Quantum-access-secure message authentication via blind-unforgeability

Gorjan Alagic[1], Christian Majenz[2], Alexander Russell[3], and Fang Song[4]

[1]QuICS, University of Maryland, and NIST, Gaithersburg, Maryland

[2]QuSoft and Centrum Wiskunde & Informatica, Amsterdam

[3]Department of Computer Science and Engineering, University of Connecticut

[4]Department of Computer Science, Portland State University

April 20, 2023

Abstract

Formulating and designing authentication of classical messages in the presence of adversaries with quantum query access has been a longstanding challenge, as the familiar classical notions of unforgeability do not directly translate into meaningful notions in the quantum setting. A particular difficulty is how to fairly capture the notion of "predicting an unqueried value" when the adversary can query in quantum superposition.

We propose a natural definition of unforgeability against quantum adversaries called *blind unforgeability*. This notion defines a function to be predictable if there exists an adversary who can use "partially blinded" oracle access to predict values in the blinded region. We support the proposal with a number of technical results. We begin by establishing that the notion coincides with EUF-CMA in the classical setting and go on to demonstrate that the notion is satisfied by a number of simple guiding examples, such as random functions and quantum-query-secure pseudorandom functions. We then show the suitability of blind unforgeability for supporting canonical constructions and reductions. We prove that the "hash-and-MAC" paradigm and the Lamport one-time digital signature scheme are indeed unforgeable according to the definition. To support our analysis, we additionally define and study a new variety of quantum-secure hash functions called *Bernoulli-preserving*.

Finally, we demonstrate that blind unforgeability is stronger than a previous definition of Boneh and Zhandry [EUROCRYPT '13, CRYPTO '13] in the sense that we can construct an explicit function family which is forgeable by an attack that is recognized by blind-unforgeability, yet satisfies the definition by Boneh and Zhandry.

Note: An earlier version of this article contained a theorem that the new security notion "blind-unforgeability" (BU) we introduce implies the notion of "plus-one" unforgeability (PO) that was hitherto the only proposed generalization of EUF-CMA to > 1 quantum chosen-message queries. Unfortunately the proof contained an error. We thank Shih-Han Hung for discovering the error. We have now removed the claim and the question whether the implication holds is currently open. We would like to emphasize that the example presented in Section 8 disqualifies PO as a quantum-access generalization of EUF-CMA, so currently there are two reasonable possible definitions of "quantum-access EUF-CMA": the notion of blind-unforgeability proposed in this article, and its conjunction with plus-one unforgeability.

1 Introduction

Large-scale quantum computers will break widely-deployed public-key cryptography, and may even threaten certain post-quantum candidates [21, 8, 9, 10, 5]. Even elementary symmetric-key constructions like Feistel ciphers and CBC-MACs become vulnerable in quantum attack models where the adversary is presumed to have quantum query access to some part of the cryptosystem [15, 16, 14, 20]. As an example, consider encryption in the setting where the adversary has access to the unitary operator $|x\rangle|y\rangle \mapsto |x\rangle|y \oplus f_k(x)\rangle$, where f_k is the encryption or decryption function with secret key k. While it is debatable if this model reflects physical implementations of symmetric-key cryptography, it appears necessary in a number of generic settings, such as public-key encryption and hashing with public hash functions. It could also be relevant when private-key primitives are composed in larger protocols, e.g., by exposing circuits via obfuscation [19]. Setting down appropriate security definitions in this quantum attack model is the subject of several threads of recent research [7, 11].

In this article, we study authentication of classical information in the quantum-secure model. Here, the adversary is granted quantum query access to the signing algorithm of a message authentication code (MAC) or a digital signature scheme, and is tasked with producing valid forgeries. In the purely classical setting, we insist that the forgeries are fresh, i.e., distinct from previous queries to the oracle. When the function may be queried in superposition, however, it's unclear how to meaningfully reflect this constraint that a forgery was previously "unqueried." For example, it is clear that an adversary that simply queries with a uniform superposition and then measures a forgery—a feasible attack against any function—should not be considered successful. On the other hand, an adversary that uses the same query to discover some structural property (e.g., a superpolynomial-size period in the MAC) should be considered a break. Examples like these indicate the difficulty of the problem. How do we correctly "price" the queries? How do we decide if a forgery is fresh? Furthermore, how can this be done in a manner that is consistent with these guiding examples? In fact, this problem has a natural interpretation that goes well beyond cryptography: *What does it mean for a classical function to appear unpredictable to a quantum oracle algorithm?*[1]

1.1 Previous approaches

The first approach to this problem was suggested by Boneh and Zhandry [6]. They define a MAC to be unforgeable if, after making q queries to the MAC, no adversary can produce $q + 1$ valid input-output pairs except with negligible probability. We will refer to this notion as "PO security" (PO for "plus one," and k-PO when the adversary is permitted a maximum of k queries). Among a number of results, Boneh and Zhandry prove that this notion can be realized by a quantum-secure pseudorandom function (qPRF).

Another approach, due to Garg, Yuen and Zhandry [12] (GYZ), considers a function *one-time* unforgeable if only a trivial "query, measure in computational basis, output result" attack[2] is allowed. Unfortunately, it is not clear how to extend GYZ to two or more queries. Furthermore, the single query is allowed in a limited query model with an non-standard restriction.[3] Zhandry recently showed a separation between PO and GYZ by means of the powerful tool of obfuscation [29].

It is interesting to note that similar problems arise in encryption schemes of *quantum* data and a convincing solution was recently found [3, 2]. However, it relies on the fact that for quantum messages, *authentication implies secrecy*. This enables "tricking" the adversary by replacing their queries with "trap" plaintexts to detect replays. As unforgeability and secrecy are orthogonal in the classical world, adversaries would easily recognize the spoofed oracle. This renders the approach of [3, 2] inapplicable in this case.

[1] The related notion of "appearing *random* to quantum oracle algorithms" has a satisfying definition, which can be fulfilled efficiently [27].

[2] Technically, the *Stinespring dilation* [23] of a computational basis measurement is the most general attack.

[3] Compared to the standard quantum oracle for a classical function, GYZ require the output register to be empty prior to the query.

1.2 Unresolved issues

PO security, the only candidate definition of quantum-secure unforgeability in the general, multi-query setting, appears to be insufficient for several reasons. First, as observed in [12], it is a priori unclear if PO security rules out forging on a message region A while making queries to a signing oracle supported on a disjoint message region B. Second, there may be unique features of quantum information, such as the destructiveness of quantum measurement, which PO does not capture. In particular, quantum algorithms must sometimes "consume" (i.e., fully measure) a state to extract some useful information, such as a symmetry in the oracle. There might be an adversary that makes one or more quantum queries but then must consume the post-query states completely in order to make a single, but convincing, forgery.

Surprisingly, prior to this work none of these plausible attack strategies have been exploited to give a separation between PO and "intuitive security."

2 Summary of results

2.1 A new definition: Blind-unforgeability

To address the abovementioned issues, and in light of the concrete "counterexample" presented below as Construction 1, we develop a new definition of many-time unforgeability we call "blind-unforgeability" (or BU). In this approach we examine the behavior of adversaries in the following experiment. The adversary is granted quantum oracle access to the MAC, "blinded" at a random region B. Specifically, we set B to be a random ϵ-fraction of the message space, and declare that the oracle function will output $\perp$ on all of B.

$$B_\epsilon \mathsf{Mac}_k(x) := \begin{cases} \perp & \text{if } x \in B_\epsilon, \\ \mathsf{Mac}_k(x) & \text{otherwise.} \end{cases}$$

Given a MAC $(\mathsf{Mac}, \mathsf{Ver})$, an adversary $\mathcal{A}$, and $\mathcal{A}$-selected parameter ϵ, the "blind forgery experiment" is:

1. Generate key k and random blinding B_ϵ;
2. Produce candidate forgery $(m, t) \leftarrow \mathcal{A}^{B_\epsilon \mathsf{Mac}_k}(1^n)$.
3. Output win if $\mathsf{Ver}_k(m, t) = \mathsf{acc}$ and $m \in B_\epsilon$; otherwise output rej.

Definition 1. *A MAC is blind-unforgeable* (BU) *if for every adversary* $(\mathcal{A}, \epsilon)$*, the probability of winning the blind forgery experiment is negligible.*

In this work, BU will typically refer to the case where $\mathcal{A}$ is an efficient quantum algorithm (QPT) and the oracle is quantum, i.e., $|x\rangle|y\rangle \mapsto |x\rangle|y \oplus B_\epsilon \mathsf{Mac}_k(x)\rangle$. We will also consider q-BU, the information-theoretic variant where the total number of queries is a priori fixed to q. We remark that the above definition is also easy to adapt to other settings, e.g., classical security against PPT adversaries, quantum or classical security for digital signatures, etc.

We remark that one could define a variant of the above where the adversary is allowed to describe the blinding distribution, rather than it being uniform. However, this is not a stronger notion. By a straightforward argument, an adversary wins in the chosen-blinding BU game if and only if it wins with a uniform ϵ-blinding for inverse-polynomial ϵ. Indeed, the adversary can just simulate its chosen blinding herself, and this still succeeds with inverse polynomial probability when interacting with a standard-blinded oracle (see Theorem 1 below).

2.2 Results about blind-unforgeability

To solidify our confidence in the new notion, we collect a series of results which we believe establish BU as a definition of unforgeability that captures the desired intuitive security requirement. In particular, we show that BU classifies a host of representative examples (in fact, all examples examined thus far) as either forgeable or unforgeable in a way that agrees with our cryptographic intuition. We show that PO certifies certain MACs as secure but are completely broken by a quantum-access attack in a *strong and intuitive sense* (Section 2.3 below). While we cannot currently prove that BU is strictly stronger than PO as a security notion, we can show that BU implies a weaker version of PO in Section 5.2.3.

Relations and characterizations. One key technical ingredient that informs our intuition and characterization results about BU is a general simulation theorem, which tightly controls the deviation in the behavior of an algorithm when subjected to the BU experiment.

Theorem 1. *Let $\mathcal{A}$ be a quantum query algorithm making at most T queries. Let $f : X \to Y$ be a function, B_ϵ a random ϵ-blinding subset of X, and for each $B \subset X$, let g_B be a function with support B. Then*

$$\mathop{\mathbb{E}}_{B_\epsilon} \left\| \mathcal{A}^f(1^n) - \mathcal{A}^{f \oplus g_{B_\epsilon}}(1^n) \right\|_1 \leq 2T\sqrt{\epsilon}\,.$$

This result can be viewed as strong evidence that algorithms that produce "good forgeries" in any reasonable sense will not be disturbed too much by blinding, and will thus also win the BU experiment. We can formulate and prove this intuition explicitly for a wide class of adversaries, as follows. Given an oracle algorithm $\mathcal{A}$, we let $\mathsf{supp}(\mathcal{A})$ denote the union of the supports of all the queries of $\mathcal{A}$, taken over all choices of oracle function.

Theorem 2 (informal)**.** *Let $\mathcal{A}$ be QPT and $\mathsf{supp}(\mathcal{A}) \cap R = \emptyset$ for some $R \neq \emptyset$. Let Mac be a MAC, and suppose $\mathcal{A}^{\mathsf{Mac}_k}(1^n)$ outputs a valid pair $(m, \mathsf{Mac}_k(m))$ with $m \in R$ with noticeable probability. Then Mac is not BU secure.*

Blind-unforgeable MACs. Next, we show that several natural constructions satisfy BU. We first show that a random function is blind-unforgeable.

Theorem 3. *Let $R : X \to Y$ be a random function such that $1/|Y|$ is negligible. Then R is a blind-unforgeable MAC.*

This together with results of Zhandry [27] and Boneh and Zhandry [6] leads to efficient BU-secure constructions.

Corollary 1. *Quantum-secure pseudorandom functions* (qPRF) *are BU-secure MACs, and (4q+1)-wise independent functions are q-BU-secure MACs.*

We can then invoke a recent result about the quantum-security of domain-extension schemes such as NMAC and HMAC [22], and obtain variable-length BU-secure MACs from any qPRF.

In the setting of public verification, we show that the one-time Lamport signature scheme [17] is BU-secure, provided that the underlying hash function family $\mathcal{R} : X \to Y$ is modeled as a random oracle.

Theorem 4. *Let $\mathcal{R} : X \to Y$ be a random function family. Then the Lamport scheme $L_\mathcal{R}$ is BU against adversaries which make one quantum query to $L_\mathcal{R}$ and poly-many quantum queries to $\mathcal{R}$.*

Hash-and-MAC. Consider the following natural variation on the blind-forgery experiment. To blind $F : X \to Y$, we first select a hash function $h : X \to Z$ and a blinding set $B_\epsilon \subseteq Z$; we then declare that F will be blinded on $x \in X$ whenever $h(x) \in B_\epsilon$. We refer to this as "hash-blinding." We call h a Bernoulli-preserving hash if, for every oracle function F, no QPT can distinguish between an oracle that has been hash-blinded with h, and an oracle that has been blinded in the usual sense. Recall the notion of *collapsing* from [25].

Theorem 5. *Let $h : X \to Y$ be a hash function. If h is Bernoulli-preserving hash, then it is also collapsing. Moreover, against adversaries with classical oracle access, h is a Bernoulli-preserving hash if and only if it is collision-resistant.*

We apply this new notion to show security of the Hash-and-MAC construction $\Pi^h = (\mathsf{Mac}^h, \mathsf{Ver}^h)$ with $\mathsf{Mac}^h_k(m) := \mathsf{Mac}_k(h(m))$.

Theorem 6. *Let $\Pi = (\mathsf{Mac}_k, \mathsf{Ver}_k)$ be a BU-secure MAC with $\mathsf{Mac}_k : X \to Y$, and let $h : Z \to X$ a Bernoulli-preserving hash. Then Π^h is a BU-secure MAC.*

We also show that the Bernoulli-preserving property can be satisfied by pseudorandom constructions, as well as a (public-key) hash based on *lossy functions* from LWE [18, 24].

2.3 A concrete "counterexample" for PO

Supporting our motivation to devise a new unforgeability definition, we present a construction of a MAC which is forgeable (in a strong intuitive sense) and yet is classified by PO as secure.

Construction 1. *Given $k = (p, f, g, h)$ where $p \in \{0,1\}^n$ is a random period and $f, g, h : \{0,1\}^n \to \{0,1\}^n$ are random functions, define $M_k : \{0,1\}^{n+1} \to \{0,1\}^{2n}$ by*

$$M_k(x) = \begin{cases} g(x' \bmod p) \| f(x') & x = 1\|x'\,, \\ 0^n \| h(x') & x = 0\|x',\ x' \neq p\,, \\ 0^{2n} & x = 0\|p\,. \end{cases}$$

Define $g_p(x) := g(x \bmod p)$ and consider an adversary that queries only on messages starting with 1, as follows:

$$\sum_{x,y} |1, x\rangle_X |0^n\rangle_{Y_1} |y\rangle_{Y_2} \longmapsto \sum_{x,y} |1, x\rangle_X |g_p(x)\rangle_{Y_1} |y \oplus f(x)\rangle_{Y_2}\,; \tag{1}$$

discarding the first qubit and Y_2 then yields $\sum_x |x\rangle|g_p(x)\rangle$, as $\sum_y |y \oplus f(x)\rangle_{Y_2} = \sum_y |y\rangle_{Y_2}$. One can then recover p via period-finding and output $(0\|p, 0^{2n})$. We emphasize that the forgery was queried with *zero* amplitude. In practice, we can interpret it as, e.g., the attacker queries only on messages starting with "From: Alice" and then forges a message starting with "From: Bob". Despite this, we can show that it is PO-secure.

Theorem 7. *The family M_k (for uniformly random $k = (p, f, g, h)$) is PO-secure.*

The PO security of M relies on a dilemma the adversary faces at each query: either learn an output of f, or obtain a superposition of $(x, g(x))$-pairs for Fourier sampling. Our proof shows that, once the adversary commits to one of these two choices, the other option is irrevocably lost. Our result can thus be understood as a refinement of an observation of Aaronson: quantumly learning a property sometimes requires *uncomputing* some information [1]. Note that, while Aaronson could rely on standard (asymptotic) query complexity techniques, our problem is quite fragile: PO security describes a task which should be hard with q queries, but is completely trivial given $q + 1$ queries. Our proof makes use of a new quantum random oracle technique of Zhandry [28].

A straightforward application of Theorem 2 shows that Construction 1 is BU-insecure. In particular, we have the following.

Corollary 2. *There exists a PO-secure MAC which is BU-insecure.*

The relationship between BU, PO and a few other notions are visualized in Figure 1.

$$\mathsf{EUF\text{-}CMA} \overset{[6]}{\Longleftrightarrow} \mathsf{PO} \overset{\text{Proposition 1}}{\Longleftrightarrow} \mathsf{BU}$$

Unforgeability against classical adversaries

$$\mathsf{PO} \overset{\text{Corollary 2}}{\not\Longrightarrow} \mathsf{BU} \underset{\text{Corollary 1}}{\overset{\text{Observation}}{\overset{\not\Longrightarrow}{\Longleftarrow}}} \mathsf{qPRF}$$

Unforgeability against quantum adversaries

Figure 1: Relationship between different unforgeability notions

2.4 Acknowledgements

The authors thanks Shih-Han Hung for discovering an error in an earlier version of this paper. CM thanks Ronald de Wolf for helpful discussions on query complexity. GA acknowledges support from the NSF under grant CCF-1763736, from the U.S. Army Research Office under Grant Number W911NF-20-1-0015, and from the U.S. Department of Energy under Award Number DE-SC0020312. CM was funded by a NWO VIDI grant (Project No. 639.022.519) and a NWO VENI grant (Project No. VI.Veni.192.159). FS acknowledges support from NSF grant CCF-2042414. AR acknowledges support from NSF grant CCF-1763773.

3 Preliminaries

Basic notation and conventions. Given a finite set X, the notation $x \in_R X$ will mean that x is a uniformly random element of X. Given a subset B of a set X, let $\chi_B : X \to \{0,1\}$ denote the characteristic function of B, i.e., $\chi_B(x) = 1$ if $x \in B$ and $\chi_B(x) = 0$ otherwise. When we say that a classical function F is efficiently computable, we mean that there exists a uniform family of deterministic classical circuits which computes F. We will consider three classes of algorithms: (i.) unrestricted algorithms, modeling computationally unbounded adversaries, (ii.) probabilistic poly-time algorithms (PPTs), modeling classical adversaries, and (iii.) quantum poly-time algorithms (QPTs), modeling quantum adversaries. We assume that the latter two are given as polynomial-time uniform families of circuits. For PPTs, these are probabilistic circuits. For QPTs, they are quantum circuits, which may contain both unitary gates and measurements. We will often assume (without loss of generality) that the measurements are postponed to the end of the circuit, and that they take place in the computational basis. Given an algorithm $\mathcal{A}$, we let $\mathcal{A}(x)$ denote the (in general, mixed) state output by $\mathcal{A}$ on input x. In particular, if $\mathcal{A}$ has classical output, then $\mathcal{A}(x)$ denotes a probability distribution. Unless otherwise stated, the probability is taken over all random coins and measurements of $\mathcal{A}$ and any randomness used to select the input x. If $\mathcal{A}$ is an oracle algorithm and F a classical function, then $\mathcal{A}^F(x)$ is the mixed state output by $\mathcal{A}$ equipped with oracle F and input x; the probability is now also taken over any randomness used to generate F.

We will distinguish between two ways of presenting a function $F : \{0,1\}^n \to \{0,1\}^m$ as an oracle. First, the usual "classical oracle access" simply means that each oracle call grants one classical invocation $x \mapsto F(x)$. This will always be the oracle model for PPTs. Second, "quantum oracle access" will mean that each oracle call grants an invocation of the $(n+m)$-qubit unitary gate $|x\rangle|y\rangle \mapsto |x\rangle|y \oplus F(x)\rangle$. This will always be the oracle model for QPTs. Note that both QPTs and unrestricted algorithms could in principle receive either oracle type.

We will need the following lemma. We use the formulation from [7, Lemma 2.1], which is a special case of a more general "pinching lemma" of Hayashi [13].

Lemma 1. *Let $\mathcal{A}$ be a quantum algorithm and $x \in \{0,1\}^*$. Let $\mathcal{A}_0$ be another quantum algorithm obtained from $\mathcal{A}$ by pausing $\mathcal{A}$ at an arbitrary stage of execution, performing a partial measurement that obtains one of k outcomes, and then resuming $\mathcal{A}$. Then* $\Pr[\mathcal{A}_0(1^n) = x] \geq \Pr[\mathcal{A}(1^n) = x]/k$.

We denote the trace distance between states ρ and σ by $\delta(\rho, \sigma)$. Recall that this is simply half the trace norm of the difference, i.e., $\delta(\rho, \sigma) = (1/2)\|\rho - \sigma\|_1$. When ρ and σ are classical probability distributions, the trace distance is equal to the total variation distance.

3.1 Quantum-secure pseudorandomness

A quantum-secure pseudorandom function (qPRF) is a family of classical, deterministic, efficiently-computable functions which appear random to QPT adversaries with quantum oracle access.

Definition 2. *An efficiently computable function family $f : K \times X \to Y$ is a quantum-secure pseudorandom function* (qPRF) *if, for all QPTs $\mathcal{D}$,*

$$\left| \Pr_{k \in_R K}\left[\mathcal{D}^{f_k}(1^n) = 1\right] - \Pr_{g \in_R \mathcal{F}_X^Y}\left[\mathcal{D}^g(1^n) = 1\right] \right| \leq \operatorname{negl}(n) \,.$$

Here $\mathcal{F}_X^Y$ denotes the set of all functions from X to Y. The standard "GGM+GL" construction of a PRF yields a qPRF when instantiated with a quantum-secure one-way function [27]. One can also construct a qPRF directly from the Learning with Errors assumption [27]. If we have an a priori bound on the number of allowed queries, then a computational assumption is not needed.

Theorem 8 (Lemma 6.4 in [6])**.** *Let $q, c \geq 0$ be integers, and $f : K \times X \to Y$ a $(2q + c)$-wise independent family of functions. Let $\mathcal{D}$ be an algorithm making no more than q quantum oracle queries and c classical oracle queries. Then*

$$\Pr_{k \in_R K}\left[\mathcal{D}^{f_k}(1^n) = 1\right] = \Pr_{g \in_R \mathcal{F}_X^Y}\left[\mathcal{D}^{g}(1^n) = 1\right].$$

3.2 PO-unforgeability

Boneh and Zhandry define unforgeability (against quantum queries) for classical MACs as follows [6]. They also show that random functions satisfy this notion.

Definition 3. *Let $\Pi = (\mathsf{KeyGen}, \mathsf{Mac}, \mathsf{Ver})$ be a MAC with message set X. Consider the following experiment with an algorithm $\mathcal{A}$:*

1. Generate key: $k \leftarrow \mathsf{KeyGen}(1^n)$.
2. Generate forgeries: *$\mathcal{A}$ receives quantum oracle for Mac_k, makes q queries, and outputs a string s;*
3. Outcome: *output* win *if s contains $q + 1$ distinct input-output pairs of Mac_k, and* fail *otherwise.*

We say that Π is PO-secure if no adversary can succeed at the above experiment with better than negligible probability.

3.3 The Fourier Oracle

Our separation proof will make use of a new technique of Zhandry [28] for analyzing random oracles. We briefly describe this framework.

A random function f from n bits to m bits can be viewed as the outcome of a quantum measurement. More precisely, let $\mathcal{H}_F = \bigotimes_{x \in \{0,1\}^n} \mathcal{H}_{F_x}$, where $\mathcal{H}_{F_x} \cong \mathbb{C}^{2^m}$. Then set $f(x) \leftarrow \mathcal{M}_{F_x}(\eta_F)$, where

$$\eta_F = |\phi_0\rangle\langle\phi_0|^{\otimes 2^n}, \qquad |\phi_0\rangle = 2^{-\frac{m}{2}} \sum_{y \in \{0,1\}^m} |y\rangle\,,$$

and $\mathcal{M}_{F_x}$ denotes the measurement of the register F_x in the computational basis. This measurement commutes with any $\mathrm{CNOT}_{A:B}$ gate with control qubit A in F_x and target qubit B outside F_x. It follows that, for any quantum algorithm making queries to a random oracle, the output distribution is identical if the algorithm is instead run with the following oracle:

1. Setup: Prepare the state η_F.
2. Upon a query with query registers X and Y, controlled on X being in state $|x\rangle$, apply $(\mathrm{CNOT}^{\otimes m})_{F_x:Y}$.
3. After the algorithm has finished, measure F to determine the success of the computation.

We denote the oracle unitary defined in step 2 above by U^{O}_{XYF}. Having defined this oracle representation, we are free to apply any unitary U_H to the oracle state, so long as we then also apply the conjugated query unitary

$$U_H(\mathrm{CNOT}^{\otimes m})_{F_x:Y}U_H^\dagger$$

in place of U^{O}_{XYF}. We choose $U_H = H^{\otimes m 2^n}$, which means that the oracle register starts in the all-zero state now. Applying Hadamard to both qubits reverses the direction of CNOT, i.e.,

$$H_A \otimes H_B \mathrm{CNOT}_{A:B} H_A \otimes H_B = \mathrm{CNOT}_{B:A}\,,$$

so the adversary-oracle-state after a first query with query state $|x\rangle_X|\phi_y\rangle_Y$ is

$$|x\rangle_X|\phi_y\rangle_Y|0^m\rangle^{\otimes 2^n} \longmapsto |x\rangle_X|\phi_y\rangle_Y|0^m\rangle^{\otimes(\mathsf{lex}(x)-1)}|y\rangle_{F_x}|0^m\rangle^{\otimes(2^n-\mathsf{lex}(x))}, \tag{2}$$

where $\mathsf{lex}(x)$ denotes the position of x in the lexicographic ordering of $\{0,1\}^n$, and we defined the Fourier basis state $|\phi_y\rangle = H^{\otimes m}|y\rangle$. In the rest of this section, we freely change the order in which tensor products are written, and keep track of the tensor factors through the use of subscripts. This adjusted representation is called the *Fourier oracle* (FO), and we denote its oracle unitary by

$$U^{\mathrm{FO}}_{XYF} = \left(H^{\otimes m2^n}\right)_F U^{\mathrm{O}}_{XYF}\left(H^{\otimes m2^n}\right)_F.$$

An essential fact about the FO is that each query can only change the number of non-zero entries in the FO's register by at most one. To formalize this idea, we define the "number operator"

$$N_F = \sum_{x\in\{0,1\}^n}(\mathbb{1}-|0\rangle\langle 0|)_{F_x}\otimes\mathbb{1}^{\otimes(2^n-1)}.$$

The number operator can also be written in its spectral decomposition,

$$N_F = \sum_{l=0}^{2^n} lP_l \qquad \text{where} \qquad P_l = \sum_{r\in S_l}|r\rangle\langle r|,$$
$$S_l = \left\{r\in\left(\{0,1\}^m\right)^{2^n}\,\middle|\,|\{x\in\{0,1\}^n|r_x\neq 0\}| = l\right\}.$$

Note that the initial joint state of a quantum query algorithm and the oracle (in the FO-oracle picture described above) is in the image of P_0. The following fact is essential for working with the Fourier Oracle; to avoid disrupting the flow of the article, the proof is given in Appendix A.1.

Lemma 2. *The number operator satisfies $\left\|\left[N_F, U^{FO}_{XYF}\right]\right\|_\infty = 1$. In particular, the joint state of a quantum query algorithm and the oracle after the q-th query is in the kernel of P_l for all $l > q$.*

4 The new notion: Blind-Unforgeability

4.1 Formal definition

For ease of exposition, we begin by introducing our new security notion in a form analogue to the standard notion of existential unforgeability under chosen-message attacks, EUF-CMA. We will also later show how to extend our approach to obtain a corresponding analogue of strong unforgeability. We begin by defining a "blinding" operation. Let $f: X\to Y$ and $B\subseteq X$. We let

$$Bf(x) = \begin{cases}\perp & \text{if } x\in B,\\ f(x) & \text{otherwise.}\end{cases}$$

We say that f has been "blinded" by B. In this context, we will be particularly interested in the setting where elements of X are placed in B independently at random with a particular probability ϵ; we let B_ϵ denote this random variable. (It will be easy to infer X from context so we do not reflect it in the notation.)

Next, we define a security game in which an adversary is tasked with using a blinded MAC oracle to produce a valid input-output pair in the blinded set.

Definition 4. *Let $\Pi = (\mathsf{KeyGen},\mathsf{Mac},\mathsf{Ver})$ be a MAC with message set X. Let $\mathcal{A}$ be an algorithm, and $\epsilon:\mathbb{N}\to\mathbb{R}_{\geq 0}$ an efficiently computable function. The blind forgery experiment $\mathsf{BlindForge}_{\mathcal{A},\Pi}(n,\epsilon)$ proceeds as follows:*

1. Generate key: $k \leftarrow \mathsf{KeyGen}(1^n)$.

2. Generate blinding: *select* $B_\epsilon \subseteq X$ *by placing each* m *into* B_ϵ *independently with probability* $\epsilon(n)$*.*

3. Produce forgery: $(m,t) \leftarrow \mathcal{A}^{B_\epsilon \mathsf{Mac}_k}(1^n)$.

4. Outcome: *output* 1 *if* $\mathsf{Ver}_k(m,t) = \mathsf{acc}$ *and* $m \in B_\epsilon$*; otherwise output* 0*.*

We say that a scheme is blind-unforgeable if, for any efficient adversary, the probability of winning the game is negligible. The probability is taken over the choice of key, the choice of blinding set, and any internal randomness of the adversary. We remark that specifying an adversary requires specifying (in a uniform fashion) both the algorithm $\mathcal{A}$ and the blinding fraction ϵ.

Definition 5. *A MAC* Π *is blind-unforgeable (BU) if for every polynomial-time uniform adversary* $(\mathcal{A}, \epsilon)$*,*

$$\Pr\big[\mathsf{BlindForge}_{\mathcal{A},\Pi}(n, \epsilon(n)) = 1\big] \leq \mathsf{negl}(n)\,.$$

We also define the "q-time" variant of the blinded forgery game, which is identical to Definition 4 except that the adversary is only allowed to make q queries to $B_\epsilon \mathsf{Mac}_k$ in step (3). We call the resulting game $\mathsf{BlindForge}^q_{\mathcal{A},\Pi}(n, \epsilon)$, and give the corresponding definition of q-time security (now against computationally unbounded adversaries).

Definition 6. *A MAC* Π *is* q*-time blind-unforgeable (*q*-BU) if for every* q*-query adversary* $(\mathcal{A}, \epsilon)$*, we have*

$$\Pr\big[\mathsf{BlindForge}^q_{\mathcal{A},\Pi}(n, \epsilon(n)) = 1\big] \leq \mathsf{negl}(n)\,.$$

The above definitions are agnostic regarding the computational power of the adversary and the type of oracle provided. For example, selecting PPT adversaries and classical oracles in Definition 5 yields a definition of classical unforgeability; we will later show that this is equivalent to standard EUF-CMA. The main focus of our work will be on BU against QPTs with quantum oracle access, and q-BU against unrestricted adversaries with quantum oracle access.

4.2 Some technical details

We now remark on a few details in the usage of BU. First, strictly speaking, the blinding sets in the security games above cannot be generated efficiently. However, a pseudorandom blinding set will suffice. Pseudorandom blinding sets can be generated straightforwardly using an appropriate pseudorandom function, such as a PRF against PPTs or a qPRF against QPT. A precise description of how to perform this pseudorandom blinding is given in the proof of Corollary 4. Note that simulating the blinding requires computing and uncomputing the random function, so we must make two quantum queries for each quantum query of the adversary. Moreover, verifying whether the forgery is in the blinding set at the end requires one additional classical query. This means that $(4q+1)$-wise independent functions are both necessary and sufficient for generating blinding sets for q-query adversaries (see [6, Lemma 6.4]). In any case, an adversary which behaves differently in the random-blinding game versus the pseudorandom-blinding game immediately yields a distinguisher against the corresponding pseudorandom function.

The blinding symbol. There is some flexibility in how one defines the blinding symbol $\perp$. In situations where the particular instantiation of the blinding symbol might matter, we will adopt the convention that the blinded version Bf of $f : \{0,1\}^n \rightarrow \{0,1\}^\ell$ is defined by setting $Bf : \{0,1\}^n \rightarrow \{0,1\}^{\ell+1}$, where $Bf(m) = 0^\ell||1$ if $m \in B$ and $Bf(m) = f(m)||0$ otherwise. One advantage of this convention (i.e., that $\perp = 0^\ell||1$) is that we can compute on and/or measure the blinded bit (i.e., the $(\ell+1)$-st bit) without affecting the output register of the function. This will also turn out to be convenient for uncomputation.

Strong blind-unforgeability. The security notion BU given in Definition 5 is an analogue of simple unforgeability, i.e., EUF-CMA, for the case of a quantum-accessible MAC/Signing oracle. It is, however, straightforward to define a corresponding analogue of strong unforgeability, i.e., SUF-CMA, as well.

The notion of strong blind-unforgeability, sBU, is obtained by a simple adjustment compared to BU: we blind (message, tag) pairs rather than just messages. We briefly describe this for the case of MACs. Let $\Pi = (\mathsf{KeyGen}, \mathsf{Mac}, \mathsf{Ver})$ be a MAC with message set M, randomness set R and tag set T, so that $\mathsf{Mac}_k : M \times R \to T$ and $\mathsf{Ver}_k : M \times T \to \{\mathsf{acc}, \mathsf{rej}\}$ for every $k \leftarrow \mathsf{KeyGen}$. Given a parameter ϵ and an adversary $\mathcal{A}$, the strong blind forgery game proceeds as follows:

1. Generate key: $k \leftarrow \mathsf{KeyGen}$; generate blinding: select $B_\epsilon \subseteq M \times T$ by placing pairs (m, t) in B_ϵ independently with probability ϵ;
2. Produce forgery: produce (m, t) by executing $\mathcal{A}(1^n)$ with quantum oracle access to the function
$$B_\epsilon \mathsf{Mac}_{k;r}(m) := \begin{cases} \perp & \text{if } (m, \mathsf{Mac}_k(m; r)) \in B_\epsilon, \\ \mathsf{Mac}_k(m; r) & \text{otherwise.} \end{cases}$$
where r is sampled uniformly for each oracle call.
3. Outcome: output 1 if $\mathsf{Ver}_k(m, t) = \mathsf{acc} \wedge (m, t) \in B_\epsilon$; otherwise output 0.

Security is then defined as before: Π is sBU-secure if for all adversaries $\mathcal{A}$ (and their declared ϵ), the success probability at winning the above game is negligible. Note that, for the case of canonical MACs, this definition coincides with Definition 5, just as $\mathsf{EUF\text{-}CMA}$ and $\mathsf{SUF\text{-}CMA}$ coincide in this case.

5 Intuitive security and the meaning of BU

In this section, we gather a number of results which build confidence in BU as a satisfactory definition of unforgeability in our setting. We begin by showing that a wide range of "intuitively forgeable" MACs (indeed, all such examples we have examined) are correctly characterized by BU as insecure.

5.1 Intuitively forgeable schemes

As indicated earlier, BU security rules out any MAC schemes where an attacker can query a subset of the message space and forge outside that region. To make this claim precise, we first define the *query support* $\mathsf{supp}(\mathcal{A})$ of an oracle algorithm $\mathcal{A}$. Let $\mathcal{A}$ be a quantum query algorithm with oracle access to the quantum oracle O for a classical function from n to m bits. Without loss of generality $\mathcal{A}$ proceeds by applying the sequence of unitaries $\mathcal{O}U_q\mathcal{O}U_{q-1}...U_1$ to the initial state $|0\rangle_{XYZ}$, followed by a POVM $\mathcal{E}$. Here, X and Y are the input and output registers of the function and Z is the algorithm's workspace. Let $|\psi_i\rangle$ be the intermediate state of of $\mathcal{A}$ after the application of U_i. Then $\mathsf{supp}(\mathcal{A})$ is defined to be the set of input strings x such that there exists a function $f : \{0,1\}^n \to \{0,1\}^m$ such that $\langle x|_X|\psi_i\rangle \neq 0$ for at least one $i \in \{1, ..., q\}$ when $\mathcal{O} = \mathcal{O}_f$.

Theorem 9. *Let $\mathcal{A}$ be a QPT such that $\mathsf{supp}(\mathcal{A}) \cap R = \emptyset$ for some $R \neq \emptyset$. Let Mac be a MAC, and suppose $\mathcal{A}^{\mathsf{Mac}_k}(1^n)$ outputs a valid pair $(m, \mathsf{Mac}_k(m))$ with $m \in R$ with non-negligible probability. Then Mac is not BU-secure.*

To prove Theorem 9, we will need the following theorem, which controls the change in the output state of an algorithm resulting from applying a blinding to its oracle. Given an oracle algorithm $\mathcal{A}$ and two oracles F and G, the trace distance between the output of $\mathcal{A}$ with oracle F and $\mathcal{A}$ with oracle G is denoted by $\delta(\mathcal{A}^F(1^n), \mathcal{A}^G(1^n))$. Given two functions $F, P : \{0,1\}^n \to \{0,1\}^m$, we define the function $F \oplus P$ by $(F \oplus P)(x) = F(x) \oplus P(x)$.

Theorem 10. *Let $\mathcal{A}$ be a quantum query algorithm making at most T queries, and $F : \{0,1\}^n \to \{0,1\}^m$ a function. Let $B \subseteq \{0,1\}^n$ be a subset chosen by independently including each element of $\{0,1\}^n$ with probability ϵ, and $P : \{0,1\}^n \to \{0,1\}^m$ be any function with support B. Then*
$$\mathop{\mathbb{E}}_B\big[\delta\big(\mathcal{A}^F(1^n), \mathcal{A}^{F\oplus P}(1^n)\big)\big] \leq 2T\sqrt{\epsilon}.$$

The proof is a relatively straightforward hybrid argument in the spirit of the lower bound for Grover search [4]. We provide the complete proof in Appendix A.2. We are now ready to prove Theorem 9.

Proof of Theorem 9. Let $\mathcal{A}$ be a quantum algorithm with $\mathsf{supp}(\mathcal{A})$ for any oracle. By our hypothesis,

$$\tilde{p} := \Pr_{k,(m,t)\leftarrow\mathcal{A}^{\mathsf{Mac}_k}(1^n)}\left[\mathsf{Mac}_k(m) = t \wedge m \notin \mathsf{supp}(\mathcal{A})\right] \geq n^{-c}\,,$$

for some $c > 0$ and sufficiently large n. Since $\mathsf{supp}(A)$ is a fixed set, we can think of sampling a random B_ε as picking $B_0 := B_\varepsilon \cap \mathsf{supp}(A)$ and $B_1 := B_\varepsilon \cap \overline{\mathsf{supp}(A)}$ independently. Let "blind" denote the random experiment of $\mathcal{A}$ running on Mac_k blinded by a random B_ε: $k, B_\varepsilon, (m,t) \leftarrow \mathcal{A}^{B_\varepsilon\mathsf{Mac}_k}(1^n)$, which is equivalent to $k, B_0, B_1, (m,t) \leftarrow \mathcal{A}^{B_0\mathsf{Mac}_k}(1^n)$. The probability that $\mathcal{A}$ wins the BU game is

$$\begin{aligned}
p &:= \Pr_{\text{blind}}[f(m) = t \wedge m \in B_\varepsilon] \geq \Pr_{\text{blind}}[f(m) = t \wedge m \in B'] \\
&\geq \Pr_{\text{blind}}[f(m) = t \wedge m \in B' \mid m \notin \mathsf{supp}(A)] \cdot \Pr_{\text{blind}}[m \notin \mathsf{supp}(A)] \\
&= \Pr_{\substack{f,B_0 \\ (m,t)\leftarrow\mathcal{A}^{Bf}}}[f(m) = t \wedge m \notin \mathsf{supp}(A)] \cdot \Pr_{\substack{f,B' \\ (m,t)\leftarrow\mathcal{A}^{Bf}}}[m \in B' | m \notin \mathsf{supp}(A)] \\
&\geq \left(\tilde{p} - 2T\sqrt{\varepsilon}\right)\varepsilon \geq \frac{\tilde{p}^3}{27T^2}\,.
\end{aligned}$$

Here the second-to-last step follows from Theorem 10; in the last step, we chose $\varepsilon = (\tilde{p}/3T)^2$. We conclude that $\mathcal{A}$ breaks the BU security of the MAC. $\square$

5.2 Relationship to other definitions

5.2.1 Classical BU is equivalent to $\mathsf{EUF\text{-}CMA}$

In the purely classical setting, our notion is equivalent to EUF-CMA. In the strong unforgeability case, this means BU with blinding on message-tag pairs, as described in Section 4.2.

Proposition 1. *A MAC is EUF-CMA if and only if it is blind-unforgeable against classical adversaries.*

Proof. Set $F_k = \mathsf{Mac}_k$. Consider an adversary $\mathcal{A}$ which violates EUF-CMA. Such an adversary, given 1^n and oracle access to F_k (for $k \in_R \{0,1\}^n$), produces a forgery (m,t) with non-negligible probability $s(n)$; in particular, $|m| \geq n$ and m is not among the messages queried by $\mathcal{A}$. This same adversary (when coupled with an appropriate ϵ) breaks the system under the blind-forgery definition. Specifically, let $p(n)$ be the running time of $\mathcal{A}$, in which case $\mathcal{A}$ clearly makes no more than $p(n)$ queries, and define $\epsilon(n) = 1/p(n)$. Consider now a particular $k \in \{0,1\}^n$ and a particular sequence r of random coins for $\mathcal{A}^{F_k}(1^n)$. If this run of $\mathcal{A}$ results in a forgery (m,t), observe that with probability at least $(1-\epsilon)^{p(n)} \approx e^{-1}$ in the choice of B_ϵ, we have $F_k(q) = B_\epsilon F_k(q)$ for every query q made by $\mathcal{A}$. On the other hand, $B_\epsilon(m) = \perp$ with (independent) probability ϵ. It follows $\phi(n, \epsilon_n)$ is at least $\epsilon s(n)/e = \Omega(s(n)/p(n))$.

On the the other hand, suppose that $(\mathcal{A}, \epsilon)$ is an adversary that breaks blind-unforgeability. Consider now the EUF-CMA adversary $\mathcal{A}'^{F_k}(1^n)$ which simulates the adversary $\mathcal{A}^{(\cdot)}(1^n)$ by answering oracle queries according to a locally-simulated version of $B_\epsilon F_k$; specifically, the adversary $\mathcal{A}'$ proceeds by drawing a subset $B_{\epsilon(n)} \subseteq \{0,1\}^*$ as described above and answering queries made by $\mathcal{A}$ according to $B_\epsilon F$. Two remarks are in order:

- When $x \in B_\epsilon$, this query is answered without an oracle call to $F(x)$.
- $\mathcal{A}'$ can construct the set B_ϵ "on the fly," by determining, when a particular query q is made by $\mathcal{A}$, whether $q \in B_\epsilon$ and "remembering" this information in case the query is asked again ("lazy sampling").

With probability $\phi(n, \epsilon(n))$ $\mathcal{A}$ produces a forgery on a point which was not queried by $\mathcal{A}'$, as desired. It follows that $\mathcal{A}$ produces a (conventional) forgery with non-negligible probability when given F_k for $k \in_R \{0,1\}^n$. $\square$

5.2.2 BU implies GYZ

In this section, we sketch how our new security notion, BU, implies the one-time security notion put forward by Garg, Yuen and Zhandry [12]. We do this for unitary adversaries without loss of generality. We expect that the ideas carry over to the fully general case. For the special case we consider here, GYZ unforgeability can be defined as follows.

Definition 7 (GYZ unforgeability)**.** *Let* $\Pi = (\mathsf{KeyGen}, \mathsf{Mac}, \mathsf{Ver})$ *be a MAC.* Π *is called* ε-GYZ*-unforgeable, if for attack unitary* V_{MTB}*, there exists a simulator sub-unitary* W_{MTB}[4] *such that* $\langle i|_C W|j\rangle_C = 0$ *for* $i \neq j$ *and for all initial states* $|\psi\rangle_{MB}$

$$\mathbb{E}_{r,k} \left\| \left(\Pi_{\mathsf{Ver},k}\right)_{MT} (V - W)_{MTB} \left(U_{\mathsf{Mac},r,k}\right)_{MT} |\psi\rangle_{MB}|0\rangle_T \right\|^2 \leq \varepsilon, \tag{3}$$

where $U_{\mathsf{Mac},r,k}|m\rangle_M|y\rangle_T = |m\rangle_M|y \oplus \mathsf{Mac}_{r;k}(m)\rangle_T$ *and*

$$\Pi_{\mathsf{Ver},k} = \sum_{(m,t)\ \text{valid}} |m\rangle\langle m| \otimes |t\rangle\langle t| \tag{4}$$

is the projector onto the subspace of valid message-tag-pairs.

We are now ready to state the desired theorem. Here, δ-1-BU denotes the one-time version of BU that allows for maximal adversarial advantage δ, and similarly for δ'-GYZ. The theorem is easiest proven using the measured version of BU, mBU.

Theorem 11. *Let* $\Pi = (\mathsf{KeyGen}, \mathsf{Mac}, \mathsf{Ver})$ *be unconditionally*[5] δ*-1-*mBU*-secure. Then it is* 16δ-GYZ*-unforgeable.*

mBU implies BU, so we immediately obtain the following

Corollary 3. *Let* $\Pi = (\mathsf{KeyGen}, \mathsf{Mac}, \mathsf{Ver})$ *be* δ*-1-*BU*-secure. Then it is* 16δ-GYZ*-unforgeable.*

For the proof of Theorem 11, we need the following lemma stating that an algorithm's success probability does not degrade too much if some operation that never leaves a computational basis state unchanged is sandwiched between a blinding projector and its complement.

Lemma 3. *Let* $\Pi_A^{(\varepsilon)}$ *be the projector onto the subspace spanned by the computational basis states corresponding to strings in the blinding set* B_ε *sampled as for the definition of* BU*. Let further* M_{AB} *be a matrix such that* $\langle x|_A M|x\rangle_A = 0$ *for all* x*. Then for all* $\rho \geq 0$

$$\mathbb{E}_{B_\varepsilon}\left[\Pi_A^{(\varepsilon)} M_{AB} \bar{\Pi}_A^{(\varepsilon)} \rho_{AB} \bar{\Pi}_A^{(\varepsilon)} M_{AB}^\dagger \Pi_A^{(\varepsilon)}\right] \geq \varepsilon^2(1-\varepsilon)^2 M_{AB}\rho_{AB}M_{AB}^\dagger, \tag{5}$$

where $\bar{\Pi}_A^{(\varepsilon)} = \mathbb{1} - \Pi_A^{(\varepsilon)}$.

Proof. Let

$$M_{AB} = \sum_{x \neq y} |x\rangle\langle y|_A \otimes M_B^{(xy)} \tag{6}$$

and

$$\rho_{AB} = \sum_{xy} |x\rangle\langle y|_A \otimes \rho_B^{(xy)}. \tag{7}$$

[4] A matrix V is sub-unitary if $V^\dagger V \leq \mathbb{1}$.

[5] Here, "unconditionally" means without any assumption on the computational complexity of the adversary.

We calculate

$$\mathbb{E}_{B_\varepsilon}\left[\Pi_A^{(\varepsilon)}M_{AB}\bar{\Pi}_A^{(\varepsilon)}\rho\bar{\Pi}_A^{(\varepsilon)}M_{AB}^\dagger\Pi_A^{(\varepsilon)}\right] \tag{8}$$

$$= \sum_{x\neq y, z\neq t}\Pr\left[y,z\in B_\varepsilon, x,t\notin B_\varepsilon\right]|x\rangle\langle t|_A\otimes\left(M_B^{(xy)}\rho_B^{(yz)}\left(M_B^{(tz)}\right)^\dagger\right) \tag{9}$$

$$= \varepsilon^2(1-\varepsilon)^2M_{AB}\rho_{AB}M_{AB}^\dagger + \varepsilon(1-\varepsilon)^3M_{AB}\mathcal{M}(\rho_{AB})M_{AB}^\dagger \tag{10}$$

$$+\,\varepsilon^3(1-\varepsilon)\mathcal{M}(M_{AB}\rho_{AB}M_{AB}^\dagger) \tag{11}$$

$$+\,\varepsilon^2(1-\varepsilon)^2\sum_x|x\rangle\langle x|_AM_{AB}|x\rangle\langle x|_A\rho_{AB}|x\rangle\langle x|_AM_{AB}^\dagger|x\rangle\langle x|_A. \tag{12}$$

In the last three lines,

$$\mathcal{M}(X) = \sum_x|x\rangle\langle x|X|x\rangle\langle x| \tag{13}$$

is the computational basis pinching channel, we used the condition $\langle i|_AM|i\rangle_B = 0$ so we can add these terms for free, and the second, third and fourth term arise because of the increase of the probability $\Pr\left[y,z\in B_\varepsilon, x,t\notin B_\varepsilon\right]$ if $y=z$, $x=t$, or both. But the second, third and fourth term are positive semidefinite, so the claimed operator inequality follows. □

Proof of Theorem 11. Let

$$V_{MTB} = \sum_{xy}|x\rangle\langle y|_M\otimes V_{TB}^{(xy)} \tag{14}$$

be an attack unitary that breaks δ'-GYZ, and define the simulator sub-unitary

$$W_{MTB} = \sum_x|x\rangle\langle x|_M\otimes V_{TB}^{(xx)} \tag{15}$$

to be the diagonal part of V. By assumption there exists a state $|\psi\rangle_{MB}$ such that

$$\mathbb{E}_k\left\|\left(\Pi_{\mathsf{Ver},k}\right)_{MT}(V-W)_{MTB}\left(U_{\mathsf{Mac},r,k}\right)_{MT}|\psi\rangle_{MB}|0\rangle_T\right\|^2 > \delta'. \tag{16}$$

Now we use the initial state $|\psi\rangle_{MB}|0\rangle_T$ and the unitary V as attacker $\mathcal{A}$ for 1-mBU: $\mathcal{A}$ queries the mBU-type oracle for Mac_k on $|\psi\rangle_{MB}|0\rangle_T$, applies V, measures MT and outputs the result. The success probability p_{succ} of this adversary can be lower bounded by the probability that the first measurement turns up "not blinded" and the adversary is successful. So for a fixed key k and randomness r, and defining $\rho = |\phi\rangle\langle\phi|$ with $|\phi\rangle = \left(U_{\mathsf{Mac},r,k}\right)_{MT}|\psi\rangle_{MB}|0\rangle_T$ we get

$$p_{\text{succ}}(k,r) \geq \mathbb{E}_{B_\varepsilon}\left\|\left(\Pi_{\mathsf{Ver},k}\right)_{MT}\Pi_M^{(\varepsilon)}V_{MTB}\bar{\Pi}_M^{(\varepsilon)}\left(U_{\mathsf{Mac},r,k}\right)_{MT}|\psi\rangle_{MB}|0\rangle_T\right\|^2 \tag{17}$$

$$= \mathbb{E}_{B_\varepsilon}\left\|\left(\Pi_{\mathsf{Ver},k}\right)_{MT}\Pi_M^{(\varepsilon)}(V-W)_{MTB}\bar{\Pi}_M^{(\varepsilon)}\left(U_{\mathsf{Mac},r,k}\right)_{MT}|\psi\rangle_{MB}|0\rangle_T\right\|^2 \tag{18}$$

$$= \mathbb{E}_{B_\varepsilon}\operatorname{Tr}\left[\left(\Pi_{\mathsf{Ver},k}\right)_{MT}\Pi_M^{(\varepsilon)}(V-W)_{MTB}\bar{\Pi}_M^{(\varepsilon)}\rho\bar{\Pi}_M^{(\varepsilon)}(V-W)_{MTB}^\dagger\Pi_M^{(\varepsilon)}\right] \tag{19}$$

$$\geq \varepsilon^2(1-\varepsilon)^2\operatorname{Tr}\left[\left(\Pi_{\mathsf{Ver},k}\right)_{MT}(V-W)_{MTB}\rho(V-W)_{MTB}^\dagger\right] \tag{20}$$

$$= \varepsilon^2(1-\varepsilon)^2\left\|\left(\Pi_{\mathsf{Ver},k}\right)_{MT}(V-W)_{MTB}\left(U_{\mathsf{Mac},r,k}\right)_{MT}|\psi\rangle_{MB}|0\rangle_T\right\|^2. \tag{21}$$

Here we have used the fact that $\Pi_M^{(\varepsilon)}W_{MTB}\bar{\Pi}_M^{(\varepsilon)} = 0$ for all blinding sets in the second line, and Lemma 3 in the second-to-last line. Taking the expectation over k and r and using Equation (16), we arrive at

$$p_{\text{succ}} \geq \varepsilon^2(1-\varepsilon)^2\delta'. \tag{22}$$

Choosing $\epsilon = 1/2$ we obtain

$$p_{\text{succ}} \geq \frac{1}{16}\delta'. \tag{23}$$

□

5.2.3 BU implies quadratic PO

It is interesting to ask if BU-security implies PO-security, as the PO definition certainly captures a natural family of attacks that one would like to rule out. We are unable to settle this question completely, but provide some weaker connection. Specifically, we show that if a function is BU-secure, then it is PO-secure with a weaker definition of PO-security that forbids an adversary from producing ck^2 forgeries from k queries with high probability.

For this purpose, consider a function $M : X \to Y$ and a PO-type adversary $\mathcal{A}$ which, given oracle access to M, makes some k queries and produces ck^2 forgeries (with probability 1); here $c \geq 1$ is a constant we set later in the discussion. We consider the behavior of this adversary $\mathcal{A}^{B_\epsilon M}$ supplied with an oracle $B_\epsilon M$ blinded at a random set B_ϵ. We will show that for an appropriate value of c and ϵ, this adversary produces a family of forgeries which includes at least one blinded forgery with constant probability. Finally selecting one of these forgeries at random produces an adversary that breaks the BU security definition.

Returning to the PO-adversary $\mathcal{A}$, we say that a particular blinding set B is γ-*evasive* if

$$\Pr_{\mathcal{A}}[\mathcal{A}^M \text{ outputs no elements of } B] \geq \gamma\,.$$

(Note that this event is determined by running $\mathcal{A}$ with the *unblinded* oracle M.) Observing that

$$\Pr_{\mathcal{A},B_\epsilon}[\mathcal{A}^M \text{ outputs no elements of } B_\epsilon] \leq (1-\epsilon)^{ck^2} \leq e^{-c\epsilon k^2}\,.$$

We note that (by Markov's inequality),

$$\Pr_{B_\epsilon}[B_\epsilon \text{ is } \gamma\text{-evasive}] \leq e^{-c\epsilon k}/\gamma\,.$$

Similarly, we say that a particular blinding set B is γ-divergent if

$$\|D_{\mathcal{A}^M} - D_{\mathcal{A}^{BM}}\|_{\text{t.v.}} \geq \gamma\,,$$

where D_M is the distribution of outputs of $\mathcal{A}^M$ and D_{BM} is the distribution of outputs of $\mathcal{A}^{BM}$ when M is blinded on set B. In light of Theorem 1,

$$\mathop{\mathbb{E}}_{B_\epsilon}\left[\|D_M - D_{B_\epsilon M}\|_{\text{t.v.}}\right] \leq 2k\sqrt{\epsilon}$$

and it follows by Markov's inequality that

$$\Pr_{B_\epsilon}[B_\epsilon \text{ is } \gamma\text{-divergent}] = \Pr_{B_\epsilon}[\|D_M - D_{BM}\|_{\text{t.v.}} \geq \gamma] \leq 2k\sqrt{\epsilon}/\gamma\,.$$

Fixing $\gamma \leq 1/2 - \delta$ for $\delta > 0$, note that if B is neither γ-evasive nor γ-divergent then

$$\Pr_{\mathcal{A}}[\mathcal{A}^M \text{ outputs an element of } B] \geq 1 - \gamma\,,$$

(associated with the distribution D_M), and hence

$$\Pr_{\mathcal{A}}[\mathcal{A}^{BM} \text{ outputs an element of } B] \geq 1 - 2\gamma \geq 2\delta\,.$$

Finally, note that the probability that B is $(1/2-\delta)$-evasive or $(1/2-\delta)$-divergent is no more than

$$\frac{1}{1/2-\delta}\underbrace{\left[e^{-c\epsilon k^2} + 2k\sqrt{\epsilon}\right]}_{(\dagger)}\,.$$

Then it is clear that one can choose the constants δ and c, and the blinding probability $\epsilon = \Theta(1/k^2)$, so that this quantity is a constant bounded away from one. (For example, set $\delta = 1/6$. Then, with $\epsilon = 1/(144k^2)$ the second term of ($\dagger$) above is no more than 1/6; setting $c = 288$ guarantees the first term is likewise no more than $e^{-2} < 1/6$ and the entire expression is a constant less than one. One can achieve better constants with more care, but the quadratic dependence on ϵ in Theorem 1 dictates the quadratic gap between k and the number of forgeries achieved by this simple method of proof.)

Finally, we create a BU adversary for M by running the PO adversary, blinded as above with $\epsilon = \Theta(1/k^2)$, and selecting one of the ck^2 output values at random.

6 Blind-unforgeable schemes

6.1 Random schemes

We now show that suitable random and pseudorandom function families satisfy our notion of unforgeability.

Theorem 12. *Let $R : X \to Y$ be a uniformly random function such that $1/|Y|$ is negligible in n. Then R is a blind-forgery secure MAC.*

Proof. For simplicity, we assume that the function is length-preserving; the proof generalizes easily. Let $\mathcal{A}$ be an efficient quantum adversary. The oracle $B_\epsilon R$ supplied to $\mathcal{A}$ during the blind-forgery game is determined entirely by B_ϵ and the restriction of R to the complement of B_ϵ. On the other hand, the forgery event

$$\mathcal{A}^{B_\epsilon F_k}(1^n) = (m, t) \wedge |m| \geq n \wedge F_k(m) = t \wedge B_\epsilon F_k(m) = \perp$$

depends additionally on values of R at points in B_ϵ. To reflect this decomposition, given R and B_ϵ define $R_\epsilon : B_\epsilon \to Y$ to be the restriction of R to the set B_ϵ and note that—conditioned on $B_\epsilon R$ and B_ϵ—the random variable R_ϵ is drawn uniformly from the space of all (length-preserving) functions from B_ϵ into Y. Note, also, that for every n the purported forgery $(m, t) \leftarrow \mathcal{A}^{B_\epsilon R}(1^n)$ is a (classical) random variable depending only on $B_\epsilon R$. In particular, conditioned on B_ϵ, (m, t) is independent of R_ϵ. It follows that, conditioned on $m \in B_\epsilon$, that $t = R_\epsilon(m)$ with probability no more than $1/2^n$ and hence $\phi(n, \epsilon) \leq 2^{-n}$, as desired. □

Next, we show that a qPRF is a blind-unforgeable MAC.

Corollary 4. *Let m and t be $\mathsf{poly}(n)(n)$, and $F : \{0,1\}^n \times \{0,1\}^m \to \{0,1\}^t$ a qPRF. Then F is a blind-forgery-secure fixed-length MAC (with length $m(n)$).*

Proof. For a contradiction, let $\mathcal{A}$ be a QPT which wins the blind forgery game for a certain blinding factor $\varepsilon(n)$, with running time $q(n)$ success probability $\delta(n)$. We will use $\mathcal{A}$ to build a quantum oracle distinguisher $\mathcal{D}$ between the qPRF F and the perfectly random function family $\mathcal{F}_m^t$ with the same domain and range.

First, let $k = q(n)$ and let $\mathcal{H}$ be a family of $(4k+1)$-wise independent functions with domain $\{0,1\}^m$ and range $\{0, 1, \ldots, 1/\varepsilon(n)\}$. The distinguisher $\mathcal{D}$ first samples $h \in_R \mathcal{H}$. Set $B_h := h^{-1}(0)$. Given its oracle $\mathcal{O}_f$, $\mathcal{D}$ can implement the function $B_h f$ (quantumly) as follows:

$$\begin{aligned} |x\rangle|y\rangle \mapsto & |x\rangle|y\rangle|H_x\rangle|\delta_{h(x),0}\rangle \mapsto |x\rangle|y\rangle|H_x\rangle|\delta_{h(x),0}\rangle|f(x)\rangle \\ \mapsto & |x\rangle|y \oplus f(x) \cdot (1 - \delta_{h(x),0})\rangle|H_x\rangle|\delta_{h(x),0}\rangle|f(x)\rangle \\ \mapsto & |x\rangle|y \oplus f(x) \cdot (1 - \delta_{h(x),0})\rangle\,. \end{aligned}$$

Here we used the CCNOT (Toffoli) gate from step 2 to 3 (with one control bit reversed), and uncomputed both h and f in the last step. After sampling h, the distinguisher $\mathcal{D}$ will execute $\mathcal{A}$ with the oracle $B_h f$. If $\mathcal{A}$ successfully forges a tag for a message in B_h, $\mathcal{A}'$ outputs "pseudorandom"; otherwise "random."

Note that the function $B_h f$ is perfectly ϵ-blinded if h is a perfectly random function. Note also that the entire security experiment with $\mathcal{A}$ (including the final check to determine if the output forgery is blind) makes at most $2k$ quantum queries and 1 classical query to h, and is thus (by Theorem 8) identically distributed to the perfect-blinding case.

Finally, by Theorem 12, the probability that $\mathcal{D}$ outputs "pseudorandom" when $f \in_R \mathcal{F}_m^t$ is negligible. By our initial assumption about $\mathcal{A}$, the probability that $\mathcal{D}$ outputs "pseudorandom" becomes $\delta(n)$ when $f \in_R F$. It follows that $\mathcal{D}$ distinguishes F from perfectly random. □

Next, we give a information-theoretically secure q-time MACs (Definition 6).

Theorem 13. *Let $\mathcal{H}$ be a $(4q+1)$-wise independent function family with range Y, such that $1/|Y|$ is a negligible function. Then $\mathcal{H}$ is a q-time BU-secure MAC.*

Proof. Let $(\mathcal{A}, \epsilon)$ be an adversary for the q-time game $\mathsf{BlindForge}^q_{\mathcal{A},h}(n, \epsilon(n))$, where h is drawn from $\mathcal{H}$. We will use $\mathcal{A}$ to construct a distinguisher $\mathcal{D}$ between $\mathcal{H}$ and a random oracle. Given access to an oracle $\mathcal{O}$, $\mathcal{D}$ first runs $\mathcal{A}$ with the blinded oracle $B\mathcal{O}$, where the blinding operation is performed as in the proof of Corollary 4 (i.e., via a $(4q+1)$-wise independent function with domain size $1/\epsilon(n)$). When $\mathcal{A}$ is completed, it outputs (m, σ). Next, $\mathcal{D}$ queries $\mathcal{O}$ on the message m and outputs 1 if and only if $\mathcal{O}(m) = \sigma$ and $m \in B$. Let $\gamma_{\mathcal{O}}$ be the probability of the output being 1.

We consider two cases: (i.) $\mathcal{O}$ is drawn as a random oracle R, and (ii.) $\mathcal{O}$ is drawn from the family $\mathcal{H}$. By Theorem 8, since $\mathcal{D}$ makes only $2q$ quantum queries and one classical query to $\mathcal{O}$, its output is identical in the two cases. Observe that γ_R (respectively, $\gamma_{\mathcal{H}}$) is exactly the success probability of $\mathcal{A}$ in the blind-forgery game with random oracle R (respectively, $\mathcal{H}$). We know from Theorem 12 that γ_R is negligible; it follows that $\gamma_{\mathcal{H}}$ is as well. ☐

Several domain-extension schemes, including NMAC (a.k.a. encrypted cascade), HMAC, and AMAC, can transform a fixed-length qPRF to a qPRF that takes variable-length inputs [22] . As a corollary, starting from a qPRF, we also obtain a number of quantum blind-unforgeable variable-length MACs.

6.2 Lamport one-time signatures

The Lamport signature scheme [17] is a EUF-1-CMA-secure signature scheme, specified as follows.

Construction 2 (Lamport signature scheme, [17])**.** *For the Lamport signature scheme using a hash function family $h : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$, the algorithms* $\mathsf{KeyGen}, \mathsf{Sign}$ *and* Ver *are specified as follows.* KeyGen*, on input 1^n, outputs a pair* $(\mathsf{pk}, \mathsf{sk})$ *with*

$$\mathsf{sk} = (s_i^j)_{i\in\{1,\dots,n\},j=0,1},\ \text{with } s_i^j \in_R \{0,1\}^n,\ \text{and} \tag{24}$$

$$\mathsf{pk} = \left(k, \left(p_i^j\right)_{i\in\{1,\dots,n\},j=0,1}\right),\ \text{with } k \in \{0,1\}^n \text{ and } p_i^j = h_k\left(s_i^j\right). \tag{25}$$

The signing algorithm is defined by $\mathsf{Sign}_{\mathsf{sk}}(x) = (s_i^{x_i})_{i\in\{1,\dots,n\}}$ *where x_i, $i = 1, \dots, n$ are the bits of x. The verification procedure checks the signature's consistency with the public key, i.e.,* $\mathsf{Ver}_{\mathsf{pk}}(x, s) = 0$ *if* $p_i^{x_i} = h_k(s_i)$ *and* $\mathsf{Ver}_{\mathsf{pk}}(x, s) = 0$ *otherwise.*

We now show that the Lamport scheme is 1-BU secure in the quantum random oracle model.

Theorem 14. *Construction 2 is 1-*BU *secure if h is modeled as a quantum-accessible random oracle.*

Proof. We implement the random oracle h as a superposition oracle with register F. In the 1-$\mathsf{BlindForge}$ experiment we execute the sampling part of the key generation by preparing a superposition as well. More precisely, we can just prepare $2n$ n-qubit registers S_i^j in a uniform superposition, with the intention of measuring them to sample s_i^j in mind. We are talking about a classical one-time signature scheme, and all computation that uses the secret key is done by an honest party, and is therefore classical. It follows that the measurement that samples s_i^j commutes with all other operations which are implemented as quantum-controlled operations controlled on the secret key registers, i.e., we can postpone it to the very end of the 1-$\mathsf{BlindForge}$ experiment, just like the measurement that samples an actual random oracle using a superposition oracle. The joint state $|\psi_0\rangle$ with oracle register F and secret key register $SK = (S_i^j)_{i\in\{1,\dots,n\},j=0,1}$ is now in a uniform superposition, i.e.,

$$|\psi_0\rangle_{SKF} = |\phi_0\rangle_{SK}^{\otimes 2n} \otimes |\phi_0\rangle_F^{\otimes 2^n}. \tag{26}$$

To subsequently generate the public key, the superposition oracle for h is queried on each of the S_i^j with an empty outrput register P_i^j, producing the state $|\psi_1\rangle_{SKPKF}$ equal to

$$2^{-2n^2} \sum_{\substack{s_i^j\in\{0,1\}^n\\ p_i^j\in\{0,1\}^n\\ i\in\{1,\dots,n\},j=0,1}} \left(\bigotimes_{\substack{i\in\{1,\dots,n\}\\ j=0,1}} |s_i^j\rangle_{S_i^j}\right) \otimes \left(\bigotimes_{\substack{i\in\{1,\dots,n\}\\ j=0,1}} |p_i^j\rangle_{P_i^j}\right) \otimes |f_{\mathsf{sk},\mathsf{pk}}\rangle_F,$$

where $|f_{\mathsf{sk},\mathsf{pk}}\rangle_F$ is the superposition oracle state where $F_{s_i^j}$ is in state $|p_i^j\rangle$ and all other registers are still in state $|\phi_0\rangle$. Then the registers P_i^j are measured to produce an actual, classical, public key that can be handed to the adversary. Note that there is no hash function key k now, as it has been replaced by the random oracle. Treating the public key as classical information from now on and removing the registers PK, the state takes the form

$$|\psi_2(\mathsf{pk})\rangle_{SKF} = 2^{-n^2} \sum_{\substack{s_i^j\in\{0,1\}^n \\ i\in\{1,...,n\},j=0,1}} \left(\bigotimes_{\substack{i\in\{1,...,n\} \\ j=0,1}} |s_i^j\rangle_{S_i^j} \right) \otimes |f_{\mathsf{sk},\mathsf{pk}}\rangle_F. \tag{27}$$

Now the interactive phase of the 1-$\mathsf{BlindForge}$ experiment can begin, and we provide both the random oracle h and the signing oracle (that can be called exactly once) as superposition oracles using the joint oracle state $|\psi_2(\mathsf{pk})\rangle$ above. The random oracle answers queries as described in Section 3.3. The signing oracle, when queried with registers XZ with $Z = Z_1...Z_n$, applies $\mathrm{CNOT}^{\otimes n}_{S_i^{x_i}:Z_i}$, $i = 1, ..., n$ controlled on X being in the state $x \notin B_\varepsilon$.

Now suppose $\mathcal{A}$, after making at most one query to Sign and an arbitrary polynomial number of queries to h, outputs a candidate message signature pair (x^0, z^0) with $z^0 = z_1^0\| \cdots \|z_n^0$. If $x^0 \notin B_\varepsilon$, $\mathcal{A}$ has lost. Suppose therefore that $x^0 \in B_\varepsilon$. We will now make a measurement on the oracle register to find an index i such that $S_i^{x_i^0}$ has not been queried. To this end we first need to decorrelate SK and F. This is easily done, as the success test only needs computational basis measurement results from the register SK, allowing us to perform any controlled operation on F controlled on SK. Therefore we can apply the operation $\bigoplus p_i^j$ followed by $H^{\otimes n}$ to the register $F_{s_i^j}$ controlled on S_i^j being in state $|s_i^j\rangle$, for all $i = 1, ..., n$ and $j = 0, 1$. For an adversary that does not make any queries to h, this has the effect that all F-registers are in state $|\phi_0\rangle$ again now.

We can equivalently perform this restoring procedure before the adversary starts interaction, and answer the adversary's h-queries as follows. Controlled on the adversary's input being equal to one of the parts s_i^j of the secret key, answer with the corresponding public key, otherwise use the superposition oracle for h.

For any fixed secret key register S_i^j, the unitary that is applied upon an h-query can hence be written as

$$U_h' = U_\perp + \sum_{x\in\{0,1\}^n} (U_x - U_\perp)|x\rangle\langle x|_X |x\rangle\langle x|_{S_i^j} \tag{28}$$

$$= U_\perp + \sum_{x\in\{0,1\}^n} |x\rangle\langle x|_X |x\rangle\langle x|_{S_i^j} (U_x - U_\perp), \tag{29}$$

where $U_\perp$ acts trivially on S_i^j and the second equality follows because the unitaries $U_\perp$ and U_x are controlled unitaries with X and S_i^j part of the control register. Using the above equation we derive a bound on the operator norm of the commutator of this unitary and the projector onto $|\phi_0\rangle$,

$$\begin{aligned}\left\| \left[U_h', |\phi_0\rangle\langle\phi_0|_{S_i^j}\right] \right\|_\infty =& 2^{-n/2} \left\| \sum_{x\in\{0,1\}^n} \Big((U_x - U_\perp)|x\rangle\langle x|_X\, |x\rangle\langle\phi_0|_{S_i^j} - |x\rangle\langle x|_X\, |\phi_0\rangle\langle x|_{S_i^j}\, (U_x - U_\perp) \Big) \right\|_\infty \\ =& 2^{-n/2} \max_{x\in\{0,1\}^n} \left\| \Big((U_x - U_\perp)|x\rangle\langle x|_X\, |x\rangle\langle\phi_0|_{S_i^j} - |x\rangle\langle x|_X\, |\phi_0\rangle\langle x|_{S_i^j}\, (U_x - U_\perp) \Big) \right\|_\infty \\ \le& 4 \cdot 2^{-n/2},\end{aligned}$$

where the second equality follows again because $U_\perp$ and U_x are controlled unitaries with X and S_i^j part of the control register.

It follows that a query to h does not decrease the number of registers S_i^j that are in state $|\phi_0\rangle$, except with negligible amplitude.

As we assume that x^0 is blinded, we have that for any message $x \notin B_\varepsilon$, there exists an $i \in \{1, ..., n\}$ such that $x_i \neq x_i^0$. But $\mathcal{A}$ interacts with a blinded signing oracle, i.e., controlled on his input being not blinded, it is forwarded to the signing oracle, otherwise $\perp$ is XORed into his output register. Therefore only non-blinded queries have been forwarded to the actual signing oracle, so the final state is a superposition of states in which the register SK has at least n subregisters S_i^j are in state $|\phi_0\rangle$, and at least one of them is such that $x_i^0 = j$. We can therefore apply an n-outcome measurement to the oracle register to obtain an index i_0 such that $S_{i_0}^{x_{i_0}^0}$ is in state $|\phi_0\rangle$. By Lemma 1, this implies that $\mathcal{A}$'s forgery is independent of s_{i_0}, so $\mathcal{A}$'s probability of succeeding in BlindForge is negligible. □

A simple proof of the PO-security of a random function can be given using a similar idea, see Theorem 20 in the appendix.

6.3 Hash-and-MAC

To authenticate messages of arbitrary length with a fixed-length MAC, it is common practice to first compress a long message by a *collision-resistant* hash functon and then apply the MAC. This is known as Hash-and-MAC. However, when it comes to BU-security, collision-resistance may not be sufficient. We therefore propose a new notion, Bernoulli-preserving hash, generalizing collision-resistance in the quantum setting, and show that it is sufficient for Hash-and-MAC with BU security. Recall that, given a subset B of a set X, $\chi_B : X \to \{0,1\}$ denotes the characteristic function of B.

Definition 8 (Bernoulli-preserving hash)**.** *Let $\mathcal{H} : X \to Y$ be an efficiently computable function family. Define the following distributions on subsets of X:*

1. *$\mathcal{B}_\epsilon$: generate $B_\epsilon \subseteq X$ by placing $x \in B_\epsilon$ independently with probability ϵ. Output B_ϵ.*
2. *$\mathcal{B}_\epsilon^{\mathcal{H}}$: generate $C_\epsilon \subseteq Y$ by placing $y \in C_\epsilon$ independently with probability ϵ. Sample $h \in \mathcal{H}$ and define $B_\epsilon^h := \{x \in X : h(x) \in C_\epsilon\}$. Output B_ϵ^h.*

We say that $\mathcal{H}$ is a Bernoulli-preserving hash if for all adversaries $(\mathcal{A}, \epsilon)$,

$$\left| \Pr_{B \leftarrow \mathcal{B}_\epsilon}[\mathcal{A}^{\chi_B}(1^n) = 1] - \Pr_{B \leftarrow \mathcal{B}_\epsilon^{\mathcal{H}}}[\mathcal{A}^{\chi_B}(1^n) = 1] \right| \leq \mathsf{negl}(n)\,.$$

The motivation for the name Bernoulli-preserving hash is simply that selecting $\mathcal{B}_\epsilon$ can be viewed as a Bernoulli process taking place on the set X, while $\mathcal{B}_\epsilon^h$ can be viewed as the pullback (along h) of a Bernoulli process taking place on Y.

We show that the standard, so-called "Hash-and-MAC" construction will work w.r.t. to BU security, if we instantiate the hash funtion with a Bernoulli-preserving hash. Recall that, given a MAC $\Pi = (\mathsf{Mac}_k, \mathsf{Ver}_k)$ with message set X and a function $h : Z \to X$, there is a MAC $\Pi^h := (\mathsf{Mac}_k^h, \mathsf{Ver}_k^h)$ with message set Z defined by $\mathsf{Mac}_k^h = \mathsf{Mac}_k \circ h$ and $\mathsf{Ver}_k^h(m,t) = \mathsf{Ver}_k(h(m), t)$.

Theorem 15 (Hash-and-MAC with Bernoulli-preserving hash)**.** *Let $\Pi = (\mathsf{Mac}_k, \mathsf{Ver}_k)$ be a BU-secure MAC with $\mathsf{Mac}_k : X \to Y$, and let $h : Z \to X$ a Bernoulli-preserving hash. Then Π^h is a BU-secure MAC.*

Proof. Let $\mathcal{A}$ be an adversary against Π^h. We build an adversary $\mathcal{A}_0$ against Π which (given oracle $f : X \to Y$) runs $\mathcal{A}$ and answers its queries with $f \circ h$, i.e., $|m\rangle|t\rangle \mapsto |m\rangle|t \oplus f(h(m))\rangle$. This can be implemented by first computing h into an extra register, then invoking the oracle, and then uncomputing h. When $\mathcal{A}$ produces its final output (m, t), $\mathcal{A}_0$ outputs $(h(m), t)$ and terminates. We claim that

$$\left|\Pr[\mathsf{BlindForge}_{\mathcal{A},\Pi^h}(n,\epsilon) = 1] - \Pr[\mathsf{BlindForge}_{\mathcal{A}_0,\Pi}(n,\epsilon) = 1]\right| \leq \mathsf{negl}(n)\,.$$

Since the right-hand-side of the difference above is negligible by BU-security of Π, establishing the claim will finish the proof.

We prove the claim by showing that the difference can be viewed as the success probability of a distinguisher $\mathcal{D}$ against the Bernoulli-preserving property of h. The distinguisher $\mathcal{D}$ receives an oracle for χ_B (where $B \subseteq Z$ is sampled according to either $\mathcal{B}_\epsilon$ or $\mathcal{B}_\epsilon^h$) and proceeds as follows:

1. generate a key k for Π;
2. run $\mathcal{A}$, answering its oracle queries with

$$|m\rangle|t\rangle \mapsto |m\rangle|t\rangle|\chi_B(m)\rangle|\mathsf{Mac}_k(h(m))\rangle \mapsto |m\rangle|t \oplus \chi_B(m) \cdot \mathsf{Mac}_k(h(m))\rangle|\chi_B(m)\rangle$$

where we invoked the oracle in the first step and CCNOT in the second;

3. when $\mathcal{A}$ outputs (m, t), compute $b = \mathsf{Ver}^h_k(m, t) = \mathsf{Ver}_k(h(m), t)$. Query the oracle to compute $b' = \chi_B(m)$, and output $b \wedge b'$.

It now remains to check that (i.) if B was sampled according to $\mathcal{B}_\epsilon$ (i.e., uniform blinding), then $\mathcal{D}$ is simulating the game $\mathsf{BlindForge}_{\mathcal{A},\Pi^h}(n, \epsilon)$, and (ii.) If B was sampled according to $\mathcal{B}^h_\epsilon$ (i.e., hash-blinding), then $\mathcal{D}$ is simulating the game $\mathsf{BlindForge}_{\mathcal{A}_0,\Pi}(n, \epsilon)$. Fact (i.) follows directly from the definition[6] of the $\mathsf{BlindForge}$ game. To see fact (ii.), observe that the $\mathsf{BlindForge}$ game against $\mathcal{A}_0$ samples a uniform blinding set $C_\epsilon \subseteq X$ and executes algorithm $\mathcal{A}$ with oracle

$$m \longmapsto \chi_{C_\epsilon}(h(m)) \cdot \mathsf{Mac}_k(h(m)) = \chi_{B^h_\epsilon}(m) \cdot \mathsf{Mac}_k(h(m))\,,$$

precisely as in the execution of $\mathcal{A}$ by $\mathcal{D}$. □

In the next section, we provide a number of additional results about Bernoulli-preserving hash functions.

7 Properties of Bernoulli-preserving hash functions

We explore the notion of Bernoulli-preserving hash functions. These results can be summarized as follows.

- If H is a random oracle or a qPRF, then it is a Bernoulli-preserving hash.
- If H is $4q$-wise independent, then it is a Bernoulli-preserving hash against q-query adversaries.
- Under the LWE assumption, there is a (public-key) family of Bernoulli-preserving hash functions.
- If we only allow classical oracle access, then the Bernoulli-preserving property is equivalent to standard collision-resistance.
- Bernoulli-preserving hash functions are *collapsing* (another quantum generalization of collision-resistance proposed in [25]).

First, we show that random and pseudorandom functions are Bernoulli-preserving, and that this property is equivalent to collision-resistance against classical queries.

Lemma 4. *Let $H : X \to Y$ be a function such that $1/|Y|$ is negligible. Then:*

1. *If H is a random oracle or a qPRF, then it is a Bernoulli-preserving hash.*
2. *If H is $4q$-wise independent, then it is a Bernoulli-preserving hash against q-query adversaries.*

Proof. The claim for random oracles is obvious: by statistical collision-resistance, uniform blinding is statistically indistinguishable from hash-blinding. The remaining claims follow from the observation that one can simulate one quantum query to $\chi_{B^h_\epsilon}$ using two quantum queries to h (see, e.g., the proof of Corollary 4). □

Theorem 16. *A function $h : \{0,1\}^* \to \{0,1\}^n$ is Bernoulli-preserving against classical-query adversaries if and only if it is collision-resistant.*

[6] Note that we have again used the convention that the blinding symbol $\perp$ is the string $0 \dots 01$; in our case, the final bit corresponds to the register containing $\chi_B(m)$. If one chooses a different convention, it may be necessary to adjust $\mathcal{D}$ to uncompute that register with an extra call to the oracle.

Proof. First, the Bernoulli-preserving hash property implies collision-resistance: testing whether two colliding inputs are either (i.) both not blinded or both blinded, or (ii.) exactly one of them is blinded, yields always outcome (i.) when dealing with a hash-blinded oracle and a uniformly random outcome for a blinded oracle and $\varepsilon = 1/2$. On the other hand, consider an adversary $\mathcal{A}$ that has inverse polynomial distinguishing advantage between blinding and hash-blinding, and let $x_1, ..., x_q$ be it's queries. Assume for contradiction that with overwhelming probability $h(x_i) \neq h(x_j)$ for all $x_i \neq x_j$. Then with that same overwhelming probability the blinded and hash blinded oracles are both blinded independently with probability ε on each x_i and are hence statistically indistinguishable, a contradiction. It follows that with non-negligible probability there exist two queries $x_i \neq x_j$ such that $h(x_i) = h(x_j)$, i.e., $\mathcal{A}$ has found a collision. □

7.1 A Bernoulli-preserving hash from LWE

We have observed that any qPRF is a Bernoulli-preserving hash function, which can be constructed from various quantum-safe computational assumpiton (e.g., LWE). Nonetheless, qPRF typically does not give short digest, which would result in long tags, and it requires a secret key.[7]

Here we point out an alternative construction of a public Bernoulli-preserving hash function based on the quantum security of LWE. In fact, we show that the collapsing hash function by Unruh [24] is also Bernoulli-preserving hash. This constructions relies on a lossy function family $F : X \to Y$ and a universal hash function $G = \{g_k : Y \to Z\}_{k \in \mathcal{K}}$. A lossy function family admits two types of keys: a lossy key $s \leftarrow \mathcal{D}_{\mathrm{los}}$ and an injective key $s \leftarrow \mathcal{D}_{inj}$, which are computationally indistinguishable. $F_s : X \to Y$ under a lossy key s is compressing, i.e., $|\operatorname{im}(F_s)| \ll |Y|$; whereas under an injective key s, F_s is injective. We refer a formal definition to [24, Definition 2], and an explicit construction based on LWE to [18]. We will also use exist efficient constructions for universal hash families [26]. Then one constructs a hash funciton family $H = \{h_{s,k}\}$ by $h_{s,k} := g_k \circ F_s$ with public parameters generated by $s \leftarrow \mathcal{D}_{\mathrm{los}}, k \leftarrow \mathcal{K}$.

The proof of Bernoulli-preserving for this hash function is similar to Unruh's proof that H is collapsing. We begin with a lemma.

Lemma 5. *Any injective function f is Bernoulli-preserving hash. Given any Bernoulli-preserving hash $f : X \to Y$ and $g : Y \to Z$ that is Bernoulli-preserving hash on $\operatorname{im}(f)$, then $h = g \circ f$ is also Bernoulli-preserving hash.*

Proof. The first part follows by observing that a ε-random subset in the codomain corresponds exactly to a ε-random subset in the domain under inverse of the function. Let $\mathcal{O} \approx \mathcal{O}'$ denote that two oracles $\mathcal{O}$ and $\mathcal{O}$ are indistinguishable by any quantum poly-time algorithm. For the second part, we need to show that $\chi_{C:C \leftarrow_\varepsilon X} \approx \chi_{C:C=h^{-1}(C_Z), C_Z \leftarrow_\varepsilon Z}$, where $\leftarrow_\varepsilon$ indicates sampling a random subset of fraction ε. Since f is Bernoulli-preserving hash, we have that

$$\chi_{C:C \leftarrow_\varepsilon X} \approx \chi_{C:C=f^{-1}(C_Y), C_Y \leftarrow_\varepsilon Y} \equiv \chi_{C:C=f^{-1}(C'_Y), C'_Y \leftarrow_\varepsilon \operatorname{im}(f)} \, .$$

The second equivalence holds by observing that for any $C_Y \subseteq Y$, $f^{-1}(C_Y) = f^{-1}(C_Y \cap \operatorname{im}(f))$. Then because g is Bernoulli-preserving on $\operatorname{im}(f)$,

$$\chi_{C'_Y : C'_Y \leftarrow_\varepsilon \operatorname{im}(f)} \approx \chi_{C'_Y : C'_Y = g^{-1}(C_Z), C_Z \leftarrow_\varepsilon Z} \, .$$

Therefore, we conclude that

$$\chi_{C:C \leftarrow_\varepsilon X} \approx \chi_{C:C=f^{-1}(g^{-1}(C_Z)), C_z \leftarrow_\varepsilon Z} = \chi_{C:C=h^{-1}(C_Z), C_z \leftarrow_\varepsilon Z} \, .$$

□

Theorem 17. *H is Bernoulli-preserving hash if LWE holds against any efficient quantum distinguisher.*

Proof. We proceed in three steps (with the help of Lemma 5 above):

[7] In practice, it is probably more convenient (and more reliable) to instantiate a qPRF from block ciphers, which may not be ideal for message authentication.

1) Since F_s is injective under an injective key, it is clearly Bernoulli-preserving hash. As a result, $F_s, s \leftarrow \mathcal{D}_{\text{los}}$ must be Bernoulli-preserving hash too, because a lossy key is indistinguishable from an injective key by definition.

2) Then g_k is chosen properly so that it is injective when restricted to $\text{im}(F_s)$ of lossy key s. Therefore g_k is Bernoulli-preserving hash too.

3) Finally, $H_{k,s}$ is Bernoulli-preserving hash by the composition of Bernoulli-preserving hash functions g_k and F_s.

□

7.2 Relationship to collapsing

Here we relate Bernoulli-preserving hash to the collapsing property, which is another quantum generalization of classical collision-resistance. We first describe the collapsing property (slightly adapting Unruh's original definition [25]) as follows. Let $h : X \to Y$ be a hash function, and let $\mathcal{S}_X$ and $\mathcal{S}_{XY}$ be the set of quantum states (i.e., density operators) on registers corresponding to the sets X and $X \times Y$, respectively. We define two channels from $\mathcal{S}_X$ to $\mathcal{S}_{XY}$. First, $\mathcal{O}_h$ receives X, prepares $|0\rangle$ on Y, applies $|x\rangle|y\rangle \mapsto |x\rangle|y \oplus h(x)\rangle$, and then measures Y fully in the computational basis. Second, $\mathcal{O}'_h$ first applies $\mathcal{O}_h$ and then also measures X fully in the computational basis.

$$
\begin{aligned}
\mathcal{O}_h : \quad & |x\rangle_X \xmapsto{h} |x, h(x)\rangle_{X,Y} \xmapsto{\text{measure } Y} (\rho_X^y, y)\,, \\
\mathcal{O}'_h : \quad & |x\rangle_X \xmapsto{h} |x, h(x)\rangle_{X,Y} \xmapsto{\text{measure } X\&Y} (x, y)\,.
\end{aligned}
$$

If the input is a pure state on X, then the output is either a superposition over a fiber $h^{-1}(s) \times \{s\}$ of h (for $\mathcal{O}_h$) or a classical pair $(x, h(x))$ (for $\mathcal{O}'_h$) .

Definition 9 (Collapsing)**.** *A hash function h is collapsing if for any single-query QPT $\mathcal{A}$, it holds that* $\left|\Pr[\mathcal{A}^{\mathcal{O}_h}(1^n) = 1] - \Pr[\mathcal{A}^{\mathcal{O}'_h}(1^n) = 1]\right| \le \mathsf{negl}(n)\,.$

To prove that Bernoulli-preserving hash implies collapsing, we need a technical fact. Recall that any subset $S \subseteq \{0,1\}^n$ is associated with a two-outcome projective measurement $\{\Pi_S, \mathbb{1} - \Pi_S\}$ on n qubits defined by $\Pi_S = \sum_{x \in S} |x\rangle\langle x|$. We will write Ξ_S for the channel (on n qubits) which applies this measurement.

Lemma 6. *Let $S_1, S_2, \ldots, S_{cn}$ be subsets of $\{0,1\}^n$, each of size 2^{n-1}, chosen independently and uniformly at random. Let Ξ_{S_j} denote the two-outcome measurement defined by S_j, and denote their composition $\tilde{\Xi} := \Xi_{S_{cn}} \circ \Xi_{S_{cn-1}} \circ \cdots \circ \Xi_{S_1}$. Let Ξ denote the full measurement in the computational basis. Then* $\Pr\big[\tilde{\Xi} = \Xi\big] \ge 1 - 2^{-\varepsilon n}$ *, whenever $c \ge 2 + \varepsilon$ with $\varepsilon > 0$,*

Proof. We give a combinatorial proof. Consider an arbitrary mixed state of density matrix $\rho = (\rho_{x,y})_{x,y \in \{0,1\}^n}$, the full measurement Ξ on ρ gives

$$
\Xi(\rho) = \sum_{x \in \{0,1\}^n} |x\rangle\langle x|\, \rho\, |x\rangle\langle x| = \sum_{x \in \{0,1\}^n} \rho_{x,x}\, |x\rangle\langle x|\ .
$$

Given a set $S \subseteq \{0,1\}^n$, the projective measurement Ξ_S on ρ operates as

$$
\begin{aligned}
\Xi_S(\rho) &= \sum_{x,y \in S} |x\rangle\langle x|\, \rho\, |y\rangle\langle y| + \sum_{x,y \notin S} |x\rangle\langle x|\, \rho\, |y\rangle\langle y| \\
&= \sum_{x,y \in S} \rho_{x,y}\, |x\rangle\langle y| + \sum_{x,y \notin S} \rho_{x,y}\, |x\rangle\langle y|\ .
\end{aligned}
$$

Namely, Ξ_S will zero-out the entries $\rho_{x,y}$ in ρ, where $(x \in S, y \notin S)$ or $(x \notin S, y \in S)$. It is easy to verify that the same effect occurs when Ξ and Ξ_S are applied to a subsystem of a bipartite state.

Now, for any $c = 2 + \varepsilon$ with $\varepsilon > 0$, consider sampling $S_1, S_2, \ldots, S_{cn}$ independently at random, each of size 2^{n-1}, and define a few random events:

$$\begin{aligned}
&E^i_{x,y} : x \in S_i \wedge y \in S_i, \text{ or } x \notin S_i \wedge y \notin S_i\,;\\
&E_{x,y} : \forall i \in \{1, \ldots, cn\} \text{ s.t. } E^i_{x,y}\,;\\
&\mathsf{BAD} : \exists x, y \in \{0,1\}^n, x \neq y \text{ s.t. } E_{x,y}\,.
\end{aligned}$$

Observe that if BAD does not occur, it implies that for any $x \neq y$, the off-diagonal entry $\rho_{x,y}$ is eliminated by one of Ξ_{S_i}, and as a result $\tilde{\Xi} = \Xi_{S_{cn}} \circ \ldots \circ \Xi_{S_1}$ will be identical to Ξ.

Fix a pair (x, y) with $x \neq y$, clearly $\Pr[E^i_{x,y}] = 1/2$. Since each S_i is chosen independently,

$$\Pr[E_{x,y}] = \Pi_i \Pr[E^i_{x,y}] = 1/2^{cn}\,.$$

By the union bound,

$$\Pr[\mathsf{BAD}] \leq \binom{2^n}{2} \cdot \Pr[E_{x,y}] \leq 2^{2n}/2^{cn} = 2^{-\varepsilon n}\,.$$

Therefore we conclude that

$$\Pr[\tilde{\Xi} = \Xi] \geq \Pr[\tilde{\Xi} = \Xi \mid \overline{\mathsf{BAD}}] \cdot \Pr[\overline{\mathsf{BAD}}] \geq 1 - 2^{-\varepsilon n}\,. \qquad \square$$

We remark that to efficiently implement each Ξ_S with a random subset S, we can sample $h_i : [M] \to [N]$ from a pairwise-independent hash family (sampling an independent h_i for each i), and then define $x \in S$ iff $h(x) \leq N/2$. For any input state $\sum_{x,z} \alpha_{x,z} |x, z\rangle$, we can compute

$$\sum_{x,z} \alpha_{x,z} |x, z\rangle \mapsto \sum_{x,z} |x, z\rangle |b(x)\rangle, \qquad \text{where } b(x) := h(x) \stackrel{?}{\leq} N/2\,,$$

and then measure $|b(x)\rangle$. Pairwise independence is sufficient by Theorem 8 because only one quantum query is made.

Theorem 18. *If $h : X \to Y$ is Bernoulli-preserving, then it is collapsing.*

Proof. Let $\mathcal{A}$ be an adversary with inverse-polynomial distinguishing power in the collapsing game. Choose n such that $X = \{0,1\}^n$. We define $k = cn$ hybrid oracles $H_0, H_1, \ldots, H_k$, where hybrid H_j is a channel from $\mathcal{S}_X$ to $\mathcal{S}_{XY}$ which acts as follows: (1.) adjoin $|0\rangle_Y$ and apply the unitary $|x\rangle_X|y\rangle_Y \mapsto |x\rangle_X|y \oplus h(x)\rangle_Y$; (2.) measure the Y register in the computational basis; (3.) repeat j times: (i.) select a uniformly random subset $S \subseteq X$ of size 2^{n-1}; (ii.) apply the two-outcome measurement Ξ_S to the X register; (4.) output registers X and Y.

Clearly, H_0 is identical to the $\mathcal{O}_h$ channel in the collapsing game. By Lemma 6, H_k is indistinguishable from the $\mathcal{O}'_h$. By our initial assumption and the triangle inequality, there exists a j such that

$$\left|\Pr[\mathcal{A}^{H_j}(1^n) = 1] - \Pr[\mathcal{A}^{H_{j+1}}(1^n) = 1]\right| \geq 1/\mathsf{poly}(n)\,. \tag{30}$$

We now build a distinguisher $\mathcal{D}$ against the Bernoulli-preserving property (with $\epsilon = 1/2$) of h. It proceeds as follows: (1.) run $\mathcal{A}(1^n)$ and place its query state in register X; (2.) simulate oracle H_j on XY (use 2-wise independent hash to select sets S); (3.) prepare an extra qubit in the $|0\rangle$ state in register W, and invoke the oracle for χ_B on registers X and W; (4.) measure and discard register W; (5.) return XY to $\mathcal{A}$, and output what it outputs.

We now analyze $\mathcal{D}$. After the first two steps of H_j (compute h, measure output register) the state of $\mathcal{A}$ (running as a subroutine of $\mathcal{D}$) is given by

$$\sum_z \sum_{x \in h^{-1}(s)} \alpha_{xz} |x\rangle_X |s\rangle_Y |z\rangle_Z\,.$$

Here Z is a side information register private to $\mathcal{A}$. Applying the j measurements (third step of H_j) results in a state of the form $\sum_z \sum_{x\in M} \beta_{xz}|x\rangle|s\rangle|z\rangle$, where M is a subset of $h^{-1}(s)$. Applying the oracle for χ_B into an extra register now yields

$$\sum_z \sum_{x\in M} \beta_{xz}|x\rangle|s\rangle|z\rangle|\chi_B(x)\rangle_W .$$

Now consider the two cases of the Bernoulli-preserving game.

First, in the "hash-blinded" case, $B = h^{-1}(C)$ for some set $C \subseteq Y$. This implies that $\chi_B(x) = \chi_C(h(x)) = \chi_C(s)$ for all $x \in M$. It follows that W simply contains the classical bit $\chi_C(s)$; computing this bit, measuring it, and discarding it will thus have no effect. The state returned to $\mathcal{A}$ will then be identical to the output of the oracle H_j. Second, in the "uniform blinding" case, B is a random subset of X of size 2^{n-1}, selected uniformly and independently of everything else in the algorithm thus far. Computing the characteristic function of B into an extra qubit and then measuring and discarding that qubit implements the channel Ξ_B, i.e., the measurement $\{\Pi_B, \mathbb{1} - \Pi_B\}$. It follows that the state returned to $\mathcal{A}$ will be identical to the output of oracle H_{j+1}.

By (30), it now follows that $\mathcal{D}$ is a successful distinguisher in the Bernoulli-preserving hash game for h. Hence h is not a Bernoulli-preserving hash. □

8 The problem with PO-unforgeability

Our search for a new definition of unforgeability for quantum-secure authentication is partly motivated by concerns about the PO security notion [6]. In this section, we make these concerns concrete by pointing out a significant security concern not addressed by this definition. Specifically, we demonstrate a MAC which is readily broken with an efficient attack, and yet is PO secure. The attack queries the MAC with a superposition over a particular subset S of the message space, and then forges a valid tag for a message lying outside S.

One of the intuitive issues with PO is that it might rule out adversaries that have to measure, and thereby destroy, one or more post-query states to produce an interesting forgery. Constructing such an example seems not difficult at first. For instance, let us look at one-time PO, and construct a MAC from a qPRF f by sampling a key k for f and a superpolynomially-large prime p, and setting

$$\mathsf{Mac}_{k,p}(m) = \begin{cases} 0^n & \text{if } m = p, \\ (f_k(m \bmod p)) & \text{otherwise.} \end{cases} \tag{31}$$

This MAC is forgeable: a quantum adversary can use a single query to perform period-finding on the MAC, and then forge at 0^n. Intuitively, it seems plausible that the MAC is 1-PO secure as period-finding uses a full measurement. This is incorrect for a somewhat subtle reason: identifying the hidden symmetry does not necessarily consume the post-query state completely, so an adversary can learn the period and a random input-output-pair of the MAC simultaneously. As shown in Lemma 8 in Appendix B.1, this is a special case of a fairly general situation, which makes establishing a proper PO "counterexample" difficult.

8.1 A counterexample to PO

Another intuitive problem with PO is that using the contents of a register can necessitate *uncomputing* the contents of another one. We exploit this insufficiency in the counterexample below. Consider the following MAC construction.

Construction 3. *Given $k = (p, f, g, h)$ where $p \in \{0,1\}^n$ is a random period and $f, g, h : \{0,1\}^n \to \{0,1\}^n$ are random functions, define $M_k : \{0,1\}^{n+1} \to \{0,1\}^{2n}$ by*

$$M_k(x) = \begin{cases} g(x' \bmod p)\|f(x') & x = 1\|x' , \\ 0^n\|h(x') & x = 0\|x',\ x' \neq p , \\ 0^{2n} & x = 0\|p . \end{cases}$$

Consider an adversary that queries as follows

$$\sum_{x,y} |1,x\rangle_X |0^n\rangle_{Y_1} |y\rangle_{Y_2} \longmapsto \sum_{x,y} |1,x\rangle_X |g_p(x)\rangle_{Y_1} |y \oplus f(x)\rangle_{Y_2}\,, \tag{32}$$

and then discards the first qubit and the Y_2 register; this yields $\sum_x |x\rangle|g_p(x)\rangle$. The adversary can extract p via period-finding from polynomially-many such states, and then output $(0\|p, 0^{2n})$. This attack only queries the MAC on messages starting with 1 (e.g., "from Alice"), and then forges at a message which starts with 0 (e.g., "from Bob.") We emphasize that the forgery was never queried, not even with negligible amplitude. It is thus intuitively clear that this MAC does not provide secure authentication. And yet, despite this obvious and intuitive vulnerability, this MAC is in fact PO-secure.

Theorem 19. *The MAC from Construction 3 is* PO*-secure.*

We briefly summarize the proof idea before presenting the details. The superposition oracle technique outlined in Section 3.3 achieves something that naively seems impossible due to the quantum no-cloning theorem: it records on which inputs the adversary has made non-trivial queries.[8] The information recorded in this way cannot, in general be utilized in its entirety—after all, the premise of the superposition oracle is that the measurement $\mathcal{M}_F$ that samples the random function is delayed until after the algorithm has finished, but it still has to be performed. Any measurement $\mathcal{M}'$ that does not commute with $\mathcal{M}_F$ and is performed before $\mathcal{M}_F$, can disturb the outcome of $\mathcal{M}_F$. If however, $\mathcal{M}'$ only has polynomially many possible outcomes, that disturbance is at most inverse polynomial according to Lemma 1.

Here, we sample the random function f using a superposition oracle, and we chose to use a measurement $\mathcal{M}'$ to determine the *number* of nontrivial queries that the adversary has made to f, which is polynomial by assumption. Random functions are PO-secure [6], so the only way to break PO security is to output $(0\|p, 0^{2n})$ and q other input-output-pairs. Querying messages that start with 0 clearly only yields a negligible advantage in guessing p by the Grover lower bound, so we consider an adversary querying only on strings starting with 1. We distinguish two cases, either the adversary makes exactly q non-trivial queries to f, or less than that. In the latter case, the success probability is negligible by the PO-security of f and h. In the former case, we have to analyze the probability that the adversary guesses p correctly. f is not needed for that, so the superposition oracle register can be used to measure the set of q queries that the adversary made. Using an inductive argument reminiscent of the hybrid method [4] we show that this set is almost independent of p, and hence the period is equal to the difference of two of the queried inputs only with negligible probability. But if that is not the case, the periodic version of g is indistinguishable from a random function for that adversary which is independent of p.

Proof. Let $\mathcal{A}$ be an adversary that makes q quantum queries and outputs $q+1$ distinct candidate forgeries (where q is selected by $\mathcal{A}$ at runtime). We let this adversary interact with a mixed oracle, where g, h and p are treated as random variables, and f is represented as a Fourier Oracle as in Section 3.3. We denote the relevant quantum registers as follows. First, the quantum oracle for Mac_k is a unitary operator on four registers: (i.) the $(n+1)$-qubit input register X, (ii.) the n-qubit output register Y_1 into which $g_p : x \mapsto g(x \mod p)$ is computed, and (iii.) the n-qubit output register Y_2 which, if the input x starts with a 1, interacts with the Fourier Oracle, which has (iv.) an $(n \cdot 2^n)$-qubit register denoted by F, with the subregister corresponding to input $x \in \{0,1\}^n$ denoted by F_x. We set $Y = Y_1 Y_2$. Finally, the workspace of $\mathcal{A}$ is a $\mathsf{poly}(n)$-qubit register denoted by E.

By the PO-unforgeability of random functions, any PO-adversary needs to output $(0\|p, 0^{2n})$ when successful, except with negligible probability. Indeed, suppose an adversary $\mathcal{A}$ output $q+1$ input-output pairs of M_k, none of which is equal $(0\|p, 0^{2n})$ with noticeable probability. Then we can use that adversary to construct a PO-adversary against $\tilde{M}_k$ defined as

$$\tilde{M}_k(x) = \begin{cases} f(x') & x = 1\|x'\,, \\ 0^n\|h(x') & x = 0\|x' \end{cases}$$

[8]For the standard unitary oracle for a classical function, a query has no effect when the output register is initialized in the uniform superposition of all strings.

by simulating an M_k-oracle for $\mathcal{A}$ using the oracle for $\tilde{M}_k$. To learn p, an adversary that makes a polynomial number of queries needs to use messages starting with 1, as the lower bound for unstructured search [4] implies that querying messages starting with 0 only provides negligible advantage for learning p. We will thus prove in the following that an adversary whose queries are entirely supported on the space of messages starting with 1 cannot succeed. The proof for a general adversary is similar, if more laborious. We thus omit h from the description in the following, focusing on the task of outputting q input-output pairs of f and the period p.

Let $|\psi\rangle_{XYEF}$ denote the final state of $\mathcal{A}$ and the Fourier Oracle, after the $q+1$ candidate forgeries have been measured, but prior to any other measurements. Recall that each "number projector" P_l from Section 3.3 projects F to the subspace spanned by basis states with exactly l non-zero entries. We apply to $|\psi\rangle$ the two-outcome measurement defined by $P_{<q} = \sum_{l=0}^{q-1} P_l$ and its complementary projector $P_{\geq q} = \mathbb{1} - P_{<q}$, effectively measuring whether F contains fewer than q non-zero entries (i.e., registers F_x containing a state other than 0^n); note that it cannot contain more than q by Lemma 2. By Lemma 1, applying this measurement decreases the success probability of $\mathcal{A}$ at any particular task by a factor 1/2. We handle the two possible outcomes ($< q$ and q) separately.

Case $< q$: Let $|\psi^{<q}\rangle_{XYEF} := P_{<q}|\psi\rangle_{XYEF}$ be the post-measurement state. Note that $P_l|\psi^{<q}\rangle = 0$ for all $l \geq q$, i.e., each basis component of $|\psi^{<q}\rangle$ has fewer than q non-zero entries in F. On the other hand, the output of $\mathcal{A}$ contains at least q candidate input-output pairs (x_i, y_i) of f (since $(0\|p, 0^{2n})$ is the only input-output pair of Mac_k that does not also contain an input-output pair of a random function). We apply the q-outcome measurement to F which asks: "among the registers $\{F_{x_i}\}_{i=1}^{q}$, which is the first one to contain 0^n?" This measurement is defined by projectors

$$\Pi_j := \bigotimes_{i=1}^{j} (\mathbb{1} - |0^n\rangle\langle 0^n|)_{F_{x_i}} \otimes |0^n\rangle\langle 0^n|_{F_{x_j}} \,.$$

Adding this measurement to $\mathcal{A}$ ensures that F_{x_j} is in the state 0^n for some j, at the cost of multiplying $\mathcal{A}$'s success probability by $1/q$ (by Lemma 1). Recalling that, in the Fourier Oracle picture, $f(x_j)$ is the result of QFT-ing and then fully measuring F_{x_j}, we see that $f(x_j)$ is now uniformly random and independent of y_j. The original $\mathcal{A}$ (i.e., without the measurement $\{\Pi_j\}_j$) thus succeeded with probability at most $q \cdot 2^{-n}$. [9]

Case q: We will denote the post-measurement state in this case by $|\psi^q_{g_p}\rangle := P_q|\psi\rangle$, emphasizing that the state was produced by interacting with the oracle g_p. By the PO-security of f (Theorem 20) it suffices to show that the correct period p is output by $\mathcal{A}$ (by measuring, say, some designated subregister of E of the state $|\psi^q_{g_p}\rangle$) with at most negligible probability. Since testing success at outputting p does not involve the register F, we are free to apply any quantum channel to the F register of $|\psi^q_{g_p}\rangle$. We choose to measure which q subregisters of F are in a non-zero state. This projective measurement is defined by projectors

$$P_K = \bigotimes_{x\in K} (\mathbb{1} - |0^n\rangle\langle 0^n|)_{F_x} \otimes \bigotimes_{x\notin K} |0^n\rangle\langle 0^n|_{F_x} \qquad \text{and} \qquad P_{\text{rest}} = \mathbb{1} - \sum_K P_K \,, \tag{33}$$

where $K \subset \{0,1\}^n$ with $|K| = q$. Note that $P_{\text{rest}} = \mathbb{1} - P_q$, so the outcome "rest" never occurs for $|\psi^q_{g_p}\rangle$. In the following we denote by $\mathbf{K}$ the random variable obtained from this measurement. We also set some other random variables in boldface to better distinguish them from particular values they can take.

Now consider the preparation of the state $|\psi^q\rangle$ (by $\mathcal{A}$ and the Fourier Oracle) with an arbitrary choice of oracle function $h : \{0,1\}^n \to \{0,1\}^n$ in place of g_p. We will denote this state by $|\psi^q_h\rangle$. We now show that, conditioned on a particular measurement outcome K, we can arbitrarily relabel the values of h outside K, without affecting the output state of the algorithm.

Lemma 7. *Let $K \subset \{0,1\}^n$ with $|K| = q$ and $h, h' : \{0,1\}^n \to \{0,1\}^n$ a pair of functions satisfying $h(x) = h'(x)$ for all $x \in K$. Then $P_K|\psi^q_h\rangle = P_K|\psi^q_{h'}\rangle$.*

[9]This argument amounts to an alternative proof that random functions are PO-secure.

Proof. Let $W^{(j)}_{XYEF} := V^{(j)}_{XYE} U^{(h)}_{XY_1} U^{\mathrm{FO}}_{XY_2F}$, where $V^{(j)}$ is $\mathcal{A}$'s j-th internal unitary, $U^{(h)}$ is the standard oracle unitary for h, and U^{FO} is the Fourier Oracle unitary as described in Section 3.3. The intermediate states are

$$|\varphi_{h,k}\rangle_{XYEF} := W^{(k)} \cdots W^{(1)} V^{(0)} |0\rangle_{XYEF} , \tag{34}$$

and the final state is $|\psi_h\rangle := |\varphi_{h,q}\rangle$. By Lemma 2, $P_l|\varphi_{k,h}\rangle = 0$ for all $l > k$, so

$$|\psi^q_h\rangle = P_q|\psi_h\rangle = P_q W^{(q)} \cdots W^{(k+1)} |\varphi_{k,h}\rangle = \sum_{l=0}^{k} P_q W^{(q)} \cdots W^{(k+1)} P_l |\varphi_{k,h}\rangle .$$

For the l term in the sum above, the unitary applies $q-k$ queries to $P_l|\varphi_{k,h}\rangle$; by Lemma 2 this term is thus zero unless $l = k$. We can therefore insert a P_k after the k-th query for free when projecting with P_q in the end. Explicitly,

$$|\psi^q_h\rangle = P_q W^{(q)} P_{q-1} W^{(q-1)} P_{q-2} \cdots P_1 W^{(1)} V^{(0)} |0\rangle_{XYEF} . \tag{35}$$

We first show that we can apply

$$\tilde{P}_K := \bigotimes_{x\in K} \mathbb{1}_{F_x} \otimes \bigotimes_{x \in K^c} |0^n\rangle\langle 0^n|_{F_x}$$

after every query of $\mathcal{A}$.

We are interested in the state $P_K|\psi\rangle_{XYEF} = P_K P_q |\psi\rangle_{XYEF}$. We can make a similar argument as above to show that we can project with $\tilde{P}_K$ after every query as well. As the FO-unitary is the only one that acts on F, and because $\tilde{P}_K|0\rangle^{\otimes n2^n} = |0\rangle^{\otimes n2^n}$, we can even apply the projector $\tilde{P}_K$ before and after each query. We write $N = N_K + N_{K^c}$, where

$$N_K = \sum_{x\in K} (\mathbb{1} - |0\rangle\langle 0|)_{F_x} \otimes \mathbb{1}^{\otimes (2^n-1)}, \tag{36}$$

i.e., N_K and N_{K^c} measure the number of non-zero entries inside and outside K, respectively. Lemma 2 applies to N_K and N_{K^c} separately, and $P_K N_K |\psi\rangle_{XYEF} = N_K P_K |\psi\rangle_{XYEF} = q P_K |\psi\rangle_{XYEF}$. Therefore we have, defining

$$U_{>k} = V^{(q)}_{XYE} U^{(h)}_{XY_1} U^{\mathrm{FO}}_{XY_2F} V^{(q-1)}_{XYE} U^{(h)}_{XY_1} U^{\mathrm{FO}}_{XY_2F} ... V^{(k+1)}_{XYE} U^{(h)}_{XY_1} U^{\mathrm{FO}}_{XY_2F} V^{(k)}_{XYE} \tag{37}$$

and using the same argument as above, that

$$P_K U_{>k} N |\psi^k\rangle = P_K U_{>k} N_K |\psi^k\rangle = k P_K U_{>k} |\psi^k\rangle, \tag{38}$$

and hence

$$P_K U_{>k} N_{K^c} |\psi^k\rangle = P_K U_{>k} N |\psi^k\rangle - P_K U_{>k} N_K |\psi^k\rangle = 0, \tag{39}$$

implying $N_{K^c}|\psi^k\rangle = 0$. But the projector onto the zero-eigenspace of N_{K^c} is $\tilde{P}_K$, so $\tilde{P}_K|\psi^k\rangle = |\psi^k\rangle$.

With an even simpler argument we can insert a projector $P^{\neq 0}_{Y_2} = (\mathbb{1} - |0\rangle\langle 0|)_{Y_2}$ before every query. This is because $U^{\mathrm{FO}}|0\rangle_{Y_2}|\gamma\rangle_{XF} = |0\rangle_{Y_2}|\gamma\rangle_{XF}$, and therefore the number operator eignenvalue does not increase.

To show that $U^{(h)} \tilde{P}_K U^{\mathrm{FO}} \left(P^{\neq 0}_{Y_2} \otimes (\tilde{P}_K)_F\right)$ is independent of the values outside K, we observe that for all $x \notin K$, $y \in \{0,1\}^n \setminus \{0^n\}$ and for all states $|\gamma\rangle_{Y_1EF}$, we have

$$\begin{aligned}
&U^{(g,p)} \tilde{P}_K U^{\mathrm{FO}} \left(P^{\neq 0}_{Y_2} \otimes (\tilde{P}_K)_F\right) |x\rangle_X \otimes |\phi_y\rangle_{Y_2} \otimes |\gamma\rangle_{Y_1EF} \\
&= U^{(g,p)} |x\rangle_X \otimes \left(\tilde{P}_K \left(H^{\otimes n}\right)_{Y_2} \mathrm{CNOT}_{Y_2:F_x} |y\rangle_{Y_2} \otimes \tilde{P}_K |\gamma\rangle_{Y_1EF} \right) \\
&= U^{(g,p)} |x\rangle_X \otimes \left(\tilde{P}_K |0\rangle\langle 0|_{F_x} \left(H^{\otimes n}\right)_{Y_2} \mathrm{CNOT}_{Y_2:F_x} |0\rangle\langle 0|_{F_x} |y\rangle_{Y_2} \otimes \tilde{P}_K |\gamma\rangle_{Y_1EF} \right) \\
&= U^{(g,p)} |x\rangle_X \otimes \left(\tilde{P}_K |0\rangle\langle 0|_{F_x} |y\rangle\langle 0|_{F_x} \otimes |\phi_y\rangle_{Y_2} \otimes \tilde{P}_K |\gamma\rangle_{Y_1EF} \right) \\
&= 0,
\end{aligned} \tag{40}$$

where we have used that for all $x \notin K$ it holds that $|0\rangle\langle 0|_{F_x} \tilde{P}_K = \tilde{P}_K$. This implies that our artificial oracle $U^{(g,p)} \tilde{P}_K U^{\mathrm{FO}} \left(P^{\neq 0}_{Y_2} \otimes (\tilde{P}_K)_F \right)$ (together with a renormalization) only gives $\mathcal{A}$ access to $g(x \mod p)$ for inputs $x \in K$.

This concludes the proof of Lemma 7. $\square$

We now continue with the "**case** q" proof of the theorem. We bound $\mathcal{A}$'s success probability separately for each outcome K. Indeed, it suffices to show that for all $K \subset \{0,1\}^n$, $|K| = q$ the probability that the output contains a pair $(0\|p, 0^{2n})$ is negligible if $\mathcal{A}$ continues with

$$|\psi^{q,K}\rangle := \frac{P_K|\psi^q\rangle}{\|P_K|\psi^q\rangle\|_2} \tag{41}$$

in place of $|\psi\rangle$.

We show that the periodic oracle can be replaced by a non-periodic one, except with negligible probability. More precisely, if p' is $\mathcal{A}$'s output, there exists an event E such that $\Pr[E] = 1 - \mathsf{negl}(n)$ and $\Pr[p' = p_0 | E, p = p_0] = \Pr[p' = p_0 | E, p = 0]$ for all $p_0 \in \{0,1\}^n$. In the following, let us denote the oracle for the MAC of Construction 1 with functions f and g and period p by $\mathcal{O}_{f,g_p}$. We define

$$\mathcal{P}^{\mathrm{bad}}_K = \left\{ p \in \{0,1\}^n \middle| \exists x, x' \in K : p | x - x' \right\}. \tag{42}$$

For $K \subset \{0,1\}^n$ and $p \in \{0,1\}^n$, if $p \notin \mathcal{P}^{\mathrm{bad}}_K$, let $T_{K,p} \subset \{0,1\}^n$ be a transversal for p (i.e., a maximal set such that for $x, y \in T_{K,p}$ it holds that $x \neq y \mod p$) such that $T_{K,p} \cap K = K$. Using this transversal, we can define for each K a random periodic function $g_p^{(K)}$ that is identically distributed with g_p, as follows.

- If $p \in \mathcal{P}^{\mathrm{bad}}_K$, we set $g_p^{(K)}(x) = g(x \mod p)$.
- If $p \notin \mathcal{P}^{\mathrm{bad}}_K$, we set $g_p^{(K)}(x) = g(y)$ for $y \in T_{K,p}$ such that $x = y \mod p$.

For a unitary algorithm $\tilde{\mathcal{A}}$ that makes ℓ queries to an oracle $\mathcal{O}_{f,g_p}$, we define the following procedures:

Procedure 0

1. Sample f, g and p.
2. Run $\tilde{\mathcal{A}}$ with oracle $\mathcal{O}_{f,g_p}$ resulting in a final adversary-oracle state $|\hat{\psi}\rangle$. Apply the measurement $\{P_{\geq \ell}, P_{<\ell}\}$ to F. If outcome is $< \ell$, output "fail."
3. Measure K. If $p \in \mathcal{P}^{\mathrm{bad}}_K$, output "bad." Otherwise, let $|\psi\rangle$ be the post-measurement state of adversary and oracle, i.e., $|\psi\rangle = P_K P_{\geq \ell} |\hat{\psi}\rangle = P_K |\hat{\psi}\rangle$.
4. Output $(K, p, |\psi\rangle)$.

Procedure $\mathbf{0_K}$

Same as Procedure 0, except with oracle $\mathcal{O}_{f,g_p^{(K)}}$ instead of $\mathcal{O}_{f,g_p}$.

Procedure 1

1. Sample f and g.
2. Run $\mathcal{A}$ with an oracle $\mathcal{O}_{f,g_0}$ resulting in a final adversary-oracle state $|\hat{\psi}\rangle$. Apply the measurement $\{P_{\geq \ell}, P_{<\ell}\}$ to F. If outcome is $< \ell$, output "fail."
3. Measure K and sample p. If $p \in \mathcal{P}^{\mathrm{bad}}_K$, output "bad." Otherwise, let $|\psi\rangle$ be the post-measurement state of adversary and oracle, i.e., $|\psi\rangle = P_K P_{\geq \ell} |\hat{\psi}\rangle = P_K |\hat{\psi}\rangle$.
4. Output $(K, p, |\psi\rangle)$.

We first observe that for all K, the outputs of procedures 0 and 0_K are identically distributed because g_p and $g_{p,K}$ are. Note that for any fixed K, $P_K P_q = P_K$; this, together with Lemma 7, implies that

$$\Pr\left[(K, p, |\psi\rangle) \leftarrow \text{Procedure } 0_K\right] = \Pr\left[(K, p, |\psi\rangle) \leftarrow \text{Procedure 1}\right]. \tag{43}$$

It follows that, still for a fixed K,

$$\Pr\left[(K, p, |\psi\rangle) \leftarrow \text{Procedure 0}\right] = \Pr\left[(K, p, |\psi\rangle) \leftarrow \text{Procedure 1}\right]. \tag{44}$$

This implies also that in any of the three procedures, conditioned on the event that the output is neither "fail" nor "bad" and on a fixed first output K, p is uniformly distributed on $\{0,1\}^n \setminus \mathcal{P}_{\text{bad}}$. In other words,

$$\Pr\left[\mathbf{p} = p \,|\, \mathbf{K} = K \wedge \mathbf{p} \notin \mathcal{P}_K^{\text{bad}}\right] = \begin{cases} \left(2^n - |\mathcal{P}_K^{\text{bad}}|\right)^{-1} & p \notin \mathcal{P}_K^{\text{bad}}, \\ 0 & \text{else.} \end{cases} \tag{45}$$

Let us denote the event that a procedure outputs a triple $(K, p, |\psi\rangle)$ by "good."

In what follows, we fix a particular period p, an outcome of the period-sampling step (step 1 in Procedures 0 and 0_K and step 3 in Procedure 1). Given a number ℓ of queries we identify three subspaces of $\mathcal{H}_F$ corresponding to the three outcomes "good," "bad" and "fail" of the procedures above:

$$S_{\text{fail}}^{\ell} = \text{range}(P_{<\ell}), \tag{46}$$

$$S_{\text{bad}}^{\ell} = \text{span}\left\{\text{range}\left(P_K\right) \middle| K \subset \{0,1\}^n,\ |K| = \ell,\ \exists x, y \in K : p|x - y\right\}, \text{ and} \tag{47}$$

$$S_{\text{good}}^{\ell} = \left(S_{\text{fail}}^{\ell}\right)^{\perp} \cap \left(S_{\text{bad}}^{\ell}\right)^{\perp}. \tag{48}$$

We emphasize that the decomposition defined by these subspaces depends on the aforementioned period p. We let P_i^{ℓ} for $i \in \{\text{good}, \text{bad}, \text{fail}\}$ denote the projectors onto these subsets.

By the above reasoning we know that for any algorithm that makes ℓ queries to an oracle $\mathcal{O}$ and has final state $|\psi_{\mathcal{O}}^{\ell}\rangle_{AF}$, it holds that $P_{\text{good}}^{\ell}|\psi_{\mathcal{O}_{f,g_p}}^{\ell}\rangle_{AF} = P_{\text{good}}^{\ell}|\psi_{\mathcal{O}_{f,g_0}}^{\ell}\rangle_{AF}$. It is easy to see that when another query is made, i.e., the $\ell+1$st query of some algorithm, some transitions from S_i^{ℓ} to $S_j^{\ell+1,p}$ are impossible. We only need one impossibility, namely that according to Lemma 2, $P_i^{\ell+1} U^{\text{FO}} P_{\text{fail}}^{\ell} = 0$ for all $i \neq \text{fail}$. In words, once an adversary has fallen behind his q-query plan of making one non-trivial query to f in every query, he can never catch up. Also note that for $\ell = 0$, $S_{\text{fail}}^{\ell} = S_{\text{bad}}^{\ell} = 0$. It is now easy to show by induction that for a q-query adversary $\mathcal{A}$ with final adversary-oracle state $|\phi\rangle$ it holds that

$$\|P_{\text{bad}}^{q}|\phi\rangle\|_2 \leq \sum_{\ell=1}^{q} \left\|P_{\text{bad}}^{\ell} U^{\text{FO}} P_{\text{good}}^{\ell-1}|\phi_\ell\rangle\right\|_2, \tag{49}$$

where $|\phi_\ell\rangle$ is the adversary oracle state before the ℓth query. The induction step is proven as follows. Assume the above formula is true for q. Then we have for a $(q+1)$-query adversary $\mathcal{A}$ with final adversary-oracle state $|\phi\rangle$

$$\begin{aligned}
\left\|P_{\text{bad}}^{q+1}|\phi\rangle\right\|_2 &= \left\|P_{\text{bad}}^{q+1}|\psi_{q+1}\rangle\right\|_2 &(50)\\
&\leq \left\|P_{\text{bad}}^{q+1} U^{\text{FO}} P_{\text{good}}^{q}|\phi_{q+1}\rangle\right\|_2 + \left\|P_{\text{bad}}^{q+1} U^{\text{FO}} P_{\text{bad}}^{q}|\phi_{q+1}\rangle\right\|_2 &(51)\\
&\quad + \left\|P_{\text{bad}}^{q+1} U^{\text{FO}} P_{\text{fail}}^{q}|\phi_{q+1}\rangle\right\|_2 &(52)\\
&= \left\|P_{\text{bad}}^{q+1} U^{\text{FO}} P_{\text{good}}^{q}|\phi_{q+1}\rangle\right\|_2 + \left\|P_{\text{bad}}^{q+1} U^{\text{FO}} P_{\text{bad}}^{q}|\phi_{q+1}\rangle\right\|_2 &(53)\\
&\leq \left\|P_{\text{bad}}^{q+1} U^{\text{FO}} P_{\text{good}}^{q}|\phi_{q+1}\rangle\right\|_2 + \left\|U^{\text{FO}} P_{\text{bad}}^{q}|\phi_{q+1}\rangle\right\|_2 &(54)\\
&= \left\|P_{\text{bad}}^{q+1} U^{\text{FO}} P_{\text{good}}^{q}|\phi_{q+1}\rangle\right\|_2 + \|P_{\text{bad}}^{q}|\psi_q\rangle\|_2. &(55)
\end{aligned}$$

Here we have used the unitary invariance of the Euclidean together with the observation that the state $|\phi\rangle$ is obtained from the state $|\psi_{q+1}\rangle$ right after the $(q+1)$-st query of $\mathcal{A}$ by a unitary acting on the adversary's space only and which therefore commutes with P^q_{bad} in the first, the triangle inequality in the second line, the observation that $P_i^{\ell+1}U^{\mathrm{FO}}P^\ell_{\mathrm{fail}}=0$ in the third line, and the fact that $\|P\|_\infty \leq 1$ for any projector P in the fourth line. In the fifth line we use the same argument as in the first line, just for $|\phi_{q+1}\rangle$ and $|\psi_q\rangle$. This proves Equation (49).

It remains to bound

$$\left\|P^\ell_{\mathrm{bad}}U^{\mathrm{FO}}P^{\ell-1}_{\mathrm{good}}|\phi_\ell\rangle\right\|_2 .$$

To this end, suppose that we measure the X-register of $|\phi_\ell\rangle$ in the computational basis with outcome $\mathbf{X}_\ell$, as well as $\mathbf{K}^{(\ell-1)}$ the set of nonzero registers in F. According to Equations (44) and (45), we have that $\mathbf{X}_\ell$ and $\mathbf{p}$ are independent and $\mathbf{p}$ is uniformly distributed on $\{0,1\}^n\setminus\mathcal{P}^{\mathrm{bad}}_K$ conditioned on $\mathbf{p}\notin\mathcal{P}^{\mathrm{bad}}_K$ and $\mathbf{K}=K$ for a fixed $(\ell-1)$-element set K. It follows that

$$\Pr\left[\mathbf{p}\in\mathcal{P}^{\mathrm{bad}}_{\mathbf{K}\cup\{\mathbf{X}_\ell\}}\middle|\mathbf{K}=K\wedge\mathbf{p}\notin P^{\mathrm{bad}}_K\right] \tag{56}$$

$$=\Pr\left[\exists y\in K:\mathbf{p}|(\mathbf{X}_\ell-y)\middle|\mathbf{K}=K\wedge\mathbf{p}\notin P^{\mathrm{bad}}_K\right] \tag{57}$$

$$\leq\frac{(\ell-1)2^{\frac{c'n}{\log n}}}{2^n-\frac{(\ell-1)(\ell-2)}{2}2^{\frac{c'n}{\log n}}}\leq(\ell-1)2^{-n\left(1-\frac{c}{\log n}\right)} . \tag{58}$$

Here the last inequality holds for some $0<c<c'$ and large enough n, and we have used in the third line that there exists a constant $c'>0$ such that the number of divisors of an integer M is bounded by $2^{c\frac{\log M}{\log\log M}}$ which also implies

$$\left|\mathcal{P}^{\mathrm{bad}}_K\right|\leq\frac{(\ell-1)(\ell-2)}{2}2^{c\frac{n}{\log n}} \tag{59}$$

for all $K\subset\{0,1\}^n$, $|K|=\ell$. We would now like to relate the above probability to

$$\mathbb{E}\left[\left\|P^\ell_{\mathrm{bad}}U^{\mathrm{FO}}P^{\ell-1}_{\mathrm{good}}|\phi_\ell\rangle\right\|_2^2\right] .$$

To this end we analyze how the operator $P^\ell_{\mathrm{bad}}U^{\mathrm{FO}}P^{\ell-1}_{\mathrm{good}}$ behaves on states of the form $|x\rangle_X\otimes|\phi_y\rangle\otimes|\zeta\rangle_{EF}$ such that $(P_K)_F|\zeta\rangle_{EF}=|\zeta\rangle_{EF}$ for some fixed $K\not\ni x$ and $p\in\{0,1\}^n$ such that $p\notin\mathcal{P}^{\mathrm{bad}}_K$. We calculate

$$U^{\mathrm{FO}}P^{\ell-1}_{\mathrm{good}}|x\rangle_X\otimes|\phi_y\rangle\otimes|\zeta\rangle_{EF} \tag{60}$$

$$=U^{\mathrm{FO}}|x\rangle_X\otimes|\phi_y\rangle\otimes|\zeta\rangle_{EF_K}\otimes|0^{n(2^n-\ell+1)}\rangle_{F_{K^c}} \tag{61}$$

$$=\left(H^{\otimes n}\right)_Y\mathrm{CNOT}_{Y:F_x}|x\rangle_X\otimes|y\rangle\otimes|\zeta\rangle_{EF_K}\otimes|0^{n(2^n-\ell+1)}\rangle_{F_{K^c}} \tag{62}$$

$$=\left(H^{\otimes n}\right)_Y|x\rangle_X\otimes|y\rangle\otimes|\zeta\rangle_{EF_K}\otimes|y\rangle_{F_x}\otimes|0^{n(2^n-\ell)}\rangle_{F_{(K\cup\{x\})^c}} \tag{63}$$

$$=|x\rangle_X\otimes|\phi_y\rangle\otimes|\zeta\rangle_{EF_K}\otimes|y\rangle_{F_x}\otimes|0^{n(2^n-\ell)}\rangle_{F_{(K\cup\{x\})^c}} . \tag{64}$$

In the first equation we have use the assumptions that $(P_K)_F|\zeta\rangle_{EF}=|\zeta\rangle_{EF}$ and $p\notin\mathcal{P}^{\mathrm{bad}}_K$; the rest of the calculation is analogous to Equation (40). This implies that

$$P_{K\cup\{x\}}U^{\mathrm{FO}}P^{\ell-1}_{\mathrm{good}}|x\rangle_X\otimes|\phi_y\rangle\otimes|\zeta\rangle_{EF}=U^{\mathrm{FO}}P^{\ell-1}_{\mathrm{good}}|x\rangle_X\otimes|\phi_y\rangle\otimes|\zeta\rangle_{EF} \tag{65}$$

and therefore

$$P^\ell_{\mathrm{bad}}U^{\mathrm{FO}}P^{\ell-1}_{\mathrm{good}}|x\rangle_X\otimes|\phi_y\rangle\otimes|\zeta\rangle_{EF} \tag{66}$$

$$=\begin{cases}U^{\mathrm{FO}}P^{\ell-1}_{\mathrm{good}}|x\rangle_X\otimes|\phi_y\rangle\otimes|\zeta\rangle_{EF} & \text{if }\exists x'\in K:p|(x-x') ,\\ 0 & \text{otherwise.}\end{cases} \tag{67}$$

We therefore calculate for a fixed p,

$$\begin{aligned}
\left\|P^{\ell}_{\mathrm{bad}}U^{\mathrm{FO}}P^{\ell-1}_{\mathrm{good}}|\phi_\ell\rangle\right\|_2^2 &= \left\|\sum_{\substack{K\subset\{0,1\}^n\\|K|=\ell-1\\p\notin\mathcal{P}^{\mathrm{bad}}_K}}\sum_{x\in\{0,1\}^n}P^{\ell}_{\mathrm{bad}}U^{FO}\left(|x\rangle\langle x|_X\otimes P_K\right)|\phi_\ell\rangle\right\|_2^2\\
&=\left\|U^{\mathrm{FO}}\sum_{\substack{K\subset\{0,1\}^n\\|K|=\ell-1\\p\notin\mathcal{P}^{\mathrm{bad}}_K}}\sum_{\substack{x\in\{0,1\}^n\setminus K\\\exists x'\in K:p|(x-x')}}\left(|x\rangle\langle x|_X\otimes P_K\right)|\phi_\ell\rangle\right\|_2^2\\
&=\sum_{\substack{K\subset\{0,1\}^n\\|K|=\ell-1\\p\notin\mathcal{P}^{\mathrm{bad}}_K}}\sum_{\substack{x\in\{0,1\}^n\setminus K\\\exists x'\in K:p|(x-x')}}\left\|\left(|x\rangle\langle x|_X\otimes P_K\right)|\phi_\ell\rangle\right\|_2^2\\
&=\Pr\left[p\notin\mathcal{P}^{\mathrm{bad}}_{\mathbf{K}}\wedge p\in\mathcal{P}^{\mathrm{bad}}_{\mathbf{K}\cup\{\mathbf{X}_\ell\}}\middle|\mathbf{p}=p\right].
\end{aligned}$$

Using Equation (58) we can bound

$$\begin{aligned}
&\mathbb{E}_{p\leftarrow\{0,1\}^n}\left[\Pr\left[p\notin\mathcal{P}^{\mathrm{bad}}_{\mathbf{K}}\wedge p\in\mathcal{P}^{\mathrm{bad}}_{\mathbf{K}\cup\{\mathbf{X}_\ell\}}\right]\right]\\
&=\Pr\left[\mathbf{p}\notin\mathcal{P}^{\mathrm{bad}}_{\mathbf{K}}\wedge \mathbf{p}\in\mathcal{P}^{\mathrm{bad}}_{\mathbf{K}\cup\{\mathbf{X}_\ell\}}\right]\\
&=\sum_{K\subset\{0,1\}^n}\Pr\left[\mathbf{p}\notin\mathcal{P}^{\mathrm{bad}}_{K}\wedge \mathbf{p}\in\mathcal{P}^{\mathrm{bad}}_{K\cup\{\mathbf{X}_\ell\}}\middle|\mathbf{K}=K_0\right]\Pr[\mathbf{K}=K_0]\\
&=\sum_{K\subset\{0,1\}^n}\Pr\left[\mathbf{p}\in\mathcal{P}^{\mathrm{bad}}_{K\cup\{\mathbf{X}_\ell\}}\middle|\mathbf{K}=K_0\wedge\mathbf{p}\notin\mathcal{P}^{\mathrm{bad}}_{\mathbf{K}}\right]\Pr\left[\mathbf{p}\notin\mathcal{P}^{\mathrm{bad}}_{K}\wedge\mathbf{K}=K\right]\\
&\leq\Pr\left[\mathbf{p}\notin\mathcal{P}^{\mathrm{bad}}_{\mathbf{K}}\right](\ell-1)2^{-n\left(1-\frac{c}{\log n}\right)}\\
&\leq(\ell-1)2^{-n\left(1-\frac{c}{\log n}\right)}.
\end{aligned}$$

Here we have used Equation (58) in the first inequality. The probability in the first line is taken over a run of the adversary with a fixed period and random g and f, and in the other lines the period is picked uniformly at random from $\{0,1\}^n$ as for a properly generated key in Construction 1. The last two equations together imply

$$\mathbb{E}\left[\left\|P^{\ell}_{\mathrm{bad}}U^{\mathrm{FO}}P^{\ell-1}_{\mathrm{good}}|\phi_\ell\rangle\right\|_2^2\right]\leq(\ell-1)2^{-n\left(1-\frac{c}{\log n}\right)}. \tag{68}$$

Plugging this into Equation (49) yields

$$\begin{aligned}
\Pr\left[\mathbf{p}\in\mathcal{P}^{\mathrm{bad}}_{\mathbf{K}}\right]&=\mathbb{E}\left[\left\|P^{q}_{\mathrm{bad}}|\phi\rangle\right\|_2^2\right]\\
&\leq\mathbb{E}\left[\left(\sum_{i=1}^{q}\left\|P^{\ell}_{\mathrm{bad}}U^{\mathrm{FO}}P^{\ell-1}_{\mathrm{good}}|\phi_\ell\rangle\right\|_2\right)^2\right]\\
&\leq q\sum_{i=1}^{q}\mathbb{E}\left[\left\|P^{\ell}_{\mathrm{bad}}U^{\mathrm{FO}}P^{\ell-1}_{\mathrm{good}}|\phi_\ell\rangle\right\|_2^2\right]\\
&\leq\left(\sum_{\ell=1}^{q}\sqrt{(\ell-1)2^{-n\left(1-\frac{c}{\log n}\right)}}\right)^2\\
&\leq\frac{q^2(q-1)}{2}2^{-n\left(1-\frac{c}{\log n}\right)}
\end{aligned}\tag{69}$$

using the Cauchy-Schwartz inequality in the second line. This finally implies that the adversary's guess p' is equal to p and the measurement $< q$ vs. $\geq q$ returns $\geq q$ with probability at most

$$
\begin{aligned}
&\Pr[\mathbf{p}=\mathbf{p}' \wedge \text{“}\geq q\text{”}] && (70)\\
\leq &\Pr\left[\mathbf{p}\in\mathcal{P}_{\mathbf{K}}^{\mathrm{bad}} \wedge \text{“}\geq q\text{”}\right] + \Pr\left[\mathbf{p}\notin\mathcal{P}_{\mathbf{K}}^{\mathrm{bad}} \wedge \mathbf{p}=\mathbf{p}' \wedge \text{“}\geq q\text{”}\right] && (71)\\
\leq &\Pr\left[\mathbf{p}\in\mathcal{P}_{\mathbf{K}}^{\mathrm{bad}} \wedge \text{“}\geq q\text{”}\right] + \Pr\left[\mathbf{p}=\mathbf{p}' \middle| \mathbf{p}\notin\mathcal{P}_{\mathbf{K}}^{\mathrm{bad}} \wedge \text{“}\geq q\text{”}\right] && (72)\\
\leq &\frac{q^2(q-1)}{2}2^{-n\left(1-\frac{c}{\log n}\right)} + \left(2^n - \frac{(\ell-1)(\ell-2)}{2}2^{\frac{c'n}{\log n}}\right)^{-1} && (73)\\
\leq &\mathsf{negl}(n). && (74)
\end{aligned}
$$

Here we have used Equation (69) and the uniformity of $\mathbf{p}$ conditioned on $\mathbf{p}\notin\mathcal{P}_{\mathbf{K}}^{\mathrm{bad}}$ and $\mathbf{K}=K$ in the last line. $\square$

Remark. It's not hard to see that the MAC from Construction 3 is not GYZ-secure. Indeed, observe that the forging adversary described above queries on messages starting with 0 only, and then forges successfully on a message starting with 1. If the scheme was GYZ secure, then in the accepting case, the portion of this adversary between the query and the final output would have a simulator which leaves the computational basis invariant. Such a simulator cannot change the first bit of the message from 0 to 1, a contradiction.

By Theorem 9, this PO-secure MAC is also not BU-secure.

Corollary 5. *The MAC from Construction 3 is* BU*-insecure.*

References

[1] Scott Aaronson. Quantum lower bound for recursive fourier sampling. *Quantum Information & Computation*, 3(2):165–174, 2003.

[2] Gorjan Alagic, Tommaso Gagliardoni, and Christian Majenz. Can you sign a quantum state? *arXiv preprint arXiv:1811.11858*, 2018.

[3] Gorjan Alagic, Tommaso Gagliardoni, and Christian Majenz. Unforgeable quantum encryption. In Jesper Buus Nielsen and Vincent Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2018*, pages 489–519, Cham, 2018. Springer International Publishing.

[4] Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh V. Vazirani. Strengths and weaknesses of quantum computing. *SIAM J. Comput.*, 26(5):1510–1523, 1997.

[5] Jean-François Biasse and Fang Song. Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '16, pages 893–902, Philadelphia, PA, USA, 2016. Society for Industrial and Applied Mathematics.

[6] Dan Boneh and Mark Zhandry. Quantum-secure message authentication codes. In *Advances in Cryptology - EUROCRYPT 2013*, pages 592–608. Springer, 2013.

[7] Dan Boneh and Mark Zhandry. Secure signatures and chosen ciphertext security in a quantum computing world. In *Advances in Cryptology – CRYPTO 2013*, pages 361–379. Springer, 2013.

[8] Lily Chen, Stephen Jordan, Yi-Kai Liu, Dustin Moody, Rene Peralta, Ray Perlner, and Daniel Smith-Tone. Report on post-quantum cryptography. Technical report, National Institute of Standards and Technology, 2016.

[9] Ronald Cramer, Léo Ducas, Chris Peikert, and Oded Regev. Recovering short generators of principal ideals in cyclotomic rings. In *Advances in Cryptology – EUROCRYPT 2016*, pages 559–585. Springer Berlin Heidelberg, 2016.

[10] Kirsten Eisenträger, Sean Hallgren, Alexei Kitaev, and Fang Song. A quantum algorithm for computing the unit group of an arbitrary degree number field. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, STOC '14, pages 293–302, New York, NY, USA, 2014. ACM.

[11] Tommaso Gagliardoni, Andreas Hülsing, and Christian Schaffner. Semantic security and indistinguishability in the quantum world. In *Advances in Cryptology – CRYPTO 2016*, pages 60–89. Springer, 2016.

[12] Sumegha Garg, Henry Yuen, and Mark Zhandry. New security notions and feasibility results for authentication of quantum data. In *Advances in Cryptology – Crypto 2017*, pages 342–371. Springer, 2017.

[13] Masahito Hayashi. Optimal sequence of quantum measurements in the sense of Stein's lemma in quantum hypothesis testing. *Journal of Physics A: Mathematical and General*, 35(50):10759, 2002.

[14] Marc Kaplan, Gaëtan Leurent, Anthony Leverrier, and María Naya-Plasencia. Breaking symmetric cryptosystems using quantum period finding. In *Advances in Cryptology – CRYPTO 2016*, pages 207–237. Springer, 2016.

[15] Hidenori Kuwakado and Masakatu Morii. Quantum distinguisher between the 3-round Feistel cipher and the random permutation. In *Proceedings of IEEE International Symposium on Information Theory*, pages 2682–2685, June 2010.

[16] Hidenori Kuwakado and Masakatu Morii. Security on the quantum-type Even-Mansour cipher. In *Proceedings of the International Symposium on Information Theory and its Applications*, pages 312–316. IEEE Computer Society, 2012.

[17] Leslie Lamport. Constructing digital signatures from a one way function. Technical Report SRI-CSL-98, SRI International Computer Science Laboratory, 1979.

[18] Chris Peikert and Brent Waters. Lossy trapdoor functions and their applications. In *Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing*, STOC '08, pages 187–196, New York, NY, USA, 2008. ACM.

[19] Amit Sahai and Brent Waters. How to use indistinguishability obfuscation: Deniable encryption, and more. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, STOC '14, pages 475–484. ACM, 2014.

[20] Thomas Santoli and Christian Schaffner. Using Simon's algorithm to attack symmetric-key cryptographic primitives. *Quantum Information & Computation*, 17(1&2):65–78, 2017.

[21] Peter W Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM journal on computing*, 26(5):1484–1509, 1997.

[22] Fang Song and Aaram Yun. Quantum security of NMAC and related constructions - PRF domain extension against quantum attacks. In *Advances in Cryptology - CRYPTO 2017*, pages 283–309. Springer, 2017.

[23] W Forrest Stinespring. Positive functions on c*-algebras. *Proceedings of the American Mathematical Society*, 6(2):211–216, 1955.

[24] Dominique Unruh. Collapse-binding quantum commitments without random oracles. In *Advances in Cryptology—ASIACRYPT 2016*, volume 10032, pages 166–195. Springer-Verlag New York, Inc., 2016.

[25] Dominique Unruh. Computationally binding quantum commitments. In Marc Fischlin and Jean-Sébastien Coron, editors, *Advances in Cryptology – EUROCRYPT 2016*, pages 497–527, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.

[26] Salil P. Vadhan. Pseudorandomness. *Foundations and Trends® in Theoretical Computer Science*, 7(1–3):1–336, 2012.

[27] Mark Zhandry. How to construct quantum random functions. In *Proceedings of the 53rd Annual Symposium on Foundations of Computer Science*, FOCS '12, pages 679–687, Washington, DC, USA, 2012. IEEE Computer Society.

[28] Mark Zhandry. How to record quantum queries, and applications to quantum indifferentiability. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology – CRYPTO 2019*, pages 239–268, Cham, 2019. Springer International Publishing.

[29] Mark Zhandry. Quantum lightning never strikes the same state twice. In Yuval Ishai and Vincent Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2019*, pages 408–438, Cham, 2019. Springer International Publishing.

A Technical proofs

A.1 The Fourier Oracle number operator

We now restate and prove Lemma 2.

Lemma 2. *The number operator satisfies $\left\|\left[N_F, U^{FO}_{XYF}\right]\right\|_\infty = 1$. In particular, the joint state of a quantum query algorithm and the oracle after the q-th query is in the kernel of P_l for all $l > q$.*

Proof. Let $|\psi\rangle_{XYEF}$ be an arbitrary query state, where X and Y are the query input and output registers, E is the algorithm's internal register and F is the FO register. We expand the state in the computational basis of X,

$$|\psi\rangle_{XYEF} = \sum_{x\in\{0,1\}^n} p(x)|x\rangle_X|\psi_x\rangle_{YEF}. \tag{75}$$

Now observe that

$$U^{\mathrm{FO}}_{XYF}|x\rangle_X|\psi_x\rangle_{YEF} = |x\rangle_X\left(\widetilde{\mathrm{CNOT}}^{\otimes m}\right)_{Y:F_x}|\psi_x\rangle_{YEF}$$

with $\widetilde{\mathrm{CNOT}}_{A:B} = H_A\mathrm{CNOT}_{A:B}H_A$, and therefore

$$\begin{aligned}\Big[N_F, U_{XYF}\Big]|x\rangle_X|\psi_x\rangle_{YEF} &= |x\rangle_X\left[N_F, \left(\widetilde{\mathrm{CNOT}}^{\otimes m}\right)_{Y:F_x}\right]|\psi_x\rangle_{YEF}\\ &= |x\rangle_X\left[(\mathbb{1}-|0\rangle\langle 0|)_{F_x}, \left(\widetilde{\mathrm{CNOT}}^{\otimes m}\right)_{Y:F_x}\right]|\psi_x\rangle_{YEF}.\end{aligned}$$

It follows that

$$\begin{aligned}
&\left\| \left[N_F, U_{XYF} \right] |\psi\rangle_{XYEF} \right\|_2 && (76)\\
&\quad = \sum_{x\in\{0,1\}^n} p(x) \left\| [N_F, U_{XYF}] |\psi_x\rangle_{YEF} \right\|_2 \\
&\quad = \sum_{x\in\{0,1\}^n} p(x) \left\| \left[(\mathbb{1} - |0\rangle\langle 0|)_{F_x}, \left(\widetilde{\mathrm{CNOT}}^{\otimes m} \right)_{Y:F_x} \right] |\psi_x\rangle_{YEF} \right\|_2 \\
&\quad \le \left\| \left[(\mathbb{1} - |0\rangle\langle 0|)_{F_{0^n}}, \left(\widetilde{\mathrm{CNOT}}^{\otimes m} \right)_{Y:F_{0^n}} \right] \right\|_\infty , && (77)
\end{aligned}$$

where we have used the definition of the operator norm and the normalization of $|\psi\rangle_{XYEF}$ in the last line. For a unitary U and a projector P, it is easy to see that $\|[U,P]\|_\infty \le 1$, as $[U,P] = PU(\mathbb{1}-P) - (\mathbb{1}-P)UP$ is a sum of two operators that have orthogonal support and singular values smaller or equal to 1. We therefore get $\|[N_F, U_{XYF}]|\psi\rangle_{XYEF}\|_2 \le 1$, and as the state $|\psi\rangle$ was arbitrary, this implies $\left\|[N_F, U_{XYF}]\right\|_\infty \le 1$. The example from equation (2) shows that the above is actually an equality. The observation that $P_l \eta_F = 0$ for all $l > 0$ and an induction argument proves the second statement of the lemma. ☐

A.2 A simulation theorem: The effect of random blinding

We now restate Theorem 10 and provide a full proof.

Theorem 10. *Let $\mathcal{A}$ be a quantum query algorithm making at most T queries, and $F : \{0,1\}^n \to \{0,1\}^m$ a function. Let $B \subseteq \{0,1\}^n$ be a subset chosen by independently including each element of $\{0,1\}^n$ with probability ϵ, and $P : \{0,1\}^n \to \{0,1\}^m$ be any function with support B. Then*

$$\mathop{\mathbb{E}}_B \left[\delta\left(\mathcal{A}^F(1^n), \mathcal{A}^{F\oplus P}(1^n) \right) \right] \le 2T\sqrt{\epsilon}.$$

Proof. For a function $Q : \{0,1\}^n \to \{0,1\}^m$, we let $\mathcal{O}_Q$ denote the unitary map $|x\rangle|y\rangle \mapsto |x\rangle|y \oplus Q(x)\rangle$. Recall that $\mathcal{A}$ is specified by a fixed initial state $|\phi_0\rangle$ in some finite-dimensional Hilbert space, a sequence of T unitary "computation" operators $C_1, \dots, C_k$, and a POVM $\{P_i : i \in I\}$. The distribution (on I) resulting from the algorithm applied to the oracle $\mathcal{O}_Q$ is given by applying the POVM to the state $|\phi^Q\rangle := C_T \mathcal{O}_Q C_{T-1} \cdots \mathcal{O}_Q C_0 |\phi_0\rangle$. Recall that if the trace distance between two such states satisfies $\delta\big(|\phi^{Q_1}\rangle, |\phi^{Q_2}\rangle\big) := \sqrt{1 - |\langle \phi^{Q_1} | \phi^{Q_2}\rangle|^2} \le \epsilon$ then the distance in total variation between the distributions produced by *any* POVM on these two states is no more than ϵ. In our case, we are interested in controlling $\mathbb{E}_B[\delta(\phi^F, \phi^{P\oplus F})]$. Define $F' = F \oplus P$. In preparation for a standard hybrid argument, define

$$|\phi_k\rangle = \underbrace{C_T \mathcal{O}_{F'} \cdots \mathcal{O}_{F'}}_{(\dagger)} C_k \underbrace{\mathcal{O}_F \dots \mathcal{O}_F C_0}_{(\ddagger)} |\phi_0\rangle \qquad |\phi_k^F\rangle = C_k \underbrace{\mathcal{O}_F \dots \mathcal{O}_F C_0}_{(\ddagger)} |\phi_0\rangle ,$$

so that all oracle invocations in (†) are answered according to $\mathcal{O}_{F'}$ and all those in (‡) are answered according to $\mathcal{O}_F$. Since δ is a metric on pure states, we have

$$\mathbb{E}\, \delta(|\phi^F\rangle, |\phi^{P\oplus F}\rangle) \le \mathbb{E} \sum_{k=1}^{T} \delta(|\phi_k\rangle, |\phi_{k-1}\rangle) = \sum_{k=1}^{T} \mathbb{E}\, \delta(|\phi_k\rangle, |\phi_{k-1}\rangle) .$$

Note that δ is invariant under (simultaneous) unitary action, and hence for any F, B, and P,

$$\begin{aligned}
&\delta(|\phi_k\rangle, |\phi_{k-1}\rangle) \\
&= \delta(C_T \mathcal{O}_{F'} \cdots \mathcal{O}_{F'} C_k \mathcal{O}_F \dots \mathcal{O}_F C_0 |\phi_0\rangle, C_T \mathcal{O}_{F'} \cdots \mathcal{O}_{F'} C_{k-1} \mathcal{O}_F \dots \mathcal{O}_F C_0 |\phi_0\rangle) \\
&= \delta(\mathcal{O}_F C_{k-1} \dots \mathcal{O}_F C_0 |\phi_0\rangle, \mathcal{O}_{F'} C_{k-1} \dots \mathcal{O}_F C_0 |\phi_0\rangle) \\
&= \delta(\mathcal{O}_F |\phi_{k-1}^F\rangle, \mathcal{O}_{F'} |\phi_{k-1}^F\rangle) = \delta(|\phi_{k-1}^F\rangle, \mathcal{O}_F \mathcal{O}_{F'} |\phi_{k-1}^F\rangle) = \delta(|\phi_{k-1}^F\rangle, \mathcal{O}_P |\phi_{k-1}^F\rangle) .
\end{aligned}$$

For pure states $|\psi\rangle$ and $|\psi'\rangle$, $\delta(|\psi\rangle, |\psi'\rangle) \leq \||\psi\rangle - |\psi'\rangle\|$. Note that $|\psi\rangle = \Pi_B|\psi\rangle + (I - \Pi_B)|\psi\rangle$, and O_P operates identically on $(I - \Pi_B)|\psi\rangle$. Therefore

$$
\begin{aligned}
\mathop{\mathbb{E}}_B[\delta(\phi^F, \phi^{P\oplus F})] &\leq T \max_{|\phi\rangle} \mathbb{E}\, \||\phi\rangle - \mathcal{O}_P|\phi\rangle\| \\
&= T \max_{|\phi\rangle} \mathop{\mathbb{E}}_B \|\Pi_B|\phi\rangle - \mathcal{O}_P\Pi_B|\phi\rangle + (1 - \mathcal{O}_P)(I - \Pi_B)|\phi\rangle\| \\
&\leq T \max_{|\phi\rangle} \mathop{\mathbb{E}}_B (\|\Pi_B|\phi\rangle\| + \|\mathcal{O}_P\Pi_B|\phi\rangle\|) \\
&= 2T \max_{|\phi\rangle} \mathop{\mathbb{E}}_B \|\Pi_B|\phi\rangle\| \\
&\leq 2T \max_{|\phi\rangle} \sqrt{\mathop{\mathbb{E}}_B |\langle\phi|\Pi_B|\phi\rangle|} \quad \text{(Jensen's inequality)}\,.
\end{aligned}
$$

Let π be a uniformly random element of the symmetric group on $\{0,1\}^n$ and U_π be the unitary operator associated with the permutation π. We have that

$$
\mathop{\mathbb{E}}_B\Big[|\langle\phi|\Pi_B|\phi\rangle|\Big] = \mathop{\mathbb{E}}_B \mathop{\mathbb{E}}_\pi \Big[|\langle\phi|U_\pi \Pi_B U_\pi^{-1}|\phi\rangle|\Big] = 2^{-n} \mathop{\mathbb{E}}_B[\mathrm{Tr}\,(\Pi_B)] = \epsilon\,.
$$

Thus we conclude that $\mathbb{E}_B[\delta(\phi^F, \phi^{P\oplus F})] \leq 2T\sqrt{\varepsilon}$. □

B SUPPLEMENTARY MATERIAL

B.1 Non-adaptive quantum queries and "double spending"

The following lemma shows that if there exists a non-adaptive quantum algorithm $\mathcal{A}$ making q queries to a function $f : \{0,1\}^n \to \{0,1\}^m$ that learns a certain property $p(f)$, then with inverse polynomial probability, there exists another non-adaptive q-query algorithm that learns $p(f)$ and q input-output-pairs with inverse polynomial probability. For this to hold, we need to assume that $\mathcal{A}$ makes its queries using a blank output register (i.e., initialized in the $|0\rangle$ state). This is the case, e.g., in period-finding and Simon's algorithm.

In the following, denote the set of n-bit-to-m-bit functions by $\mathcal{F}(n, m)$.

Lemma 8 (Double spending lemma)**.** *Let $F \subseteq \mathcal{F}(n, m)$ be a set of functions, P a set, $p : F \to P$ a function, and D a probability distribution on F. Suppose there exists a quantum query algorithm $\mathcal{A}$ which makes q non-adaptive quantum queries to $\mathcal{O}_f$ with blank output register for $f \leftarrow D$ and outputs $p(f)$ with $1/\mathsf{poly}(n)$ probability. Then there also exists an algorithm $\mathcal{A}'$ which makes q non-adaptive quantum queries to $\mathcal{O}_f$ for $f \leftarrow D$ and outputs both $p(f)$ and q input-output pairs of f with $1/\mathsf{poly}(n)$ probability.*

Proof. Let $\mathcal{X} = \{0,1\}^n$, $\mathcal{Y} = \{0,1\}^m$ and $\mathcal{H}_Z = \mathbb{C}\mathcal{Z}$ for $Z = X, Y$. Set $\mathcal{A}^{\mathcal{O}}(1^n) = \mathcal{E}\left(\mathcal{O}^{\otimes q}|\psi\rangle_{X^q} \otimes |0\rangle_{Y^q}\right)$ where $|\psi\rangle$ is some input state and $\mathcal{E} = \{E_p\}_{p\in P}$ is a POVM on $\mathcal{H}_X^{\otimes q} \otimes \mathcal{H}_Y^{\otimes q}$ with outcomes labelled by the possible properties of f. Let $|\psi_1\rangle = \mathcal{O}^{\otimes q}|\psi\rangle_{X^q} \otimes |0\rangle_{Y^q}$. $\mathcal{A}$ outputs $p(f)$ with inverse polynomial probability, say with probability $p_{\mathrm{succ}} = \langle\psi_1|E_{p(f)}|\psi_1\rangle$. It follows that the post-measurement state conditioned on the outcome $p(f)$,

$$
|\psi_2^{p(f)}\rangle = \frac{\sqrt{E_{p(f)}}|\psi_1\rangle}{\sqrt{\langle\psi_1|E_{p(f)}|\psi_1\rangle}},
$$

has inverse polynomial overlap with $|\psi_1\rangle$,

$$
\begin{aligned}
\left\langle \psi_1 \mid \psi_2^{p(f)} \right\rangle &= \frac{\langle\psi_1|\sqrt{E_{p(f)}}|\psi_1\rangle}{\sqrt{\langle\psi_1|E_{p(f)}|\psi_1\rangle}} \\
&\geq \sqrt{\langle\psi_1|E_{p(f)}|\psi_1\rangle}. \qquad (78)
\end{aligned}
$$

This implies immediately that measuring $|\psi_2^{p(f)}\rangle$ in the computational basis will yield q input output pairs of f with inverse polynomial success probability. □

We remark that the distribution of input-output pairs is at most $1 - 1/\mathsf{poly}(n)$ far from the distribution one would get by simply measuring immediately after the query of $\mathcal{A}$. This means that, in the case of period-finding and Simon's algorithm (where the queries are uniform), the input-output pairs will be distinct with non-negligible probability.

B.2 Alternative proof that random functions are PO-secure

Using Lemma 2, we can give a simple proof of the fact that a random function is PO-secure. Because of its simplicity, and because much of it can be reused to prove a separation between PO and BU, we provide this proof below.

Theorem 20 ([6]). *An algorithm making q quantum queries to a random oracle $f : \{0,1\}^n \to \{0,1\}^m$ produces $q+1$ input-output pairs of f with probability at most*

$$\frac{2^{\lceil \log(q+1) \rceil}}{2^m}. \tag{79}$$

Note that the probability bound is within a factor of 2 of the one obtained in [6], and matches it for $q + 1 = 2^k$, $k \in \mathbb{N}$.

Proof. Let $\mathcal{A}$ be an adversary that, when provided with the quantum random oracle f, outputs $q+1$ candidate input-output pairs. Formally, let $\rho_{(X,Y)^{q+1}F}$ be the joint cq-state of the adversary and the FO, where the classical registers $(X,Y)^{q+1}$ contain $\mathcal{A}$'s output and F is the FO's register. If we wanted to determine the success of $\mathcal{A}$ at this point, we would apply the Fourier transform to F, and then measure F and check if the outcome for F_{x_i} is y_i for each (x_i, y_i) output by $\mathcal{A}$.

Note that $P_l \rho = 0$ for all $l > q$ by Lemma 2, i.e., there are at most q entries of F that are nonzero. This implies that the entry corresponding to at least one of the inputs that $\mathcal{A}$ has output is, in fact, equal to 0^m. However, this is only true in superposition: different branches of the superposition may have different entries in the state $|0^m\rangle$. We will deal with this issue by thinking about a new algorithm $\mathcal{B}$, which will simulate the entire execution of $\mathcal{A}$ (including the oracle) and then perform a small number of additional measurements prior to the success check. The additional measurements will find a pair (x_{i_0}, y_{i_0}) in $(X,Y)^{q+1}$ such that $F_{x_{i_0}}$ is actually in the state $|0^m\rangle$ (in every branch of the superposition). The probability that $y_{i_0} = f(x_{i_0})$ (in the execution of $\mathcal{B}$) will then be 2^{-m}. We will then apply Lemma 1 to show that the success probability of $\mathcal{A}$ is not much better.

We now describe $\mathcal{B}$ in detail. Initially, $\mathcal{B}$ simulates both $\mathcal{A}$ and the oracle. After $\mathcal{A}$ has finished, but before the success check is performed, $\mathcal{B}$ (which is in the state ρ) applies binary search to the $q+1$ inputs that $\mathcal{A}$ has output. The goal is to find an input x_{i_0} such that $F_{x_{i_0}}$ is in state $|0^m\rangle$. We do this using binary measurements that ask "are any of the registers $F_{x_{i_1}}, ..., F_{x_{i_k}}$ in the state $|0^m\rangle$?" We split up the set $S_0 = \{x_1, ..., x_{q+1}\}$ into two subsets $S_0^{\mathsf{L}} = \{x_1, ..., x_{\lfloor (q+1)/2 \rfloor}\}$ and $S_0^{\mathsf{R}} = \{x_{\lfloor (q+1)/2 \rfloor + 1}, ..., x_{q+1}\}$, and measure whether F_x is in a state different from $|0^n\rangle$ for all $x \in S_0^{\mathsf{L}}$. This is done using the binary measurement given by

$$P_1 = (\mathbb{1} - |0\rangle\langle 0|)_{x_1} \otimes ... \otimes (\mathbb{1} - |0\rangle\langle 0|)_{x_{\lfloor (q+1)/2 \rfloor}} \otimes \mathbb{1}^{\otimes (2^n - \lfloor (q+1)/2 \rfloor)} \tag{80}$$

and its complementary projector $P_0 = \mathbb{1} - P_1$. If the outcome is no, we set $S_1 = S_0^{\mathsf{L}}$, if it is yes then we set $S_1 = S_0^{\mathsf{R}}$. This makes sure that we continue with a set that contains an input such that the corresponding FO register is in state $|0^n\rangle$. Now we repeat the described steps using S_1 in place of S_0 and continue recursively until we encounter a set S_l with only one element, say w. Continuing with the success check, we now know that F_w is in the state $|0^m\rangle$, which implies that $f(w)$ is uniformly random and independent of $\mathcal{A}$'s output. Indeed, a register that is in a pure state is automatically in product with the rest of the universe, and $f(w)$ is determined by applying $H^{\otimes m}$, which transforms $|0^m\rangle$ into $|\phi_0\rangle$, and measuring, which yields a uniformly random outcome. Therefore $\mathcal{A}$'s success probability is at most 2^{-m}. The total number of binary measurements for the binary search procedure is upper-bounded by $\lceil \log(q+1) \rceil$, so an application of Lemma 1 finishes the proof. $\square$

B.3 A useful lemma on Bernoulli-preserving hash

Recall that blinding a function $f : \{0,1\}^n \to \{0,1\}^t$ on a set $B \subseteq \{0,1\}^n$ results in the blinded function Bf defined by $Bf(x) = \perp = (0^t, 1)$ for $x \in B$ and $Bf(x) = (f(x), 0)$ for $x \notin B$. The following lemma, which is implicit in the proof of Hash-and-MAC construction, could be useful in security reductions involving Bernoulli-preserving hash.

Lemma 9. *Let $h : \{0,1\}^n \to \{0,1\}^m$ be a Bernoulli-preserving hash and $f : \{0,1\}^n \to \{0,1\}^t$ an efficiently computable function. Then for all oracle QPTs $(\mathcal{A}, \epsilon)$, we have*

$$\left| \Pr_{B \leftarrow \mathcal{O}_\epsilon} \left[\mathcal{A}^{Bf}(1^n) = 1 \right] - \Pr_{B \leftarrow \mathcal{O}_\epsilon^h} \left[\mathcal{A}^{Bf}(1^n) = 1 \right] \right| \leq \mathsf{negl}(n)\,.$$

Proof. It suffices to observe that one can simulate the oracle for Bf using two calls to an oracle for χ_B and two executions of f, as follows.

$$\begin{aligned}
|x\rangle|y\rangle|b\rangle &\mapsto |x\rangle|y\rangle|b\rangle|\chi_B(x)\rangle|f(x)\rangle \\
&\mapsto |x\rangle|y \oplus \chi_B(x) \cdot f(x)\rangle|b \oplus \chi_B(x)\rangle|\chi_B(x)\rangle|f(x)\rangle \\
&\mapsto |x\rangle|y \oplus \chi_B(x) \cdot f(x)\rangle|b \oplus \chi_B(x)\rangle \\
&= |x\rangle|y \oplus Bf(x)\rangle.
\end{aligned}$$

In the second step, we applied the CCNOT (Toffoli) gate to the second register, with the fourth and fifth register as the controls and a CNOT to the third register with the fourth register as a control. With this observation, it is straightforward to turn any distinguisher for $B_\epsilon f$ vs. $B_\epsilon^h f$ into one for χ_{B_ϵ} vs. $\chi_{B_\epsilon^h}$. $\square$