m'. That is,*

$$Com(m)\rightleftarrows Rec'\approx Com(m')\rightleftarrows Rec'$$

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

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

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

If the receiver never sends any messages in either stage, then we say the commitment scheme is non-interactive. In this case, we write $Com(m) \to (c, d)$ where c is the message sent in the first round (the commitment) and d is the message sent in the second round (the decommitment). We then describe the final output of Rec by $Rec(c, d) \to m'$.

Definition 5. *An EFID pair is a randomized algorithm $\mathrm{Gen}(1^\lambda, b)$ taking a unary security parameter λ and a classical bit $b \in \{0, 1\}$ which outputs a classical string satisfying the following two properties:*

1. *Statistically far:*
$$\Delta(\mathrm{Gen}(1^\lambda, 0), \mathrm{Gen}(1^\lambda, 1)) \geq 1 - \epsilon$$

2. *Computationally close: For all QPT $\mathcal{A}$ and for all sufficiently large λ, the distributions $\mathrm{Gen}(1^\lambda, 0)$ and $\mathrm{Gen}(1^\lambda, 1)$ are indistinguishable.*

If Gen *is a quantum algorithm (with classical output), then we call* Gen *a quantum EFID pair (or QEFID).*

4.2 Complexity

Definition 6. *We say a promise problem $\Pi : \{0, 1\}^* \to \{0, 1, \perp\}$ is in* Promise QCMA *if there exists a QPT algorithm $\mathcal{V}(x, y)$ and a polynomial p such that:*

1. *Completeness: If $\Pi(x) = 1$, then there exists a $p(|x|)$-bit string y such that*
$$\Pr[\mathcal{V}(x, y) \to 1] \geq \frac{2}{3}$$

2. *Soundness: If $\Pi(x) = 0$, then for all $p(|x|)$-bit strings y,*
$$\Pr[\mathcal{V}(x, y) \to 1] \leq \frac{1}{3}$$

Definition 7. *We say a promise problem $\Pi : \{0, 1\}^* \to \{0, 1, \perp\}$ is in* Promise QMA *if there exists a QPT algorithm $\mathcal{V}(x, |\phi\rangle)$ and a polynomial p such that:*

1. *Completeness: If $\Pi(x) = 1$, then there exists a $p(|x|)$-qubit state $|\phi\rangle$ such that*
$$\Pr[\mathcal{V}(x, |\phi\rangle) \to 1] \geq \frac{2}{3}$$

2. *Soundness: If $\Pi(x) = 0$, then for all $p(|x|)$-qubit states $|\phi\rangle$,*
$$\Pr[\mathcal{V}(x, |\phi\rangle) \to 1] \leq \frac{1}{3}$$

4.3 Oracles

We define, in the spirit of Kretschmer[Kre21], a query to a single unitary $\mathcal{U}$ to be a single quantum call of either $\mathcal{U}$ or controlled-$\mathcal{U}$. We do not allow queries to $\mathcal{U}^\dagger$. $\mathcal{A}^{\mathcal{U}}(x)$ refers to a quantum algorithm on a classical input x which can make quantum queries to the unitary (or collection of unitaries) $\mathcal{U}$. In terms of computational cost, a single query to $\mathcal{U}_n$ will be charged n units of computation. This allows us to define quantum polynomial-time (QPT) algorithms relative to an oracle $\mathcal{U}$. In particular, a QPT algorithm relative to $\mathcal{U}$ on an input of length ℓ can query $\mathcal{U}_n$ for any $n < \mathsf{poly}(\ell)$.

Also in the style of Kretschmer, we consider versions of PromiseBQP, Promise QCMA, and PromiseQMA augmented with a collection of quantum oracles $\mathcal{U} = \{\mathcal{U}_n\}_{n\in\mathbb{N}}$. We denote these by $\mathsf{PromiseBQP}^{\mathcal{U}}$, $\mathsf{PromiseQCMA}^{\mathcal{U}}$, and $\mathsf{PromiseQMA}^{\mathcal{U}}$ respectively. For $\mathsf{PromiseBQP}^{\mathcal{U}}$, the deciding algorithm is allowed to be a QPT algorithm relative to $\mathcal{U}$, and for $\mathsf{PromiseQCMA}^{\mathcal{U}}$ and $\mathsf{PromiseQMA}^{\mathcal{U}}$, the verifying algorithm is allowed to be a QPT algorithm relative to $\mathcal{U}$.

It is easy to see that in this model, the traditional inequalities still hold. In particular, for any oracle $\mathcal{U}$, $\mathsf{PromiseBQP}^{\mathcal{U}} \subseteq \mathsf{PromiseQCMA}^{\mathcal{U}} \subseteq \mathsf{PromiseQMA}^{\mathcal{U}}$.

We also consider cryptographic primitives in the oracle setting. In this case, we allow the cryptographic algorithm to be a uniform QPT algorithm relative to $\mathcal{U}$, and we consider security against non-uniform QPT algorithms relative to $\mathcal{U}$.

5 Constructions of $\mathsf{EV}-\mathsf{OWPuzz}$ from QCCC primitives

In this section we give the results that $\mathsf{EV}-\mathsf{OWPuzz}$ are equivalent to QCCC one time signatures, and can be constructed from QCCC non-interactive commitments and QPRGs.

Theorem 14. *There exists a one-time signature scheme if and only if there exists a* $\mathsf{EV}-\mathsf{OWPuzz}$.

Proof of theorem 14. [KT24] show that you can construct OWPuzz from one-time signature schemes (in fact zero-time signature schemes). They do not define $\mathsf{EV}-\mathsf{OWPuzz}$, but it is clear that their construction has efficient verification. We repeat their construction here for completeness.

Let $(KeyGen, S, V)$ be a one-time signature scheme. Then $(\mathsf{Samp}, \mathsf{Ver})$ defined as follows is a $\mathsf{EV}-\mathsf{OWPuzz}$

1. $\mathsf{Samp}(1^\lambda)$: Sample $KeyGen(1^\lambda) \to (vk, sk)$. Output $(k = vk, s = sk)$
2. $\mathsf{Ver}(k, s)$: Sample m uniformly at random from the message space. Output $V(s, m, S(k, m))$.

To show the other direction (that $\mathsf{EV}-\mathsf{OWPuzz} \to$ signitures), it is not hard to see that the Lamport signature scheme [Lam79] building one-time signatures from one-way functions can be generalized to work with $\mathsf{EV}-\mathsf{OWPuzz}$. In particular, let $(\mathsf{Samp}, \mathsf{Ver})$ be a $\mathsf{EV}-\mathsf{OWPuzz}$, we define a signature scheme using it as follows. For simplicity, the message space will be $\{0, 1\}$.

1. $KeyGen(1^\lambda)$: Run Samp twice to generate two key-puzzle pairs (k_0, s_0) and (k_1, s_1). Output $(vk = (s_0, s_1), sk = (k_0, k_1))$.
2. $S((k_0, k_1), b)$: Output k_b.
3. $V((s_0, s_1), b, \sigma)$: Output $\mathsf{Ver}(\sigma, s_b)$.

Correctness is immediate from correctness of the one-way-puzzle scheme. To show security, we will assume towards contradiction that their exists some pair of messages $m_0 \neq m_1$ and an adversary $\mathcal{A}$ breaking security of the signature scheme. Without loss of generality we will assume $m_0 = 0$ and $m_1 = 1$. Thus,

$$\Pr[V(vk, 1, \mathcal{A}(vk, S(sk, 0))) \to 1] > \lambda^{-c}$$

for some c. Rewriting this in the notation of the underlying one way puzzle we have

$$\Pr_{\mathsf{Samp}(1^\lambda)\to(k_0,s_0),(k_1,s_1)}[\mathsf{Ver}(\mathcal{A}((s_0,s_1),k_0),s_1)] > \lambda^{-c}$$

We will define a new adversary $\mathcal{B}$ breaking the one-way puzzle as follows. On input s, $\mathcal{B}$ runs $\mathsf{Samp}(1^\lambda) \to (k', s')$ and outputs $\mathcal{A}((s, s'), k')$. It is clear that

$$\Pr_{\mathsf{Samp}(1^\lambda)\to(k,s)}[\mathsf{Ver}(\mathcal{B}(s), s) \to 1]$$

$$= \Pr_{\mathsf{Samp}(1^\lambda)\to(k_0,s_0),(k_1,s_1)}[\mathsf{Ver}(\mathcal{A}((s_0,s_1),k_0),s_1)] > \lambda^{-c}$$

But as $(\mathsf{Samp}, \mathsf{Ver})$ is a $\mathsf{EV}-\mathsf{OWPuzz}$, this is a contradiction, and so the one-time signature scheme is secure. □

[KT24] shows that $\mathsf{EV}-\mathsf{OWPuzz}$ can be built from one-time signatures even if the signing key or the signature are quantum. Thus, an interesting corollary of Theorem 14 is that QCCC one-time signatures with classical signature, signing and verification keys can be built from one-time signatures where either the signing key or the signature is quantum.

Theorem 15. *If there exists a non-interactive commitment scheme (Com, Rec), then there exists a $\mathsf{EV}-\mathsf{OWPuzz}$ $(\mathsf{Samp}, \mathsf{Ver})$.*

Proof of theorem 15. Our construction is as follows

1. Samp: Pick m uniformly at random. Run $Com(m) \to (c, d)$. Output $(k = (m, d), s = c$.
2. $\mathsf{Ver}(k = (m, d), s = c)$: Run $Rec(c, d) \to m'$. Output 1 if and only if $m' = m$.

Correctness immediately implies that

$$\Pr_{\mathsf{Samp}\to(k,s)}[\mathsf{Ver}(k,s)] = \Pr_{\$\to m, Com(m)\to(c,d)}[Rec(c,d) = m] \geq 1 - \mathsf{negl}(\lambda)$$

We now proceed to show security. Let $\mathcal{A}$ be any QPT adversary. We will show

$$\Pr_{\mathsf{Samp}\to(k,s)}[\mathsf{Ver}(\mathcal{A}(s), s) \to 1] \leq \mathsf{negl}(\lambda)$$

Observe that

$$\begin{aligned}
&\Pr_{\mathsf{Samp}\to(k,s)}[\mathsf{Ver}(\mathcal{A}(s), s) \to 1] \\
&= \Pr_{Com(m)\to(c,d)}[Rec(c,d') = m'; \mathcal{A}(c) \to (d', m')] \\
&= \Pr_{Com(m)\to(c,d)}[Rec(c,d') = m' \wedge m = m'; \mathcal{A}(c) \to (d', m')] \\
&+ \Pr_{Com(m)\to(c,d)}[Rec(c,d') = m' \wedge m \neq m'; \mathcal{A}(c) \to (d', m')]
\end{aligned}$$

But hiding implies that the probability that $\mathcal{A}$ computes m from c is negligible, so

$$\Pr_{Com(m)\to(c,d)}[Rec(c,d') = m' \wedge m = m'; \mathcal{A}(c) \to (d', m')] \leq \mathsf{negl}(\lambda)$$

And binding says that after an honest commitment, there is no way to open to a different message, and so

$$\Pr_{Com(m)\to(c,d)}[Rec(c,d') = m' \wedge m \neq m'; \mathcal{A}(c) \to (d', m')] \leq \mathsf{negl}(\lambda)$$

Together, we have

$$\Pr_{\mathsf{Samp}\to(k,s)}[\mathsf{Ver}(\mathcal{A}(s), s) \to 1] \leq \mathsf{negl}(\lambda)$$

□

Theorem 16. *If there exists a* QPRG G *with stretch* $\ell(\lambda) \geq 3n(\lambda)$*, then there exists a* $\mathsf{EV} - \mathsf{OWPuzz}$ $(\mathsf{Samp}, \mathsf{Ver})$*.*

Proof of theorem 16. We define our $\mathsf{EV} - \mathsf{OWPuzz}$ as follows

1. $\mathsf{Samp}(1^\lambda)$: Sample $k_1, \ldots, k_\lambda$ uniformly from $\{0,1\}^n$. Sample $G_\lambda(k_i) \to s_i$. Output key $k = k_1, \ldots, k_\lambda$, and puzzle $s = s_1, \ldots, s_\lambda$.
2. $\mathsf{Ver}(k, s)$: For each i sample $G(k_i) \to \widetilde{s_i}$. If for any i, $s_i = \widetilde{s_i}$, output 1. Otherwise, output 0.

We first will show correctness. Pseudodeterminism of G gives us that for sufficiently large λ, the probability over a random k that two runs of $G(k)$ give the same result is $\geq \frac{1}{2}$. Thus,

$$\begin{aligned}\Pr_{\mathsf{Samp}(1^\lambda)\to(k,s)}[\mathsf{Ver}(k,s) \to 0] &= \prod_{i=1}^{\lambda} \Pr_{\{0,1\}^n \to k_i, G(k_i)\to s_i, G(k_1)\to \widetilde{s_i}}[s_i = \widetilde{s_i}] \\ &\leq \prod_{i=1}^{\lambda} \frac{1}{2} \leq \frac{1}{2^\lambda} \leq \mathsf{negl}(\lambda)\end{aligned}$$

which gives us correctness.

We will now argue security via a reduction. Let $\mathcal{A}$ be any non-uniform QPT algorithm such that

$$\Pr_{\mathsf{Samp}(1^\lambda)\to(k,s)}[\mathsf{Ver}(\mathcal{A}(s), s) \to 1] \geq \epsilon$$

By an averaging argument, there must exist some index $i \in [\lambda]$ such that

$$\Pr_{\mathsf{Samp}(1^\lambda)\to(k,s)}[G((\mathcal{A}(s))_i) = s_i] \geq \frac{\epsilon}{\lambda}$$

We define an adversary $\mathcal{A}'$ against G as follows

1. On input y
2. Sample $k_1, \ldots, k_{i-1}, k_{i+1}, \ldots, k_\lambda$ uniformly from $\{0,1\}^n$
3. Send $G(k_1), \ldots, G(k_{i-1}), y, G(k_{i+1}), \ldots, G(k_\lambda)$ to $\mathcal{A}$, getting response $\widetilde{k}$
4. If $G(\widetilde{k_i}) = y$, then output 1, otherwise output 0,

First, we will lower bound $\Pr_{\{0,1\}^n \to x}[\mathcal{A}'(G(x)) \to 1]$. Note that on input $y = G(x)$, the view of $\mathcal{A}$ is exactly as it should be in its own game, and so

$$\Pr_{\{0,1\}^n \to x}[\mathcal{A}'(G(x)) \to 1] = \Pr_{\mathsf{Samp}(1^\lambda) \to (k,s)}[G((\mathcal{A}(s))_i) = s_i] \geq \frac{\epsilon}{\lambda}$$

Now, we will upper bound $\Pr_{\{0,1\}^\ell \to y}[\mathcal{A}'(G(y)) \to 1]$. Note that the optimal adversary for this problem, on input y, returns $\mathrm{argmax}_x \Pr[G(x) = y]$. Thus, we have the following trivial bound

$$\Pr_{\{0,1\}^\ell \to y}[\mathcal{A}'(G(y)) \to 1] \leq \mathop{\mathbb{E}}_{\{0,1\}^\ell \to y}[\max_x \Pr[G(x) = y]]$$

But there can be at most 2^{2n} values of y such that $\max_x \Pr[G(x) = y] \geq \frac{1}{2^n}$. To see this, observe

$$\sum_y \max_x \Pr[G(x) = y] \leq \sum_{x,y} \Pr[G(x) = y] = \sum_x \sum_y \Pr[G(x) = y] = \sum_x 1 = 2^n$$

But since $\ell(\lambda) \geq 3n(\lambda)$, we have that

$$\Pr_{\{0,1\}^\ell \to y}[\max_x \Pr[G(x) = y] \geq 2^{-n}] \leq \frac{2^{2n}}{2^\ell} \leq \frac{2^{2n}}{2^{3n}} = 2^{-n}$$

Thus, we conclude with

$$\begin{aligned}\mathop{\mathbb{E}}_{\{0,1\}^\ell \to y}[\max_x \Pr[G(x) = y]] &\leq \Pr_{\{0,1\}^\ell \to y}[\max_x \Pr[G(x) = y] \geq 2^{-n}] \cdot 1 + 2^{-n} \\ &\leq 2^{-n} + 2^{-n}\end{aligned}$$

Putting this together, we get that

$$\Pr_{\{0,1\}^n \to x}[\mathcal{A}'(G(x)) \to 1] - \Pr_{\{0,1\}^\ell \to y}[\mathcal{A}'(y) \to 1] \geq \frac{\epsilon}{\lambda} + 2^{-n+1}$$

and so since

$$\Pr_{\{0,1\}^n \to x}[\mathcal{A}'(G(x)) \to 1] - \Pr_{\{0,1\}^\ell \to y}[\mathcal{A}'(y) \to 1] \leq \mathsf{negl}(\lambda)$$

by QPRG security, we get that $\epsilon \leq \mathsf{negl}(\lambda)$ and so $(\mathsf{Samp}, \mathsf{Ver})$ satisfies security. □

6 Efficiently verifiable one way puzzles can be broken with a QCMA oracle

Theorem 17. *For every efficiently verifiable one way puzzle* $(\mathsf{Samp}, \mathsf{Ver})$*, there exists a promise problem* $\Pi \in \mathsf{PromiseQCMA}$ *and a QPT algorithm* $\mathcal{A}^\Pi$ *with oracle access to* Π *which breaks security. That is*

$$\Pr_{\mathsf{Samp}(1^\lambda) \to (k,s)}[\mathsf{Ver}(\mathcal{A}(s), s) \to 1] \geq \frac{1}{\mathsf{poly}(\lambda)}$$

Proof. Let $(\mathsf{Samp}, \mathsf{Ver})$ be a $\mathsf{EV} - \mathsf{OWPuzz}$. Define Π to be the following promise problem:

1. (yes): (x, s) is a yes instance if there exists y such that $\Pr[\mathsf{Ver}(x||y, s) \to 1] \geq \frac{2}{3}$
2. (no): (x, s) is a no instance if for all y, $\Pr[\mathsf{Ver}(x||y, s) \to 1] < \frac{2}{3}$

It is trivial to see that $\Pi \in \mathsf{PromiseQCMA}$.

We will then show how to use oracle access to Π to break the $\mathsf{EV}-\mathsf{OWPuzz}$. This is just a standard search-to-decision reduction. The algorithm works as follows:

1. On input s, we define $k = k_1, \ldots, k_n$ bit by bit.
2. For $i = 1, \ldots, n$
   (a) Check if $((k_1, \ldots, k_{i-1}, 0), s)$ is a yes instance in Π. If so, set $k_i = 0$.
   (b) Otherwise, check if $((k_1, \ldots, k_{i-1}, 1), s)$ is a yes instance in Π. If so, set $k_i = 1$.
   (c) Otherwise, output $\perp$.
3. Output s.

Observe that as long as this algorithm does not output $\perp$, (k, s) is a yes instance in Π and so $\Pr[\mathsf{Ver}(k, s) \to 1] \geq \frac{2}{3}$.

We observe that if (x, s) is a yes instance of Π, then there exists $b \in \{0, 1\}$ such that $(x||b, s)$ is also a yes instance of Π. In particular, let y be such that $\Pr[\mathsf{Ver}(x||y, s) \to 1] \geq \frac{2}{3}$. It is clear that by setting $b = y_1$, we have that $(x||y_1, s)$ is a yes instance of Π.

Then, a simple induction argument says that as long as s is a yes instance of Π, then $\mathcal{A}(s)$ always outputs a key k such that $\Pr[\mathsf{Ver}(k, s) \to 1] \geq \frac{2}{3}$.

Correctness says that

$$\Pr_{\mathsf{Samp}(1^\lambda) \to (k,s)}[\mathsf{Ver}(k, s) \to 1] \geq 1 - \mathsf{negl}(\lambda)$$

and so an averaging argument gives us that

$$\Pr_{\mathsf{Samp}(1^\lambda) \to (k,s)}\left[\Pr[\mathsf{Ver}(k, s) \to 1] \geq \frac{2}{3}\right] \geq 1 - \mathsf{negl}(\lambda)$$

which means that with overwhelming probability, a sampled puzzle s will be a yes instance of Π.

Putting this together, we get that

$$\begin{aligned}
&\Pr_{\mathsf{Samp}(1^\lambda) \to (k,s)}[\mathsf{Ver}(\mathcal{A}(s), s) \to 1] \\
\geq &\Pr_{\mathsf{Samp}(1^\lambda) \to (k,s)}[\mathsf{Ver}(\mathcal{A}(s), s) \to 1 | \Pi(s) = 1] \Pr_{\mathsf{Samp}(1^\lambda) \to (k,s)}[\Pi(s) = 1] \\
&\geq \frac{2}{3}(1 - \mathsf{negl}(\lambda)) \\
&\geq \frac{1}{\mathsf{poly}(\lambda)}
\end{aligned}$$

□

7 A black-box separation between OWPuzz and EV − OWPuzz

We begin by recalling the very powerful quantum black-box separation theorem by Kretschmer.

Theorem 18 ([Kre21])**.** *There exists a set of quantum oracles $\mathcal{U}$ such that with probability 1 over $\mathcal{U}$,*

1. $\mathsf{PromiseBQP}^{\mathcal{U}} = \mathsf{PromiseQMA}^{\mathcal{U}}$.

2. *Relative to* $\mathcal{U}$*, there exists a PRS family mapping* λ *bits to* λ *qubit states.*

First, we observe that all of our theorems are black-box and thus relativize. In particular, we get the following corollaries

Corollary 6. *Let* $\mathcal{U}$ *be any collection of classical or quantum oracles. For every efficiently verifiable one way puzzle* $(\mathsf{Samp}^{\mathcal{U}}, \mathsf{Ver}^{\mathcal{U}})$*, there exists a promise problem* $\Pi \in \mathsf{PromiseQCMA}^{\mathcal{U}}$ *and a QPT algorithm* $\mathcal{A}^{\Pi}$ *with oracle access to* Π *which breaks security. That is*

$$\Pr_{\mathsf{Samp}^{\mathcal{U}}(1^\lambda)\to(k,s)}[\mathsf{Ver}^{\mathcal{U}}(\mathcal{A}^{\Pi}(s), s) \to 1] \geq \frac{1}{\mathsf{poly}(\lambda)}$$

Corollary 7. *Let* $\mathcal{U}$ *be any collection of classical or quantum oracles. If there exists a* QPRG $G^{\mathcal{U}}$ *with stretch* $\ell(\lambda) \geq 3n(\lambda)$ *secure relative to* $\mathcal{U}$*, then there exists a* $\mathsf{EV}-\mathsf{OWPuzz}$ $(\mathsf{Samp}^{\mathcal{U}}, \mathsf{Ver}^{\mathcal{U}})$ *secure relative to* $\mathcal{U}$*.*

Let us then consider the oracle $\mathcal{U}$ from Theorem 18. We know that relative to $\mathsf{PromiseBQP}^{\mathcal{U}} \subseteq \mathsf{PromiseQCMA}^{\mathcal{U}} \subseteq \mathsf{PromiseQMA}^{\mathcal{U}}$, and we also have $\mathsf{PromiseBQP}^{\mathcal{U}} = \mathsf{PromiseQMA}^{\mathcal{U}}$. Thus, this gives us that $\mathsf{PromiseBQP}^{\mathcal{U}} = \mathsf{Promise\ QCMA}^{\mathcal{U}}$. So Corollary 6 immediately shows that relative to $\mathcal{U}$, there does not exist any $\mathsf{EV}-\mathsf{OWPuzz}$.

But it is also known from [KT24] and [MY22] that OWPuzz can be built in a black-box manner from PRSs. Thus, we get the following corollary:

Theorem 19. *There exists a set of quantum oracles* $\mathcal{U}$ *such that with probability* 1 *over* $\mathcal{U}$*,*

1. *There does not exist any* $\mathsf{EV}-\mathsf{OWPuzz}$ *relative to* $\mathcal{U}$*.*

2. *There exists a* OWPuzz *relative to* $\mathcal{U}$*.*

[ALY23b] shows that PRSs with output length $c\log\lambda$ for $c > 12$ can be used to build QPRGs with triple stretch. Note that this reduction is itself black-box, and so holds relative to any quantum oracle. Applying Corollary 7 then gives the following result

Corollary 8. *There exists a set of quantum oracles* $\mathcal{U}$ *such that with probability* 1 *over* $\mathcal{U}$*, relative to* $\mathcal{U}$

1. *There does not exist any PRS family mapping* λ*-bits to* $c\log\lambda$*-qubits for any constant* $c > 12$*.*

2. *There does exist a PRS family mapping* λ*-bits to* λ*-qubits.*

8 OWPuzz Amplification, Combiners, and Universal Constructions

In this section we present security and correctness amplifiers, combiners, and universal constructions for both OWPuzz and $\mathsf{EV}-\mathsf{OWPuzz}$. We defer the proof of OWPuzz security amplification to the next section. With these results we establish that both primitives are well behaved and have many of the desirable properties of one-way functions.

8.1 Amplification

Theorem 20 (Correctness amplification for OWPuzz and EV − OWPuzz)**.** *Let* $(\mathsf{Samp}, \mathsf{Ver})$ *be a* (α, β) OWPuzz*. Define* $(\mathsf{Samp}', \mathsf{Ver}')$ *by*

1. $\mathsf{Samp}' = \mathsf{Samp}^{\otimes t}$
2. $\mathsf{Ver}'((k_1, \ldots, k_t), (s_1, \ldots, s_t))$*: output* 1 *if* $\mathsf{Ver}(k_i, s_i)$ *for some* $i \in [t]$.

Then $(\mathsf{Samp}', \mathsf{Ver}')$ *is a* $(\alpha^t, t\beta)$OWPuzz*.*

Proof of theorem 20. To see correctness, we observe that

$$\begin{aligned}
&\Pr_{\mathsf{Samp}'(1^\lambda)\to((k_1,\ldots,k_t),(s_1,\ldots,s_t))}[\mathsf{Ver}'((k_1,\ldots,k_t),(s_1,\ldots,s_t))=0] \\
&\quad= \Pr_{\mathsf{Samp}'(1^\lambda)\to((k_1,\ldots,k_t),(s_1,\ldots,s_t))}[\text{ all of } \mathsf{Ver}(k_i,s_i)=0] \\
&\quad\le \Pr_{\mathsf{Samp}(1^\lambda)\to(k,s)}[\mathsf{Ver}(k,s)=0] \le \alpha^t
\end{aligned}$$

To show security, we will do a simple reduction. Let $\mathcal{A}$ be such that

$$\Pr_{\mathsf{Samp}'(1^\lambda)\to((k_1,\ldots,k_t),(s_1,\ldots,s_t))}[\mathsf{Ver}(\mathcal{A}(s_1,\ldots,s_t),(s_1,\ldots,s_t))\to 1] > t\beta$$

We will construct an adversary $\mathcal{B}$ breaking $(\mathsf{Samp}, \mathsf{Ver})$.

1. On input s, pick i uniformly at random from $[t]$. Set $s_i = s$.
2. Run $\mathsf{Samp}(1^\lambda)$ $t-1$ times to generate s_j for $j \neq i$.
3. Output $\mathcal{A}(s_1, \ldots, s_t)$.

It is clear that

$$\begin{aligned}
&\Pr_{\mathsf{Samp}(1^\lambda)\to(k,s)}[\mathsf{Ver}(\mathcal{B}(s),s)\to 1] \\
&= \Pr_{\mathsf{Samp}'(1^\lambda)\to((k_1,\ldots,k_t),(s_1,\ldots,s_t)),[t]\to i}[\mathsf{Ver}(\mathcal{A}(s_1,\ldots,s_t)_i,s_i)] \\
&\ge \frac{1}{t}\Pr_{\mathsf{Samp}'(1^\lambda)\to((k_1,\ldots,k_t),(s_1,\ldots,s_t))}[\exists i \text{ s.t. } \mathsf{Ver}(\mathcal{A}(s_1,\ldots,s_t)_i,s_i)\to 1] \\
&\ge \frac{1}{t}\Pr_{\mathsf{Samp}'(1^\lambda)\to((k_1,\ldots,k_t),(s_1,\ldots,s_t))}[\mathsf{Ver}'(\mathcal{A}(s_1,\ldots,s_t),(s_1,\ldots,s_t))\to 1] > \frac{1}{t}t\beta > \beta
\end{aligned}$$

Thus, $(\mathsf{Samp}', \mathsf{Ver}')$ satisfies β security and we are done. □

Remark 1. *We note that if* Ver *is efficient, so is* Ver'*.*

Theorem 21 (Weak correctness amplification for OWPuzz)**.** *Let* (Samp, Ver) *be a* (α, β) OWPuzz*. Define* $(\mathsf{Samp}', \mathsf{Ver}')$ *by*

1. $\mathsf{Samp}' = \mathsf{Samp}$
2. $\mathsf{Ver}'(k, s)$*: If* $\Pr_{\mathsf{Samp}(1^\lambda)\to(k',s')}[\mathsf{Ver}(k',s')\to 0 | s'=s] \ge t$*, output* 1*. Otherwise, output* $\mathsf{Ver}(k, s)$*. (The idea here is that if* s *was sampled in a way that it would not verify honestly then we accept it anyway, otherwise we do normal verification)*

Then $(\mathsf{Samp}', \mathsf{Ver}')$ *is a* $(t, \alpha/t + \beta)$ OWPuzz*.*

Proof of theorem 21. To show correctness, we observe that

$$\begin{aligned}
&\Pr_{\mathsf{Samp}'(1^\lambda)\to(k,s)}[\mathsf{Ver}'(k,s)\to 0] \\
&= \Pr_{\mathsf{Samp}(1^\lambda)\to(k,s)}\left[\mathsf{Ver}(k,s)\to 0 \wedge \Pr_{\mathsf{Samp}(1^\lambda)\to(k',s')}[\mathsf{Ver}(k',s')\to 0|s'=s] < t\right] \\
&\leq \Pr_{\mathsf{Samp}(1^\lambda)\to(k,s)}\left[\mathsf{Ver}(k,s)\to 0| \Pr_{\mathsf{Samp}(1^\lambda)\to(k',s')}[\mathsf{Ver}(k',s')\to 0|s'=s] < t\right] \\
&< t
\end{aligned}$$

Before showing security, let us define X to be the random variable defined by sampling $\mathsf{Samp}(1^\lambda)\to(k,s)$ and outputting $\Pr_{\mathsf{Samp}(1^\lambda)\to(k',s')}[\mathsf{Ver}(k',s')\to 0|s'=s]$. We will begin by investigating $\Pr[X<t]$.

Observe that $\mathbb{E}[X] = \Pr_{\mathsf{Samp}(1^\lambda)\to(k,s)}[\mathsf{Ver}(k,s)\to 0] \leq \alpha$. Markov's inequality then gives that

$$\Pr[X\geq t]\leq \frac{\alpha}{t}.$$

Let $\mathcal{A}$ be any PPT algorithm

$$\begin{aligned}
&\Pr_{\mathsf{Samp}'(1^\lambda)\to(k,s)}[\mathsf{Ver}'(\mathcal{A}(s),s)\to 1] \\
&= \Pr_{\mathsf{Samp}'(1^\lambda)\to(k,s)}[\mathsf{Ver}(\mathcal{A}(s),s)\to 1 \vee X\geq t] \\
&\leq \Pr_{\mathsf{Samp}'(1^\lambda)\to(k,s)}[\mathsf{Ver}(\mathcal{A}(s),s)\to 1] + \Pr[X\geq t] \\
&\leq \beta + \frac{\alpha}{t}
\end{aligned}$$

□

Remark 2. *Note that this construction only holds for normal one-way puzzles, as* Ver' *is not efficient.*

Theorem 22 (Security amplification for OWP)**.** *If, for some* $c>0$*, there exists a* $(\mathsf{negl}(\lambda), 1-\lambda^{-c})$ *one-way puzzle* $(\mathsf{Samp},\mathsf{Ver})$*, then there exists a strong one-way puzzle.*

We defer the proof of the above theorem to the next section.

Theorem 23 (Weak correctness amplification for $\mathsf{EV}-\mathsf{OWPuzz}$)**.** *Let* $(\mathsf{Samp},\mathsf{Ver})$ *be a* (α,β) $\mathsf{EV}-\mathsf{OWPuzz}$*. Define* $(\mathsf{Samp}',\mathsf{Ver}')$ *by*

1. $\mathsf{Samp}'(1^\lambda)$*: Run* $\mathsf{Samp}(1^\lambda)\to(k,s)$*. If* $\mathsf{Ver}(k,s)\to 1$*, output* (k,s)*. Otherwise, output* $(\bot,\bot)$*.*
2. $\mathsf{Ver}'(k,s)$*: If* $s=\bot$*, output* 1*. Otherwise, output* $\mathsf{Ver}(k,s)$*.*

Then $(\mathsf{Samp}',\mathsf{Ver}')$ *is a* $(1/4,\alpha+\beta)$ OWPuzz*.*

Theorem 24 (Security amplification for $\mathsf{EV}-\mathsf{OWPuzz}$)**.** *Let* $(\mathsf{Samp},\mathsf{Ver})$ *be a* (α,β) $\mathsf{EV}-\mathsf{OWPuzz}$*. Then* $(\mathsf{Samp}^{\otimes t},\mathsf{Ver}^{\otimes t})$ *is a* $(t\alpha,\beta^t)$ $\mathsf{EV}-\mathsf{OWPuzz}$*.*

Proof of Theorem 24. Correctness follows from the union bound applied to correctness of $(\mathsf{Samp}, \mathsf{Ver})$:

$$\begin{aligned}
&\Pr_{\mathsf{Samp}^{\otimes t}(1^\lambda)\to((k_1,\ldots,k_t),(s_1,\ldots,s_t))}[\mathsf{Ver}^{\otimes t}((k_1,\ldots,k_t),(s_1,\ldots,s_t))\to 0]\\
=&\Pr_{\mathsf{Samp}^{\otimes t}(1^\lambda)\to((k_1,\ldots,k_t),(s_1,\ldots,s_t))}[\text{ there exists an } i \text{ such that } \mathsf{Ver}(k_i,s_i)\to 0]\\
&\leq \sum_i \Pr_{\mathsf{Samp}(1^\lambda)\to(k_i,s_i)}[\mathsf{Ver}(k_i,s_i)\to 0]\\
&\leq t\alpha
\end{aligned}$$

Security follows from Theorem 4.1 from [BQSY23]. □

‘

Remark 3. *The proof of Theorem 4.1 from [BQSY23] requires that the soundness game be efficiently falsifiable. Thus, this same amplification theorem does not hold for* OWPuzz *with inefficient verification. Amplification of such puzzles is an open question.*

8.2 Combiners

Corollary 9. *Let* $(\mathsf{Samp}, \mathsf{Ver})$ *be a* OWPuzz *candidate and define* $(\mathsf{Samp}', \mathsf{Ver}')$ *to be the constructions from Theorem 21 and Theorem 20 applied in sequence with* $t = 1/2$ *and* $t = \lambda$ *respectively. Then if* $(\mathsf{Samp}, \mathsf{Ver})$ *is a* OWPuzz*, so is* $(\mathsf{Samp}', \mathsf{Ver}')$*. Furthermore, regardless of whether* $(\mathsf{Samp}, \mathsf{Ver})$ *is a* OWPuzz*,* $(\mathsf{Samp}', \mathsf{Ver}')$ *satisfies* n^{-c} *correctness for all* c*.*

Proof. Let us say that $(\mathsf{Samp}, \mathsf{Ver})$ satisfies (α, β) correctness and security. Then $(\mathsf{Samp}', \mathsf{Ver}')$ satisfies $(2^{-\lambda}, \lambda(\alpha/2 + \beta))$ correctness and security.

Observe that if $\alpha, \beta = \mathsf{negl}(\lambda)$, then $\lambda(\alpha/2 + \beta) = \mathsf{negl}(\lambda)$. Furthermore, no matter what, $2^{-\lambda} = \mathsf{negl}(\lambda)$. □

Corollary 10. *Let* $(\mathsf{Samp}, \mathsf{Ver})$ *be a* $\mathsf{EV} - \mathsf{OWPuzz}$ *candidate and define* $(\mathsf{Samp}', \mathsf{Ver}')$ *to be the constructions from Theorem 23 and Theorem 20 applied in sequence with* $t = \lambda$*. Then if* $(\mathsf{Samp}, \mathsf{Ver})$ *is a* $\mathsf{EV} - \mathsf{OWPuzz}$*, so is* $(\mathsf{Samp}', \mathsf{Ver}')$*. Furthermore, regardless of whether* $(\mathsf{Samp}, \mathsf{Ver})$ *is a* $\mathsf{EV} - \mathsf{OWPuzz}$*,* $(\mathsf{Samp}', \mathsf{Ver}')$ *satisfies* n^{-c} *correctness for all* c*.*

Proof of Theorem 23. By correctness, $\Pr[\mathsf{Samp}'(1^\lambda) \to (\bot, \bot)] \leq \alpha$. Thus,

$$\begin{aligned}
&\Pr_{\mathsf{Samp}'(1^\lambda)\to(k,s)}[\mathsf{Ver}'(k,s)\to 0]\\
=&\Pr_{\mathsf{Samp}'(1^\lambda)\to(k,s)}[\mathsf{Ver}'(k,s)\to 0 \wedge s=\bot]\\
&+\Pr_{\mathsf{Samp}'(1^\lambda)\to(k,s)}[\mathsf{Ver}'(k,s)\to 0 \wedge s\neq\bot]\\
=&\Pr_{\mathsf{Samp}(1^\lambda)\to(k,s)}[\mathsf{Ver}(k,s)\to 0 \wedge \mathsf{Ver}(k,s)\to 1]\\
&\text{where these are two separate calls to } \mathsf{Ver}
\end{aligned}$$

Take any k, s. Say $p \coloneqq \Pr[\mathsf{Ver}(k,s) \to 1]$. Then $\Pr[\mathsf{Ver}(k,s) \to 0 \wedge \mathsf{Ver}(k,s) \to 1] = p(1-p) \leq \frac{1}{4}$. Thus,

$$\Pr_{\mathsf{Samp}'(1^\lambda)\to(k,s)}[\mathsf{Ver}'(k,s)\to 0] \leq 1/4$$

Let $\mathcal{A}$ be any PPT algorithm. We will prove security by showing that

$$\begin{aligned}\Pr_{\mathsf{Samp}'(1^\lambda)\to(k,s)}[\mathsf{Ver}'(\mathcal{A}(s),s)\to 1] &\leq \beta+\alpha\\ \Pr_{\mathsf{Samp}'(1^\lambda)\to(k,s)}[\mathsf{Ver}'(\mathcal{A}(s),s)\to 1] &= \Pr_{\mathsf{Samp}'(1^\lambda)\to(k,s)}[\mathsf{Ver}(\mathcal{A}(s),s)\to 1 \vee s=\bot]\\ &\leq \Pr_{\mathsf{Samp}'(1^\lambda)\to(k,s)}[\mathsf{Ver}(\mathcal{A}(s),s)\to 1]+\Pr_{\mathsf{Samp}'(1^\lambda)\to(k,s)}[s=\bot]\\ &= \alpha+\beta\end{aligned}$$

by the union bound. □

Proof. Same argument as before □

Corollary 11. *Let* $(\mathsf{Samp}_0,\mathsf{Ver}_0)$ *and* $(\mathsf{Samp}_1,\mathsf{Ver}_1)$ *be two* OWPuzz *candidates. Let* $(\mathsf{Samp}_0',\mathsf{Ver}_0')$ *and* $(\mathsf{Samp}_1',\mathsf{Ver}_1')$ *be the construction from Corollary 9 applied to* $(\mathsf{Samp}_0,\mathsf{Ver}_0)$ *and* $(\mathsf{Samp}_1,\mathsf{Ver}_1)$ *respectively. Define* $(\widetilde{\mathsf{Samp}},\widetilde{\mathsf{Ver}})$ *by*

1. $\widetilde{\mathsf{Samp}}(1^\lambda)$*: Run* $\mathsf{Samp}_0'\to(k_0,s_0)$ *and* $\mathsf{Samp}_1'\to(k_1,s_1)$*. Output* $(k=(k_0,k_1),s=(s_0,s_1))$*.*
2. $\widetilde{\mathsf{Ver}}((k_0,k_1),(s_0,s_1))$*: Output* 1 *if and only if* $\mathsf{Ver}_0'(k_0,s_0)=1$ *and* $\mathsf{Ver}_1'(k_1,s_1)=1$*.*

Then $(\widetilde{\mathsf{Samp}},\widetilde{\mathsf{Ver}})$ *is a* OWPuzz *as long as one of* $(\mathsf{Samp}_0,\mathsf{Ver}_0)$ *or* $(\mathsf{Samp}_1,\mathsf{Ver}_1)$ *is a* OWPuzz*.*

The same corollary holds for $\mathsf{EV}-\mathsf{OWPuzz}$ *when replacing Corollary 9 with Corollary 10.*

Remark 4. *Given any polynomial* p *different* OWPuzz *candidates* $\{(\mathsf{Samp}_i,\mathsf{Ver}_i)\}_{i\in[p]}$*, we can use the same idea to build a* OWPuzz $(\widetilde{\mathsf{Samp}},\widetilde{\mathsf{Ver}})$ *which is secure as long as one of the candidates is secure. This follows the same logic as above with* $\widetilde{\mathsf{Samp}}(1^\lambda)\to((k_0,...,k_p),(s_0,...,s_p))$ *and* Ver *only outputting* 1 *if all* p Ver_i'*'s output* 1*.*

8.3 Universal Construction

The universal construction of EV follows standard techniques.

Definition 8 (Universal $\mathsf{EV}-\mathsf{OWPuzz}$)**.** *A set of uniform algorithms* $(\mathsf{Samp},\mathsf{Ver})$ *is a universal construction of a* $\mathsf{EV}-\mathsf{OWPuzz}$ *if it is a* $\mathsf{EV}-\mathsf{OWPuzz}$ *so long as some* $\mathsf{EV}-\mathsf{OWPuzz}$ *exists.*

Theorem 25 (Universal $\mathsf{EV}-\mathsf{OWPuzz}$)**.** *Define* Samp_i *as the* i^{th} *quantum Turing machine and* Ver_j *as the* j^{th} *quantum Turing machine each of which with an attached alarm clock which will halt the machine after* λ^3 *steps.*

Define $(\mathsf{Samp}_U,\mathsf{Ver}_U)$ *using the construction from Remark 4 for the* λ^2 *candidates* $(\mathsf{Samp}_i,\mathsf{Ver}_j)$ *for all* $(i,j)\in[\lambda]^2$*. If any* $\mathsf{EV}-\mathsf{OWPuzz}$ $(\mathsf{Samp},\mathsf{Ver})$ *exists, then* $(\mathsf{Samp}_U,\mathsf{Ver}_U)$ *is a* OWPuzz*.*

Proof. Using standard padding arguments see [Lev87, HKNY23] we can show that if there exists a (α,β) $\mathsf{EV}-\mathsf{OWPuzz}$ with both Ver and Samp running in time less than λ^c, then there exists a $(\alpha^{(1/c)},\beta^{(1/c)})$ $\mathsf{EV}-\mathsf{OWPuzz}$ with Ver' and Samp' running in time less than

λ^3. If $\mathsf{EV}-\mathsf{OWPuzz}$'s exist then there exists a $(\mathsf{negl},\mathsf{negl})\mathsf{EV}-\mathsf{OWPuzz}$ and therefore a $(\mathsf{negl},\mathsf{negl})\mathsf{EV}-\mathsf{OWPuzz}$ running in time less than λ^3.

If some one-way puzzle exists then for some i,j, $(\mathsf{Samp}_i,\mathsf{Ver}_j)$ will be a one way puzzle with both algorithms running in time less than n^3. Once $\lambda \geq \max(i,j)$ a one way puzzle will one of the candidates. By Remark 4 we know that$(\mathsf{Samp}_U,\mathsf{Ver}_U)$ this will be a OWPuzz. □

Theorem 26 (Universal OWPuzz)**.** *Define* Samp_i *as the* i^{th} *quantum Turing machine with an attached alarm clock which will halt the machine after* λ^3 *steps. Define* Ver_i *to be the function such that the sum of the correctness and security error of* $(\mathsf{Samp}_i,\mathsf{Ver}_i)$ *is minimized.*

Define $(\mathsf{Samp}_U)(1^\lambda)$ *using the construction from Remark 4 for the* λ^2 *candidates* $(\mathsf{Samp}_i,\mathsf{Ver}_i)$ *for all* $(i,j)\in[\lambda]^2$*. If any* OWPuzz $(\mathsf{Samp},\mathsf{Ver})$ *exists, then* $(\mathsf{Samp}_U,\mathsf{Ver}_U)$ *is a* OWPuzz*.*

Proof. Observe that if there exists a OWPuzz $(\mathsf{Samp},\mathsf{Ver})$ then when i is such that $\mathsf{Samp}_i = \mathsf{Samp}$, $(\mathsf{Samp}_i,\mathsf{Ver}_i)$ is also a OWPuzz. The rest of the proof follows by the same argument as Theorem 25. □

9 OWPuzz security amplification

In this section we prove the following theorem:

Theorem 27 (Restatement of Theorem 22)**.** *If, for some* $c>0$*, there exists a* $(\mathsf{negl}(\lambda),1-\lambda^{-c})$ *one-way puzzle* $(\mathsf{Samp},\mathsf{Ver})$*, then there exists a strong one-way puzzle.*

We do this be showing that weak OWPuzz imply non-uniform EFID pairs, and then show that non-uniform EFID pairs imply strong OWPuzz. The first of these steps serves as a simpler and more general version of the argument presented in [KT24].

9.1 OWPuzzs imply non-uniform EFID

Definition 9 (From [KT24])**.** *An* ν^**-non-uniform EFID pair is a QPT algorithm* $G_\nu(1^\lambda,b)$ *with classical parameter-dependent advice* ν*. On input a unary security parameter* λ *and* $b\in\{0,1\}$ *outputs a classical string such that*

1. *For all QPT* $\mathcal{A}$ *and for all sufficiently large* λ*, the distributions* $G_{\nu^*}(1^\lambda,0)$ *and* $G_{\nu^*}(1^\lambda,1)$ *are indistinguishable.*
2. $$\Delta(G_{\nu^*}(1^\lambda,0),G_{\nu^*}(1^\lambda,1))\geq 1-\epsilon$$

Theorem 28 (Restatement of Corollary 13)**.** *If there exists a* OWPuzz*, then there exists a non-uniform EFID pair.*

Corollary 12 (Restatement of Corollary 15)**.** *If there exists a* OWPuzz*, then there exists an EFI pair.*

This argument will follow from adapting the techniques of [VZ12] and [HRV10]. We recall a number of technical lemmas from these papers, and observe that since these lemmas rely only on black-box techniques, they also hold in the post-quantum setting. While we will focus on the non-uniform setting, the results should also hold against uniform adversaries by adapting the arguments of [VZ12] and [HRV10] in the uniform setting.

We first recall the following definitions used in [VZ12, HRV10] (all of which operate in the non-uniform setting)

Definition 10. *Let X, B be two jointly sampled random variables. We say that B has (t, ϵ) (quantum) pseudoentropy at least k given X if there exists a random variable C jointly sampled with X such that*

1. *$H(C|X) \geq k$*
2. *(X, B) and (X, C) are (t, ϵ)-indistinguishable (by quantum circuits)*

When $t = 1/\mathsf{negl}$ and $\epsilon = \mathsf{negl}$ we can omit the t, ϵ and simply say that B has (quantum) pseudoentropy k.

Definition 11. *Let $B^{(i)}$ be a random variable over $[q]$ for each $i \in [m]$. We say that $B = (B^{(1)}, \ldots, B^{(m)})$ has next-block (quantum) pseudoentropy at least k if the random variable $B^{(I)}$ has (quantum) pseudoentropy at least k/m given $B^{(1)}, \ldots, B^{(I-1)}$, for $I \sim [m]$. If $q = 2$, then we will use the term next-bit pseudoentropy.*

Definition 12. *Let $B^{(i)}$ be a random variable for each $i \in [m]$. We say that every block of $B = (B^{(1)}, \ldots, B^{(m)})$ has next-block (quantum) pseudoentropy at least k if for all i, the random variable $B^{(i)}$ has (quantum) pseudoentropy at least k given $B^{(1)}, \ldots, B^{(i-1)}$.*

Definition 13. *For random variables A, B the KL divergence from A to B is defined as*

$$KL(A||B) = \mathop{\mathbb{E}}_{a \sim A}\left[\log \frac{\Pr[A \to a]}{\Pr[B \to a]}\right]$$

Definition 14. *Let X, B be jointly sampled random variables. We say that B is (t, δ) (quantum) KL-hard for sampling given X if for all size-t randomized (quantum) circuits S,*

$$KL(X, B||X, S(X)) > \delta$$

Remark 5. *We will also define pseudo-min-entropy to have the same definition as pseudoentropy, but with Shannon entropy replaced with min-entropy. We analogously define next-block pseudo-min-entropy.*

9.1.1 OWPuzz imply next-bit pseudoentropy

We state quantum versions of the corresponding lemmas from [VZ12]

Lemma 1 (Quantum version of Theorem 3.15 and Lemma 3.6 from [VZ12])**.** *Let (X, B) be jointly sampled random variables over $\{0,1\}^n \times [q]$. If B is quantum (t, δ) KL-hard for sampling given X, then for every $\epsilon > 0$, B has quantum (t', ϵ) psuedoentropy at least $H(B|X) + \delta - \epsilon$ given X, for $t' = t^{\Omega(1)}/\mathsf{poly}(n, q, 1/\epsilon)$.*

Proof. The proof is exactly the same as the proofs of Lemma 3.6 and Theorem 3.15 from [VZ12], but with the use of the Min-Max theorem replaced by its quantum equivalent (Theorem 4.1 from [CCL+17]). □

Lemma 2 (Quantum chain rule for KL-hardness (variant of Lemma 4.3 from [VZ12])**.** *Let Y be a distribution over $\{0,1\}^n$, jointly distributed with Z. If Y is quantum (t, δ) KL-hard for sampling given Z, then Y_I is quantum $(t', \delta/n)$ KL-hard for sampling given $(Z, Y_1, \ldots, Y_{I-1})$, for $I \sim [n]$, $t' = t/O(n)$.*

Proof. Same as Lemma 4.3 from [VZ12]. □

Theorem 29. *Let* $(\mathsf{Samp}, \mathsf{Ver})$ *be a* (ω, γ)-OWPuzz *secure against all time* t *adversaries with puzzle length* $m(\lambda)$*. Then for all* $\epsilon > 0$*,* $\mathsf{Samp} \to (k, s)$ *has quantum* $(t', \epsilon/m)$ *next-bit pseudoentropy at least*

$$H(k, s) + \delta - \epsilon$$

for $t' = t^{\Omega(1)}/\mathsf{poly}(\lambda, 1/\epsilon)$ *and*

$$\delta = (1 - \omega) \log \frac{1 - \omega}{\gamma} + \omega \log \frac{\omega}{1 - \gamma}$$

Proof. We first observe that k is (t, δ) KL-hard to sample given s. Let S be any time t quantum circuit. We will show that $KL(s, k||s, S(s)) \geq \delta$.

By monotonicity of quantum relative entropy,

$$KL(\mathsf{Ver}(s, k)||\mathsf{Ver}(s, S(s))) \leq KL(s, k||s, S(s))$$

But observe that $\mathsf{Ver}(s, k)$ is a Bernoulli random variable $Bern(p)$ for some $p > 1 - \omega$ by correctness. But $\mathsf{Ver}(s, S(s))$ is a Bernoulli random variable $Bern(p')$ for p' the advantage of S in the OWPuzz game. By security of the OWPuzz, $p' \leq \gamma$.

$$KL(\mathsf{Ver}(s, k)||\mathsf{Ver}(s, S(s))) \geq KL(Bern(1 - \omega), Bern(\gamma))$$

$$= (1 - \omega) \log \frac{1 - \omega}{\gamma} + \omega \log \frac{\omega}{1 - \gamma} = \delta.$$

Let $m = \mathsf{poly}(\lambda)$ be the length of k. By Lemma 2, for $I \sim [m]$, k_I is quantum $\left(\frac{t}{O(m)}, \frac{\delta}{m}\right)$ KL-hard for sampling given $(s, k_1, \ldots, k_{I-1})$.

And so by Lemma 1, for every $\epsilon > 0$, k_I has quantum $(t', \epsilon/m)$ pseudoentropy at least $H(k_I|s, k_1, \ldots, k_{I-1}) + \frac{\delta}{m} - \epsilon/m$ given s. But by definition, this means that k has $(t', \epsilon/m)$ next-bit pseudoentropy $m \cdot H(k_I|s, k_1, \ldots, k_{I-1}) + \delta - \epsilon$ given s. But $m \cdot H(k_I|s, k_1, \ldots, k_{I-1}) \geq H(k|s)$, so (k, s) has $(t', \epsilon/m)$ next-bit pseudoentropy $H(s) + H(k|s) + \delta - \epsilon = H(k, s) + \delta - \epsilon$ for $t' = t^{\Omega(1)}/\mathsf{poly}(\lambda, 1/\epsilon)$. □

9.1.2 Next-bit pseudoentropy implies non-uniform EFID

This argument follows the framework of [HRV10]. The argument is nearly identical to the classical case. The primary difference is that instead of lower bounding the length of the resultant string by the length of the input, we instead lower bound the length of the resultant string by the entropy of the input.

Lemma 3 (Lemma 5.2 from [HRV10])**.** *For* $i \in [m]$*,* $x^{(1)}, \ldots, x^{(\ell)} \in \mathcal{M}^m$*, we define*

$$Equalizer(i, x^{(1)}, \ldots, x^{(\ell)}) := x_i^{(1)}, \ldots, x_m^{(1)}, x^{(2)}, \ldots, x^{(\ell-1)}, x_1^{(\ell)}, \ldots, x_{i-1}^{(\ell)}$$

That is, $Equalizer(i, x^{(1)}, \ldots, x^{(\ell)})$ *truncates the first* $i - 1$ *blocks from the first input and the last* $m - (i - 1)$ *blocks from the last input.*

Let X *be a random variable over* $\mathcal{M}^m$ *with* (t, ϵ) *next block quantum pseudoentropy at least* k*. Let* $X^{(1)}, \ldots, X^{(\ell)}$ *be* ℓ *independent and identically distributed copies of* X*, and let* I *be uniformly distributed over* $[m]$*. Define* $\widetilde{X} = Equalizer(I, X^{(1)}, \ldots, X^{(\ell)})$*. Then every block of* $\widetilde{X}$ *has* $(t - O(\ell \cdot m \cdot \log |\mathcal{M}|), \ell \cdot \epsilon)$ *next-block quantum pseudoentropy at least* k/m*.*

Lemma 4 (Lemma 5.3 from [HRV10])**.** *Let* X *be a random variable over* $\mathcal{M}^m$ *where every block of* X *has* (t, ϵ) *next-block quantum pseudo-min-entropy at least* k*. Let* X^a *refer to* a *i.i.d. copies of* X*. For every* $\kappa > 0$*,* X^a *has* (t', ϵ') *next-block quantum pseudo-min-entropy* k' *where*

1. $t' = t - O(ma \log |\mathcal{M}|)$
2. $\epsilon' = a^2(\epsilon + 2^{-\kappa} + 2^{-ca})$ *for a universal constant* $c > 0$
3. $k' = ak - O(\log(a|\mathcal{M}|)\sqrt{a\kappa})$

Lemma 5 (Lemma 5.4 from [HRV10])**.** *There exists an efficient procedure* $\mathsf{Ext} \in NC^1$ *on input* $x \in (\{0,1\}^a)^m$ *and* $s \in \{0,1\}^a$ *which outputs a string of length* $y \in \{0,1\}^{a+m(k-k')}$ *such that the following holds: Let* X *be a random variable over* $(\{0,1\}^a)^m$ *such that every block of* X *has* (t,ϵ) *next-bit quantum pseudo-min-entropy* k*, then for all QPT* $\mathcal{A}$ *running in time* $t - m \cdot a^{O(1)}$

$$\left|\Pr[\mathcal{A}(\mathsf{Ext}(X,\mathcal{U}_a)) \to 1] - \Pr[\mathcal{A}(\mathcal{U}_{a+m(k-k')}) \to 1]\right| \leq m(\epsilon + 2^{-k'/2})$$

Proof. The reductions for all three of these lemmas from [HRV10] are fully black-box, and so also hold in the quantum setting. □

Lemma 6. *Let* X *be a random variable with* $|X| = m$*. If* $H(X) \leq m - \delta$ *for some* $\delta > 0$*, then*

$$SD(X,\mathcal{U}_m) \geq \frac{\delta}{2m-\delta} - 2^{-\delta/2}$$

In particular, if $m = O(p(\lambda))$ *and* $\delta = \Omega(p'(\lambda))$ *for some polynomials* p *and* p'*, then there exists a polynomial* q *such that*

$$SD(X,\mathcal{U}_m) = \Omega\left(\frac{1}{q(\lambda)}\right)$$

Proof of lemma 6.

$$S := \{x : \Pr[X \to x] > 2^{-m+\delta/2}\}$$

Observe that $1 \geq \sum_{x \in S} \Pr[X \to x] \geq |S| 2^{-m+\delta/2}$ and so $|S| \leq 2^{m-\delta/2}$. Thus,

$$\Pr[\mathcal{U} \in S] \leq 2^{m-\delta/2-m} = 2^{-\delta/2}$$

We have

$$H(X) = \mathop{\mathbb{E}}_{X \to x}[-\log \Pr[X \to x]] \leq m - \delta$$

By Markov bound, we get

$$\Pr_{X \to x}[-\log \Pr[X \to x] \geq m - \delta/2] \leq \frac{m-\delta}{m-\delta/2}$$

and so

$$\Pr[X \in S] = \Pr_{X \to x}[\Pr[X \to x] > 2^{-m+\delta/2}] \geq 1 - \frac{m-\delta}{m-\delta/2} = \frac{\delta}{2m-\delta}$$

Putting these two statements together, we have

$$SD(X,\mathcal{U}) \geq \Pr[X \in S] - \Pr[\mathcal{U} \in S] \geq \frac{\delta}{2m-\delta} - 2^{-\delta/2}$$

□

Theorem 30 (Adapted from Theorem 5.5 from [HRV10]). *Let $m(\lambda), \Delta(\lambda)$ be two computable functions such that $\Delta = \Delta(\lambda) \in [1/\mathsf{poly}(\lambda), \lambda]$. Let $\{X\}_{\lambda\in\mathbb{N}}$ be a family of efficiently samplable random variables of length $m(\lambda)$ with next-bit pseudoentropy at least $H(X) + \Delta$. Then there exists a function $\nu^*(\lambda) \leq |X|$ such that there exists a QPT algorithm $D_\nu(1^\lambda)$ outputting a classical string such that*

1. $D_{\nu^*}(1^\lambda) \approx \mathcal{U}$
2. $SD(D_{\nu^*}(1^\lambda), \mathcal{U}) \geq \frac{1}{p(\lambda)}$

for some efficiently computable polynomial p.

The proof of this theorem follows essentially the same lines as the proof from [HRV10], but with all references to the input length replaced by the entropy of X. This gives a distribution indistinguishable from uniform but with less than full entropy. Lemma 6 then gives a bound on the statistical distance. However, this bound is not $1 - \mathsf{negl}$. Fortunately, taking the product distribution amplifies statistical distance, so we simply take the direct product of the construction from [HRV10].

Proof. Without loss of generality, we will assume that the block-length of X is a power of 2 (otherwise, we can just append 0s). We have that for all $c > 0$ and sufficiently large λ, X has $(t = \lambda^c, \epsilon = \lambda^{-c})$ next-bit pseudoentropy $H(X) + \Delta$. We set $\ell := \lceil 2(\nu^* + \Delta + \log m)/\Delta \rceil = \Omega(\nu^*/\Delta)$ in Lemma 3 and get a new random variable $\widetilde{X} = Equalizer(I, X^{(1)}, \ldots, X^{(\ell)})$. We will define

$$D_\nu(1^\lambda, 1) := \mathsf{Ext}((\widetilde{X})^a, \mathcal{U}_a)$$

using the Ext from Lemma 5 for some value of $a = \mathsf{poly}(\lambda)$ and with output length $d_\nu := a + m(\ell - 1)(k'_\nu - \kappa)$ for $k'_\nu := ak_\nu - O(\log(a|\mathcal{M}|)\sqrt{a\kappa})$ and $k_\nu := (\nu + \Delta)/m$.

Observe that when $\nu = \nu^* = H(X)$, Lemma 3 shows that every bit of $\widetilde{X}$ has $(t - O(\ell m), \ell\epsilon)$ next-bit quantum pseudoentropy at least $k_{\nu^*} = (\nu^* + \Delta)/m$.

Next, Lemma 4 shows that every block of $(\widetilde{X})^a$ has $(t - O(m\ell a), a^2(\epsilon + 2^{-\kappa} + 2^{-\Omega(a)}))$ next-bit quantum pseudo-min-entropy at least $k'_{\nu^*} = ak_{\nu^*} - O(\sqrt{a\kappa}\log a)$.

Finally, Lemma 5 shows that for all QPT $\mathcal{A}$ running in time $t - O(m\ell a^{O(1)}) = t - \mathsf{poly}(\lambda)$,

$$\left|\Pr[\mathcal{A}(D_{\nu^*}(1^\lambda) \to 1] - \Pr[\mathcal{A}(\mathcal{U}) \to 1]\right| \leq m\ell(a^2(\epsilon + 2^{-\kappa} + 2^{-\Omega(a)}) + 2^{-\kappa/2})$$

$$\leq \mathsf{poly}(\lambda)(\epsilon + 2^{-\kappa/2} + 2^{-\Omega(a)})$$

Setting $\kappa = \lambda/2$ and using the fact that this holds for all $t = n^c, \epsilon = n^{-c}$, we get that $D_{\nu^*}(1^\lambda)$ and $\mathcal{U}_{d_\nu}$ are indistinguishable.

It just remains to be shown that

$$\Delta(D_{\nu^*}(1^\lambda), \mathcal{U}_{d_{\nu^*}}) \geq \frac{1}{\mathsf{poly}(\lambda)}$$

We will do this by showing that $H(D_{\nu^*}) \leq H(\mathcal{U}_{d_{\nu^*}}) - \Omega(\mathsf{poly}(\lambda)) = d_{\nu^*} - \Omega(\mathsf{poly}(\lambda))$ and then applying Lemma 6.

Observe that when $\nu^* = H(X)$, $H(\widetilde{X}) \leq \ell H(X) + \log m$. And so $H(\widetilde{X}^a) \leq a(\ell H(X) + \log m)$. It is clear to see that $H(D_{\nu^*}) \leq a(\ell H(X) + \log m) + a$. Let us denote this value by $d' := H(D_{\nu^*})$.

$$
\begin{aligned}
d_{\nu^*} &= a + m(\ell-1)(k'_{\nu^*} - \kappa) \\
&= a + m(\ell-1)(ak_{\nu^*} - O(\sqrt{a\kappa}\log a) - \kappa) \\
&= a + a(\ell H(X) + \log m) + a\ell\Delta - a(\nu^* + \Delta + \log m) - m(\ell-1)(O(\sqrt{a\kappa}\log a) + \kappa) \\
&\geq a + a(\ell H(X) + \log m) + a\ell\Delta/2 - m(\ell-1)(O(\sqrt{a\kappa}\log a) + \kappa) \\
&\geq d' + a\ell\Delta/2 - m(\ell-1)(O(\sqrt{a\kappa}\log a) + \kappa) \\
&\geq d' + a\ell\Delta/4 \\
&= d' + \Omega(a\nu^*) \\
&= d' + \Omega(\mathsf{poly}(\lambda))
\end{aligned}
$$

when

$$
a = \Theta\left(\left(\left(\frac{m(\ell-1)}{\Delta\ell}\right)^2 \kappa \log^2\left(\frac{m(\ell-1)\kappa}{\Delta\ell}\right)\right)\right) = \Theta\left(\frac{m^2\kappa\log^2\lambda}{\Delta^2}\right)
$$

□

Lemma 7 (Amplification of statistical distance.)**.** *Let $SD(X,Y) \geq \delta$. Then if $q \geq \frac{12t}{\delta^2}$ $SD(X^q, Y^q) \geq 1 - 2e^{-t}$.*

Proof of lemma 7. Since $SD(X,Y) \geq \delta$, we know there exists a set S such that

$$
\Pr[X \in S] - \Pr[Y \in S] \geq \delta
$$

Define $\alpha \coloneqq \Pr[X \in S]$ and $\beta \coloneqq \Pr[Y \in S]$. We will define a new set

$$
S' \coloneqq \{(x_1, \ldots, x_q) : \text{ at least an } \frac{\alpha+\beta}{2} \text{ fraction of } x_i \in S\}.
$$

By the Chernoff bound,

$$
\begin{aligned}
\Pr[X^q \notin S'] &\leq e^{-\left(1-\frac{\alpha+\beta}{2\alpha}\right)^2 \alpha q/2} \\
&\leq e^{-q\frac{(\alpha-\beta)^2}{8\alpha}} \\
&\leq e^{-q\frac{\delta^2}{8}}
\end{aligned}
$$

Similarly,

$$
\begin{aligned}
\Pr[Y^q \in S] &\leq e^{-\left(\frac{\alpha-\beta}{2\beta}\right)^2 \beta q/3} \\
&= e^{-q\frac{(\alpha-\beta)^2}{12\beta}} \\
&\leq e^{-q\frac{\delta^2}{12}}
\end{aligned}
$$

So if $q \geq \frac{12t}{\delta^2}$,

$$
SD(X^q, Y^q) \geq 1 - 2e^{-q\frac{\delta^2}{12}} = 1 - 2e^{-t}
$$

□

Corollary 13. *Let $\Delta = \Delta(n) \in [1/\mathsf{poly}(n), n]$ and let $\{X\}_{n\in\mathbb{N}}$ be a family of efficiently samplable random variables of length m with next-bit pseudoentropy at least $H(X) + \Delta$. Then there exists a function $\nu^*(\lambda) \leq |X|$ such that there exists a non-uniform ν^*-EFID.*

Proof. Define $G_\nu(1^\lambda, 0)$ to be uniform over $q(\lambda) \cdot d_s$ bits for some $q = \mathsf{poly}(\lambda)$ to be set later. Define $G_\nu(1^\lambda, 1) = D_{\nu^*}(1^\lambda)^{q(\lambda)}$.

Since $D_{\nu^*}(1^\lambda) \approx \mathcal{U}_{d_\nu}$, $D_{\nu^*}(1^\lambda)^{q(\lambda)} \approx \mathcal{U}_{q(\lambda)\cdot d_\nu}$.

Since $SD(D_{\nu^*}(1^\lambda), \mathcal{U}_{d_\nu}) \geq \frac{1}{p(\lambda)}$, if we define $q(\lambda) = 12\lambda p(\lambda)$, then by Lemma 7

$$SD(G_\nu(1^\lambda, 0), G_\nu(1^\lambda, 1)) \geq 1 - 2e^{-\lambda} = 1 - \mathsf{negl}(\lambda)$$

□

Corollary 14. *If, for some $c > 0$, there exists a $(\mathsf{negl}(\lambda), 1 - \lambda^{-c})$ one-way puzzle $(\mathsf{Samp}, \mathsf{Ver})$, then there exists an ν^*-non-uniform EFID pair with $\nu^* \leq m + n$.*

Proof. Let $\omega = \mathsf{negl}(\lambda)$ and $\gamma = 1 - \lambda^{-c}$ be the correctness and security parameters of $(\mathsf{Samp}, \mathsf{Ver})$, and let $n(\lambda)$ be the length of the key and let $m(\lambda)$ be the length of the puzzle. For all $t = \mathsf{poly}(\lambda)$, $\epsilon > 0$, by Theorem 29, $\mathsf{Samp} \to (k, s)$ has $(t^{\Omega(1)}/\mathsf{poly}(\lambda, 1/\epsilon), \epsilon/m)$ quantum next-bit pseudoentropy at least $H(k, s) + \delta - \epsilon$ for

$$\delta = (1 - \omega) \log \frac{1 - \omega}{\gamma} + \omega \log \frac{\omega}{1 - \gamma}.$$

But observe,

$$\begin{aligned} &(1 - \omega) \log \frac{1 - \omega}{\gamma} + \omega \log \frac{\omega}{1 - \gamma} \\ \geq\ &\frac{1}{2} \log \frac{1}{1 - \lambda^{-c}} + (1 - \omega) \log(1 - \omega) + \omega \log \omega + \omega \log \lambda^c \\ \geq\ &\frac{1}{2} \log \frac{1}{1 - \lambda^{-c}} - \mathsf{negl}(\lambda) \geq \lambda^{-(c+1)} \end{aligned}$$

for all sufficiently large λ.

Thus, for all sufficiently large d such that $n^{-d} \cdot m \leq \lambda^{-(c+1)}/2$, for all $t = \mathsf{poly}(\lambda)$, $\mathsf{Samp} \to (k, s)$ has $(t^{\Omega(1)}/\mathsf{poly}(\lambda, \lambda^d), n^{-d})$ quantum next-bit pseudoentropy at least $H(k, s) + \frac{1}{2}\lambda^{-(c+1)}$. Adjusting the value of t shows us that $\mathsf{Samp} \to (k, s)$ has quantum next-bit pseudoentropy at least $H(k, s) + \frac{1}{2}\lambda^{-(c+1)}$.

By Corollary 13 there exists a ν^*-EFID for $\nu^* \leq m + n$. □

Corollary 15. *If, for some $c > 0$, there exists a $(\mathsf{negl}(\lambda), 1 - \lambda^{-c})$ one-way puzzle $(\mathsf{Samp}, \mathsf{Ver})$, then there exists an EFI pair.*

Proof. Prior work shows that there exists a quantum combiner for quantum bit commitments [HKNY23], which are equivalent to EFI pairs [BCQ22]. We observe that this combiner (when composed with the construction of EFI pairs from commitments) operates separately on each security parameter. Thus, given a non-uniform EFID pair, we can apply the combiner from [HKNY23] to the non-uniform construction instantiated with each possible value of the advice. This process maintains security, but produces a quantum output. Thus, this process takes a non-uniform EFID pair and produces an EFI pair.

Since Corollary 14 shows that weak one-way puzzles can be used to build a non-uniform EFID pair, composing that construction with this approach produces an EFI pair from any weak one-way puzzle. □

9.2 QEFID imply OWPuzz

Definition 15 (ν^*-non-uniform OWPuzz)**.** *Let $\nu^*(\lambda)$ be some function. A ν^*-non-uniform one way puzzle (*OWPuzz*) is a pair of a sampling algorithm and a verification function $(\mathsf{Samp}_\nu, \mathsf{Ver}_\nu)$ taking in advice ν with the following syntax:*

1. $\mathsf{Samp}_\nu(1^\lambda) \to (k, s)$ *is a uniform QPT algorithm which outputs a pair of classical strings (k, s). We refer to s as the puzzle and k as the key. Without loss of generality, we can assume $k \in \{0,1\}^\lambda$.*

2. $\mathsf{Ver}_\nu(k, s) \to b$ *is a function which takes in a key and puzzle and outputs a bit $b \in \{0, 1\}$.*

satisfying the following properties:

1. *Correctness: Outputs of the sampler pass verification with overwhelming probability*

$$\Pr_{\mathsf{Samp}_{\nu^*}(1^\lambda)\to(k,s)}[\mathsf{Ver}_{\nu^*}(k, s) \to 1] \geq 1 - \alpha$$

2. *Security: Given a puzzle s, it is computationally infeasible to find a key s which verifies. That is, for all non-uniform QPT algorithms $\mathcal{A}$,*

$$\Pr_{\mathsf{Samp}_{\nu^*}(1^\lambda)\to(k,s)}[\mathsf{Ver}_{\nu^*}(\mathcal{A}(s), s) \to 1] \leq \beta$$

That is, a non-uniform one-way puzzle is a one-way puzzle for which correctness and security are only guaranteed to hold when given the correct advice.

Lemma 8 (From QEFID to OWPuzz)**.** *If there exists QEFID pair G, then there exists a* OWPuzz $(\mathsf{Samp}, \mathsf{Ver})$*.*

Proof of lemma 8. We will define the OWPuzz as follows

1. $\mathsf{Samp}(1^\lambda)$: Sample k uniformly at random from $\{0,1\}^\lambda$. For each $i \in [\lambda]$ sample s_i from $G(1^\lambda, k_i)$. Output $(k, (s_1, \ldots, s_\lambda))$.

2. $\mathsf{Ver}(1^\lambda, k, s)$: Set

$$T^* = \text{argmax}_{T:\{0,1\}^m\to\{0,1\}} \Big(\Pr[T(G(1^\lambda, 1)) \to 1] - \Pr[T(G(1^\lambda, 0)) \to 1\Big)$$

   For each $i \in [\lambda]$, check if $T^*(s_i) = k_i$. If all tests pass, output 1. Otherwise, output 0.

To prove correctness, let us observe that since $SD(G(1^\lambda, 0), G(1^\lambda, 1)) \geq 1 - \mathsf{negl}(\lambda)$, there exists a T^* such that

$$\Big(\Pr[T^*(G(1^\lambda, 1)) \to 1] - \Pr[T^*(G(1^\lambda, 0)) \to 1\Big) \geq 1 - \mathsf{negl}(\lambda).$$

In particular, for any such T^*, we have $\Pr[T^*(G(1^\lambda, 1) \to 1] \geq 1 - \mathsf{negl}(\lambda)$. Thus, $\Pr[\mathsf{Ver}(1^\lambda, \mathsf{Samp}(1^\lambda)) \to 1] \geq (1 - \mathsf{negl}(\lambda))^\lambda \geq 1 - \mathsf{negl}(\lambda)$.

To prove security, we observe that the QEFID game is a three-round quantum interactive protocol. Thus, we can apply quantum amplification (Theorem 4.1 from [BQSY23]) to see that for all non-uniform QPT $\mathcal{A}$,

$$\begin{aligned}
&\Pr_{\mathsf{Samp}(1^\lambda)\to(k,(s_1,\ldots,s_\lambda))}[\mathcal{A}(s_1,\ldots,s_\lambda)=k] \\
&\leq \left(\Pr_{\{0,1\}\to b, G(1^\lambda,b)\to s}[\mathcal{A}(s)\to b]\right)^\lambda \\
&\leq \left(\frac{1}{2}+\mathsf{negl}(\lambda)\right)^\lambda \\
&\leq \mathsf{negl}(\lambda).
\end{aligned}$$

To conclude, we observe that verification simply checks whether k is equal to the output of T^* on all of $(s_1,\ldots,s_\lambda)$. Thus, the only way to invert the one-way puzzle is to output the key used for generation. Formally,

$$\begin{aligned}
&\Pr_{\mathsf{Samp}(1^\lambda)\to(k,(s_1,\ldots,s_\lambda))}[\mathsf{Ver}(\mathcal{A}(s_1,\ldots,s_\lambda),(s_1,\ldots,s_\lambda))] \\
&\leq \Pr_{\mathsf{Samp}(1^\lambda)\to(k,(s_1,\ldots,s_\lambda))}[\mathcal{A}(s_1,\ldots,s_\lambda)=k] \\
&+\Pr_{\mathsf{Samp}\to(k,(s_1,\ldots,s_\lambda))}[\mathsf{Ver}(k',s)\to 1|\mathcal{A}(s_1,\ldots,s_\lambda)\to k'\neq k] \\
&\leq \mathsf{negl}(\lambda)+\Pr_{\mathsf{Samp}\to(k,(s_1,\ldots,s_\lambda))}[\text{ there exists an index } i \text{ such that } T^*(s_i)\neq k_i] \\
&\leq \mathsf{negl}(\lambda)+\lambda\Pr_{\{0,1\}\to b,G(b)\to s}[T^*(s)\neq b] \\
&\leq \mathsf{negl}(\lambda)+\lambda\mathsf{negl}(\lambda)=\mathsf{negl}(\lambda)
\end{aligned}$$

□

We observe that the same argument also works relative to an advice string, and so we have

Lemma 9 (From non-uniform QEFID to non-uniform OWPuzz)**.** *Let $\nu^*(\lambda)$ be some function. If there exists a ν^*-non-uniform QEFID pair G_ν, then there exists a ν^*-non-uniform* OWPuzz $(\mathsf{Samp}_\nu,\mathsf{Ver}_\nu)$*.*

Lemma 10 (From non-uniform OWPuzz to OWPuzz)**.** *Let $p(\lambda)$ be some computable polynomial. Let $\nu^*(\lambda)$ be some function satisfying $\nu^*(\lambda)\leq p(\lambda)$. If there exists a ν^*-non-uniform* OWPuzz $(\mathsf{Samp}_\nu,\mathsf{Ver}_\nu)$*, then there exists a* OWPuzz $(\widetilde{\mathsf{Samp}},\widetilde{\mathsf{Ver}})$*.*

Proof. We simply apply the construction from Remark 4 to $(\mathsf{Samp}_1,\mathsf{Ver}_1),\ldots,(\mathsf{Samp}_p,\mathsf{Ver}_p)$. An analogous argument to the proof that this construction is a combiner will give that the resulting $(\widetilde{\mathsf{Samp}},\widetilde{\mathsf{Ver}})$ is a OWPuzz. □

We now have all the pieces to show theorem 22 that security for OWPuzz can be amplified. First Corollary 14 gives us that weak OWPuzz imply non-uniform QEFID, Then the two lemmas above give us that non-uniform QEFID imply non-uniform OWPuzz which in turn imply strong OWPuzz.

10 Random Input OWPuzz $\leftrightarrow$ OWPuzz

In this section we answer an open question left by [KT24] of whether one-way puzzles imply random input one-way puzzles.

Definition 16. *A random input one-way puzzle is a pair of a sampling algorithm and a verification function* $(\mathsf{PuzzSamp}, \mathsf{Ver})$ *with the following syntax:*

1. $\mathsf{PuzzSamp}(1^\lambda, k) \to s$ *is a uniform QPT algorithm which takes in a key* k *and outputs a puzzle* s*.*
2. $\mathsf{Ver}(1^\lambda, k, s) \to b$ *is a function which takes in a key and a puzzle and outputs a bit* $b \in \{0,1\}$

satisfying the following properties:

1. *Correctness: Outputs of the sampler pass verification with overwhelming probability*
$$\Pr_{\{0,1\}^\lambda \to k, \mathsf{PuzzSamp}(1^\lambda,k)\to s}[\mathsf{Ver}(k,s) \to 1] \geq 1 - \mathsf{negl}(\lambda)$$
2. *Security: Given a puzzle* s*, it is computationally infeasible to find a key* s *which verifies. That is, for all non-uniform QPT algorithms* $\mathcal{A}$*,*
$$\Pr_{\{0,1\}^\lambda \to k, \mathsf{PuzzSamp}(1^\lambda,k)\to s}[\mathsf{Ver}(\mathcal{A}(s),s) \to 1] \leq \mathsf{negl}(\lambda)$$

Theorem 31. *There exists a one-way puzzle* $(\mathsf{Samp}, \mathsf{Ver})$ *if and only if there exists a random input one-way puzzle* $(\mathsf{PuzzSamp}', \mathsf{Ver}')$*. If* Ver *is efficient, then so is* Ver'*.*

Proof. Any random input one-way puzzle gives a one-way puzzle by having the sampler just sample the random key itself. Thus, we focus on building a random input one-way puzzle from any one-way puzzle.

We will define the random input one-way puzzle as follows

1. $\mathsf{PuzzSamp}'(1^\lambda, k')$: Run $\mathsf{Samp} \to (k, s)$. Output $s' = (k \oplus k', s)$.
2. $\mathsf{Ver}'(k', s')$: Parse s' as (a, b). Output $\mathsf{Ver}(a \oplus k', b)$.

Observe that $\mathsf{Ver}(k', s') = \mathsf{Ver}(k' \oplus (k \oplus k'), s) = \mathsf{Ver}(k, s)$ and so correctness follows from correctness of $(\mathsf{Samp}, \mathsf{Ver})$.

We will show security using a reduction. Let $\mathcal{A}$ be any adversary such that
$$\Pr_{\{0,1\}^\lambda \to k', \mathsf{PuzzSamp}(k')\to s'}[\mathsf{Ver}'(\mathcal{A}(s'),s') \to 1] \geq \epsilon$$

We will define $\mathcal{A}'$ as follows. On input s, sample r uniformly at random. Output $\mathcal{A}(r, s) \oplus r$.

Note that (r, s) is identically distributed to the output distribution of $\mathsf{PuzzSamp}$ on random inputs. Thus,
$$\begin{aligned}
&\Pr_{\mathsf{Samp}\to(k,s)}[\mathsf{Ver}(\mathcal{A}'(s),s) \to 1] \\
=\ &\Pr_{\{0,1\}^\lambda \to k', \mathsf{PuzzSamp}(k')\to(r,s)}[\mathsf{Ver}(\mathcal{A}(r,s)\oplus r,s)] \\
=\ &\Pr_{\{0,1\}^\lambda \to k', \mathsf{PuzzSamp}(k')\to s'}[\mathsf{Ver}'(\mathcal{A}(s'),s') \to 1] \geq \epsilon
\end{aligned}$$

and so $\epsilon \leq \mathsf{negl}(\lambda)$.

Note that if Ver was originally efficient, so is Ver', and so this same construction works for efficiently verifiable one-way puzzles, producing a random input efficiently verifiable one-way puzzle. □

The philosophical message behind this theorem is that it doesn't matter whether or not the key and puzzle are sampled together, the fundamental difference between one-way puzzles and one-way functions is that the puzzle is sampled using a quantum algorithm instead of classical randomness.

11 Acknowledgments

We thank Yanyi Liu for insightful discussion. E. Goldin was supported by a National Science Foundation Graduate Research Fellowship.

References

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

[ACC+22] Per Austrin, Hao Chung, Kai-Min Chung, Shiuan Fu, Yao-Ting Lin, and Mohammad Mahmoody. On the impossibility of key agreements from quantum random oracles. In Yevgeniy Dodis and Thomas Shrimpton, editors, *Advances in Cryptology – CRYPTO 2022*, pages 165–194, Cham, 2022. Springer Nature Switzerland.

[ALY23a] Prabhanjan Ananth, Yao-Ting Lin, and Henry Yuen. Pseudorandom strings from pseudorandom quantum states. *arXiv preprint arXiv:2306.05613*, 2023.

[ALY23b] Prabhanjan Ananth, Yao-Ting Lin, and Henry Yuen. Pseudorandom strings from pseudorandom quantum states. Cryptology ePrint Archive, Paper 2023/904, 2023. https://eprint.iacr.org/2023/904.

[AQY22] Prabhanjan Ananth, Luowen Qian, and Henry Yuen. Cryptography from pseudorandom quantum states. In Yevgeniy Dodis and Thomas Shrimpton, editors, *Advances in Cryptology – CRYPTO 2022*, pages 208–236, Cham, 2022. Springer Nature Switzerland.

[BCQ22] Zvika Brakerski, Ran Canetti, and Luowen Qian. On the computational hardness needed for quantum cryptography. Cryptology ePrint Archive, Paper 2022/1181, 2022. https://eprint.iacr.org/2022/1181.

[BQSY23] John Bostanci, Luowen Qian, Nicholas Spooner, and Henry Yuen. An efficient quantum parallel repetition theorem and applications, 2023.

[CCL+17] Yi-Hsiu Chen, Kai-Min Chung, Ching-Yi Lai, Salil P. Vadhan, and Xiaodi Wu. Computational notions of quantum min-entropy, 2017.

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

[CLM23] Kai-Min Chung, Yao-Ting Lin, and Mohammad Mahmoody. Black-box separations for non-interactive classical commitments in a quantum world. In Carmit Hazay and Martijn Stam, editors, *Advances in Cryptology – EUROCRYPT 2023*, pages 144–172, Cham, 2023. Springer Nature Switzerland.

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *J. ACM*, 33(4):792–807, aug 1986.

[GL89] O. Goldreich and L. A. Levin. A hard-core predicate for all one-way functions. In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, STOC '89, page 25–32, New York, NY, USA, 1989. Association for Computing Machinery.

[GMR87] Shafi Goldwasser, Silvio Micali, and Ronald L. Rivest. A digital signature scheme secure against adaptive chosen message attack**this research was supported by nsf grant mcs-80-06938, an ibm/mit faculty development award, and darpa contract n00014-85-k-0125.: Extended abstract. In David S. Johnson, Takao Nishizeki, Akihiro Nozaki, and Herbert S. Wilf, editors, *Discrete Algorithms and Complexity*, pages 287–310. Academic Press, 1987.

[Gol90] Oded Goldreich. A note on computational indistinguishability. *Information Processing Letters*, 34(6):277–281, 1990.

[HILL99] Johan HÅstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.

[HKN+05] Danny Harnik, Joe Kilian, Moni Naor, Omer Reingold, and Alon Rosen. On robust combiners for oblivious transfer and other primitives. In Ronald Cramer, editor, *Advances in Cryptology – EUROCRYPT 2005*, pages 96–113, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[HKNY23] Taiga Hiroka, Fuyuki Kitagawa, Ryo Nishimaki, and Takashi Yamakawa. Robust combiners and universal constructions for quantum cryptography. Cryptology ePrint Archive, Paper 2023/1772, 2023. https://eprint.iacr.org/2023/1772.

[HMY23] Minki Hhan, Tomoyuki Morimae, and Takashi Yamakawa. From the hardness of detecting superpositions to cryptography: Quantum public key encryption and commitments, 2023.

[HRV10] Iftach Haitner, Omer Reingold, and Salil Vadhan. Efficiency improvements in constructing pseudorandom generators from one-way functions. In *Proceedings of the Forty-Second ACM Symposium on Theory of Computing*, STOC '10, page 437–446, New York, NY, USA, 2010. Association for Computing Machinery.

[IL89] R. Impagliazzo and M. Luby. One-way functions are essential for complexity based cryptography. In *30th Annual Symposium on Foundations of Computer Science*, pages 230–235, 1989.

[Imp95] R. Impagliazzo. A personal view of average-case complexity. In *Proceedings of Structure in Complexity Theory. Tenth Annual IEEE Conference*, pages 134–147, 1995.

[KNY23] Fuyuki Kitagawa, Ryo Nishimaki, and Takashi Yamakawa. Publicly verifiable deletion from minimal assumptions. Cryptology ePrint Archive, Paper 2023/538, 2023. https://eprint.iacr.org/2023/538.

[Kre21] William Kretschmer. Quantum pseudorandomness and classical complexity. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

[KT24] Dakshita Khurana and Kabir Tomer. Commitments from quantum one-wayness, 2024.

[Lam79] Leslie Lamport. Constructing digital signatures from a one way function. Technical Report CSL-98, October 1979. This paper was published by IEEE in the Proceedings of HICSS-43 in January, 2010.

[Lev87] Leonid A. Levin. One way functions and pseudorandom generators. *Combinatorica*, 7(4):357–363, Dec 1987.

[MP23] Noam Mazor and Rafael Pass. Counting unpredictable bits: A simple prg from one-way functions. Cryptology ePrint Archive, Paper 2023/1451, 2023. https://eprint.iacr.org/2023/1451.

[MY22] Tomoyuki Morimae and Takashi Yamakawa. Quantum commitments and signatures without one-way functions. In *Annual International Cryptology Conference*, pages 269–295. Springer, 2022.

[Nao91] Moni Naor. Bit commitment using pseudorandomness. *Journal of Cryptology*, 4(2):151–158, Jan 1991.

[VZ12] Salil Vadhan and Colin Jia Zheng. Characterizing pseudoentropy and simplifying pseudorandom generator constructions. In *Proceedings of the 44th Annual ACM Symposium on Theory of Computing (STOC '12)*, pages 817–836. ACM, ACM, 2012.

[Yao82] Andrew C. Yao. Theory and application of trapdoor functions. In *23rd Annual Symposium on Foundations of Computer Science (sfcs 1982)*, pages 80–91, 1982.