Quantum Information and Computation, Vol. 21, No. 3&4 (2021) 0181–0202
© Rinton Press

EVERLASTING SECURITY OF QUANTUM KEY DISTRIBUTION WITH 1K-DWCDM AND QUADRATIC HASH

KHODAKHAST BIBAK
Department of Computer Science and Software Engineering, Miami University
Oxford, Ohio, 45056, USA
Email: bibakk@miamioh.edu

ROBERT RITCHIE
Department of Computer Science and Software Engineering, Miami University
Oxford, Ohio, 45056, USA
Email: ritchirp@miamioh.edu

BEHROUZ ZOLFAGHARI
CSE Department, Indian Institute of Technology Guwahati
Guwahati, 781039, India
Cyber Science Lab, School of Computer Science, University of Guelph
Guelph, Ontario, N1G 2W1, Canada
Email: behrouz@cybersciencelab.org

Received November 27, 2020
Revised January 16, 2021

Quantum key distribution (QKD) offers a very strong property called everlasting security, which says if authentication is unbroken during the execution of QKD, the generated key remains information-theoretically secure indefinitely. For this purpose, we propose the use of certain universal hashing based MACs for use in QKD, which are fast, very efficient with key material, and are shown to be highly secure. Universal hash functions are ubiquitous in computer science with many applications ranging from quantum key distribution and information security to data structures and parallel computing. In QKD, they are used at least for authentication, error correction, and privacy amplification. Using results from Cohen [*Duke Math. J.*, 1954], we also construct some new families of ε-almost-Δ-universal hash function families which have much better collision bounds than the well-known Polynomial Hash. Then we propose a general method for converting any such family to an ε-almost-strongly universal hash function family, which makes them useful in a wide range of applications, including authentication in QKD.

Keywords: Quantum key distribution; Everlasting security; 1K-DWCDM; Universal hashing; Quadratic congruence

Communicated by: S Braunstein & H Zbinden

1 Introduction

Highly sensitive data, such as government, military, and medical data, may have to be kept secure for decades. But the attacker can store all these data (storage is cheap) and after having enough computational power break the systems. Quantum key distribution (QKD) provides a solution for this fundamental weakness of classical systems. Sharing some secret information by two or more parties for encrypting and authenticating messages is typically accomplished through the key agreement protocols in public-key cryptography, such as Diffie–Hellman key exchange (DH) or elliptic-curve Diffie–Hellman (ECDH). However, the security of public-key cryptography schemes rely on the computational difficulty of certain mathematical problems (namely, the discrete logarithm problem, the elliptic-curve discrete logarithm problem, and the integer factorization problem) which can be solved in polynomial time on a quantum computer running Shor's algorithm. So, if sufficiently powerful quantum computers are ever realized, then we can no longer rely on such key agreement protocols.

QKD, on the other hand, enjoys a higher level of security than key agreement protocols based on public-key cryptography, as it is not based on the difficulty assumptions of certain problems. In fact, QKD relies on the foundations of quantum mechanics for its security. Thus, it is provably secure even against an adversary with unbounded computational power. Most importantly, QKD enjoys a powerful property called *everlasting security* [1, 2, 3], which says that, if authentication remains secure during the execution of the QKD protocol, then the resulting key is information-theoretically secure; breaking authentication after the protocol has output the key will not change the security of the generated key. So even a computationally unbounded adversary cannot recover the key after the QKD execution. This is of particular importance for security of highly sensitive data, such as government, military, and medical applications, where data may have to be kept secure for decades.

QKD is also becoming increasingly feasible to implement. For example, in the U.S.A., the company Batelle, partnered with the company Quantique, installed a QKD network between Batelle's offices in Columbus, Ohio and Washington, DC [4]. The largest QKD network is in China connecting Bejing, Jifan, Shanghai, and Hefei [4].

The universal hash function families constructed in this paper are applicable in various steps of the QKD protocol, since universal hashing is used not only for authentication in QKD, but also in other steps of QKD like error correction and privacy amplification. In a QKD network, there are both quantum and classical channels of communication. First, the parties obtain some quantum states and measure them. Using classical channels, they determine which results of their measurements can produce secret bits, and discard the rest. Then they perform error correction and privacy amplification, both of which utilize universal hash functions. Error correction utilizes error correcting codes to fix any noise which may occur during communication. These codes have a direct correspondence to universal hash functions. Privacy amplification compresses the raw key material with a shared secret universal hash function, in order to compress the adversary's knowledge on the key to an arbitrarily small amount. It is crucial that these steps are authenticated with a pre-shared secret. Otherwise, an eavesdropper could perform a man-in-the-middle (MITM) attack, and completely recover the key material. Thus, the classical channel needs to be authenticated, usually using the original MACs proposed by Wegman and Carter [5] (where the message is first hashed with an ε-almost-strongly universal hash function and then encrypted with a one-time pad), since the Wegman–Carter construction is information-theoretically (unconditionally) secure, has low key usage, and also is very fast.

To authenticate the QKD protocols, Price *et al.* [6] suggest using the output of AES-256 evaluated at a unique nonce to encrypt the hash value (see the WCBK paradigm discussed in Section 3) instead of using a one-time pad. While this version of QKD, which they call BB84-

AES [6], is no longer information-theoretically secure, Price *et al.* [6] argue that, in practice, this is acceptable in terms of security. Because some QKD networks are already utilizing computationally secure encryption schemes (c.f. [7, 8]), they claim computationally secure authentication would not degrade the security of the system. Furthermore, Price *et al.* [6] claim that using BB84-AES would increase resilience against Denial-of-Service (DoS) attacks, which QKD systems are known to be vulnerable to. They also offer several suggestions for implementing BB84-AES, and discuss some security implications.

In this paper, we suggest applying some *computationally secure* universal hashing based MACs for authentication in QKD. Because of the everlasting security property of QKD [1, 2, 3], using a computationally secure MAC will still result in a secure key (which even a computationally unbounded adversary cannot recover after the QKD protocol execution) if the MAC is not broken during the QKD execution. A portion of the key generated by QKD can also be used as key material for authentication in future rounds of QKD. So the subsequent rounds of QKD can still achieve everlasting security, even if computationally secure authentication is employed in the prior rounds.

There are more appealing universal hashing based MACs which can be employed in QKD. For example, Decrypted–Wegman–Carter with Davies–Meyer (DWCDM), discussed in Section 3, offers increased security and low key usage for relatively little additional cost of computation. This would help impede an adversary from breaking authentication quickly enough to perform MITM. It has been shown [9] that ε-almost-strongly universal (ε-ASU) hash functions are universally composable (UC) [10], and therefore they are sufficient for authentication in QKD systems. Because ε-ASU hash function families are the main ingredient in the Wegman–Carter construction, DWCDM, and other universal hashing based MACs, we propose a method for constructing such families. Using some results of Cohen [11] on the number of solutions of certain quadratic congruences, we construct some new ε-almost-Δ-universal (ε-AΔU) hash function families which have much better collision bounds than the well-known Polynomial Hash. Then we propose a general method for converting any such families to ε-ASU hash function families. Because the latter are the strongest relaxation of strong universality, the ε-ASU families constructed in this paper can be used for authentication in QKD, and everywhere universal hashing is needed (see Section 2 for a wide range of applications).

The rest of this paper is organized as follows. In Section 2, we formally define universal hashing, its variants, and discuss some of their applications. We also prove a result which relaxes the preconditions of the MACs proposed by Datta *et al.* [12], and also helps us to simplify their security bounds. In Section 3, we describe some of the main methods for constructing MACs based on universal hashing, and propose the use of certain universal hashing based MACs, including the modified variants of some MACs proposed by Datta *et al.* [12] for authentication in QKD. These MACs are fast, very efficient with key material, and are shown to be highly secure. In Section 4, we define our families of universal hash functions that we call Quadratic Hash (QH) and Odd Quadratic Hash (OQH), and investigate their universality using some deep results of Cohen [11]. Finally, in Section 5, we propose a general method for converting any ε-AΔU hash function family to an ε-ASU hash function family, which makes them useful for many applications including authentication in QKD.

2 Universal hashing, its variants, and applications

Universal hash functions satisfy some special collision resistance properties. These hash functions, introduced by Carter and Wegman [13], have many applications in computer science, including quantum key distribution [14, 9, 6, 4], cryptography and information security [15, 16, 17, 18, 19, 20, 21, 22, 23, 24], error-correcting codes [25, 26], pseudorandomness [27, 28], complexity theory [29, 30], randomized algorithms [31, 32], data structures [33, 34], and parallel computing [35, 36, 37].

We now provide a formal definition of universal hashing and its variants [13, 17, 38, 23, 39, 26]. For a set $\mathcal{X}$, we write $x \leftarrow \mathcal{X}$ to denote that x is chosen uniformly at random from $\mathcal{X}$.

Definition 2.1. Let H be a family of functions from a finite domain D to a finite range R, and let ε be a constant such that $\frac{1}{|R|} \leq \varepsilon < 1$.

- The family H is a *universal* family of hash function if the probability, over a random choice of a hash function from H, that two distinct elements of D *collide* (i.e., have the same hash value) is at most $1/|R|$ (that is, distinct elements of D do not collide too often). Formally, H is universal if for any two distinct $x, y \in D$, we have $\Pr_{h \leftarrow H}[h(x) = h(y)] \leq \frac{1}{|R|}$. Also, H is an ε-*almost universal* (ε-AU) family of hash functions if for any two distinct $x, y \in D$, we have $\Pr_{h \leftarrow H}[h(x) = h(y)] \leq \varepsilon$. Note that an ε-AU family, for a sufficiently small ε, is *close* to being universal.

- Suppose R is a finite additive Abelian group. The family H is a Δ-*universal* family of hash functions if, given a randomly chosen hash function from H, the difference of the hash values of any two distinct elements of D is uniformly distributed in R. Formally, H is Δ-universal if for any two distinct $x, y \in D$, and all $b \in R$, we have $\Pr_{h \leftarrow H}[h(x) - h(y) = b] = \frac{1}{|R|}$, where '$-$' denotes the group subtraction operation. Also, H is an ε-*almost*-Δ-*universal* (ε-AΔU) family of hash functions if for any two distinct $x, y \in D$, and all $b \in R$, we have $\Pr_{h \leftarrow H}[h(x) - h(y) = b] \leq \varepsilon$. When $R = \mathbb{Z}_2^k = \{0, 1\}^k$ for some k, the operation '$-$' can be replaced by '$\oplus$' (XOR), and H is also called ε-*almost XOR universal* (ε-AXU) or ε-*otp-secure*.

- The family H is a *strongly universal* (or 2-*independent*) family of hash functions if, given a randomly chosen hash function from H, the hash values of any two distinct elements of D are independent and uniformly distributed in R. Formally, H is strongly universal if for any two distinct $x, y \in D$, and all $a, b \in R$, we have $\Pr_{h \leftarrow H}[h(x) = a,\ h(y) = b] = \frac{1}{|R|^2}$. Also, H is an ε-*almost-strongly universal* (ε-ASU) family of hash functions if for any two distinct $x, y \in D$, and all $a, b \in R$, we have

  - $\Pr_{h \leftarrow H}[h(x) = a] = \frac{1}{|R|}$ (that is, given a randomly chosen h from H, $h(x)$ is uniformly distributed in R), and
  - $\Pr_{h \leftarrow H}[h(x) = a \,|\, h(y) = b] \leq \varepsilon$ (that is, given a randomly chosen h from H, $h(x)$ is *hard to guess* even if $h(y)$ is known).

  Equivalently, H is ε-ASU if for any two distinct $x, y \in D$, and all $a, b \in R$, we have

  - $\Pr_{h \leftarrow H}[h(x) = a] = \frac{1}{|R|}$, and

– $\Pr_{h \leftarrow H}[h(x) = a,\ h(y) = b] \leq \frac{\varepsilon}{|R|}$.

We will also use the following definitions from Datta *et al.* [12].

Definition 2.2. Let H be a family of functions from a finite domain D to a finite range R, and let ε be a constant such that $\frac{1}{|R|} \leq \varepsilon < 1$.

- The family H is ε regular if for any $x \in D$, and $r \in R$, we have $\Pr_{h \leftarrow H}[h(x) = r] \leq \varepsilon$,
- The family H is ε 3-way regular if for any $x, y, z \in D$, and $r \in R$, we have $\Pr_{h \leftarrow H}[h(x) \oplus h(y) \oplus h(z) = r] \leq \varepsilon$.

However, we show that these properties are directly implied by ε-almost-Δ-universality. In Section 3.3, we use this to relax the preconditions of the MACs proposed by Datta *et al.* [12], and simplify their security bounds.

Theorem 2.3. *If H is an ε-AΔU family of hash functions from a domain D to a range R, where R is an Abelian group with subtraction operation $\oplus$, then H is also ε regular, and ε 3-way regular.*

Proof. Let x, y, z be distinct arbitrary elements of D, r be an arbitrary element of R, and $h \in H$. Define $r_1 = r \oplus h(z)$. Since $r, h(z) \in R$, we also have $r_1 \in R$, because R is closed under $\oplus$. Note that

$$h(x) \oplus h(y) \oplus h(z) = r$$

if and only if

$$h(x) \oplus h(y) = r_1.$$

Therefore,

$$\Pr_{h \leftarrow H}[h(x) \oplus h(y) \oplus h(z) = r] = \Pr_{h \leftarrow H}[h(x) \oplus h(y) = r_1] \leq \varepsilon,$$

where the inequality is implied by the assumption that H is ε-AΔU. So, H is ε 3-way regular.

Similarly,

$$\Pr_{h \leftarrow H}[h(x) = r] \leq \varepsilon$$

is equivalent to

$$\Pr_{h \leftarrow H}[h(x) \oplus h(y) = r_2] \leq \varepsilon,$$

where $r_2 = r \oplus h(y)$. So, H is ε regular. □

As mentioned above, applications of universal hashing have been found in many fields, but they have received the most attention in the construction of message authentication codes (MACs). A MAC algorithm outputs an *authentication tag* computed by the sender using a message and the secret key. The receiver verifies the integrity of the message by recomputing the tag using the secret key and the received message, and comparing it to the tag received. There are three main approaches for constructing MACs, namely, constructions based on block ciphers (like CBC-MAC [40], CMAC [41, 42, 43], and PMAC [44]), collision resistant hash functions (like HMAC [45]), and universal hash functions (like UMAC [15], GMAC [46], and Poly1305 [47]).

3 MAC constructions based on universal hashing

MACs based on universal hashing are among the highly secure and fastest MAC algorithms. Some of the main methods for constructing such MACs are the Wegman–Carter–Brassard–Krawczyk paradigm, the Decrypted–Wegman–Carter with Davies–Meyer paradigm and their variants. These constructions are described in the following subsections, but first we need to describe pseudorandom functions and their security.

3.1 Pseudorandom functions

Pseudorandom functions (PRFs) are vital tools in cryptography. A PRF $F : \mathcal{X} \times \mathcal{K} \to \mathcal{Y}$ maps an input block $x \in \mathcal{X}$ using a key $k \in \mathcal{K}$, to an output block $y \in \mathcal{Y}$. The idea is that if k is chosen uniformly at random, then the output of the function should also appear random. Here, F is a family of functions, where choosing a key k defines a specific function $f_k : \mathcal{X} \to \mathcal{Y}$.

Let $\mathrm{Funs}[\mathcal{X}, \mathcal{Y}]$ be the set of all functions from $\mathcal{X}$ to $\mathcal{Y}$. Here, and in the rest of the paper, what is meant by an *efficient adversary*, is a probabilistic polynomial time (PPT) adversary. We measure the security of the PRF family F against an efficient adversary $\mathcal{A}$ by $\mathcal{A}$'s advantage in distinguishing F from $\mathrm{Funs}[\mathcal{X}, \mathcal{Y}]$ as described in the following experiment (we follow closely the description of [48]).

For $b = 0, 1$, Experiment b includes the following steps:

- The challenger first picks a function f as follows:

  if $b = 0$, the challenger randomly selects a key $k \leftarrow \mathcal{K}$ which defines some $f = f_k \in F$;

  if $b = 1$, the challenger randomly selects $f \leftarrow \mathrm{Funs}[\mathcal{X}, \mathcal{Y}]$.

- The adversary queries the challenger with input blocks $x_1, x_2, \ldots \in \mathcal{X}$, and the challenger responds with the output of f on these inputs, $f(x_1), f(x_2), \ldots \in \mathcal{Y}$.

- The adversary guesses whether f is a PRF or a truly random function by outputting a bit $b' = 0$ or $b' = 1$.

The *advantage* of $\mathcal{A}$ with respect to F is defined as,

$$\mathrm{PRFadv}[\mathcal{A}, F] := |\Pr[E_0] - \Pr[E_1]|,$$

where E_b is the event that the adversary outputs 1 in Experiment b. In other words, adversary's advantage measures its ability in distinguishing a random function of the PRF family from a truly random function.

Theorem 3.1. *The PRF family F is secure if for any efficient adversary $\mathcal{A}$,* $\mathrm{PRFadv}[\mathcal{A}, F]$ *is negligible.*

It is important to note that the adversary's queries can be *adaptive* in this experiment. That is, it does not have to send all queries at once, it may wait on the response of one query to decide the next. If $\mathcal{X} = \mathcal{Y}$, and every $f_k : \mathcal{X} \to \mathcal{X}$ is a bijection, then F is instead called a family of pseudorandom permutations (PRP) or a block cipher. The security of PRPs is analogous to the security of PRFs, except an adversary attempts to distinguish F from the set of random permutations over $\mathcal{X}$, $\mathrm{Perms}[\mathcal{X}]$.

3.2 Wegman–Carter–Brassard–Krawczyk paradigm

Wegman and Carter [5] proposed the following method for MAC construction. In this scheme, the legitimate parties share a secret hash function chosen uniformly at random from a strongly universal family of hash functions, and a secret encryption key (a sequence of random one-time pads). A message is authenticated by first hashing it with the shared hash function and then encrypting the resulting hash value with the shared encryption key (shared one-time pad). Note that one-time pads are of the length of the hash value rather than of the length of the message. The resulting encrypted hash value, called an *authentication tag*, is transmitted together with the message (as a pair). Upon receiving this pair, the legitimate party recomputes and validates it. This scheme is provably secure in the information-theoretic setting. In fact, such a MAC algorithm is *information-theoretically secure*, that is, even an adversary who has unbounded computational power cannot forge the MAC with probability greater than the collision probability of the hash family [5].

Brassard [49] constructed a computationally secure MAC by replacing the one-time pad by the output of a pseudorandom function (PRF) applied to a nonce. Also, Krawczyk [38] showed that one can use an ε-AΔU family in this construction; in this case the adversary cannot forge the MAC with a probability greater than ε. The Wegman–Carter–Brassard–Krawczyk (WCBK) paradigm is described formally as follows:

Let H be an ε-AΔU family of hash functions and F be a PRF family. The secret key for this construction is a pair $\langle h, f\rangle$, with $h \leftarrow H$ and $f \leftarrow F$. The authentication tag $\mathbf{t}$ for a message m is computed as a pair

$$\mathbf{t} = \langle r, h(m) \oplus f(r)\rangle,$$

where r is a nonce (typically, a counter which is incremented by one following the generation of each authentication tag).

Krawczyk [38] proved the following result (in a slightly different form) about the security of this MAC algorithm (see also [50], [5]).

Theorem 3.2. *In the WCBK paradigm, the probability that an efficient adversary $\mathcal{A}$ who performs q_m black-box MAC queries and q_v black-box verification queries to successfully forge the MAC is independent of q_m and is at most*

$$\mathrm{PRFadv}[\mathcal{A}, F] + q_v\varepsilon.$$

Thus, if the PRF is secure, WCBK enjoys exceptional security, however, the security of F is a limitation in concrete implementations. Often, in practice, a PRP is used instead of a PRF, and by the well known PRP-PRF Switching Lemma, if F consists of n-bit PRPs, then

$$\mathrm{PRFadv}[\mathcal{A}, F] \leq \mathrm{PRPadv}[\mathcal{A}, F] + \frac{(q_m + q_v)^2}{2^{n+1}},$$

which introduces a birthday-type term. This poses a problem in lightweight environments, since a MAC which uses a 64-bit block cipher would only be secure up to 2^{32} MAC queries, which is usually insufficient. Furthermore, it is critically important that nonces are not reused in this scheme. If an adversary sees tags $\mathbf{t_1} = \langle r, h(m_1) \oplus f(r)\rangle$ and $\mathbf{t_2} = \langle r, h(m_2) \oplus f(r)\rangle$, then they can calculate

$$\mathbf{t_1} \oplus \mathbf{t_2} = \langle 0, h(m_1) \oplus h(m_2)\rangle,$$

which reveals information about h (two-time pad attack). For example, if H is polynomial-based, then

$$\mathbf{t_1} \oplus \mathbf{t_2} \oplus \langle 0, h(m_1)\rangle \oplus \langle 0, h(m_2)\rangle = \langle 0, 0\rangle,$$

which gives a polynomial equation with a root at the index of the hash function and with coefficients which depend on $\mathbf{t_1}$, $\mathbf{t_2}$, m_1, and m_2 [51].

This type of MAC construction is used in various applications and standards. For example, Galois/Counter Mode (GCM) [46] (which is used in IPsec, SSH, and TLS) and Poly1305 [47] (which is used in Google Chrome's TLS, and later was added to OpenSSH) use this scheme.

3.3 Decrypted–Wegman–Carter with Davies–Meyer

In order to overcome the WCBK paradigm's weakness to nonce misuse, the tag can be encrypted with a PRF as a final layer of security. That is,

$$\mathbf{t} = \langle r, f_2(h(m) \oplus f_1(r))\rangle,$$

where f_1, f_2 are PRFs and h comes from an ε-AΔU family of hash functions. This construction has the same security bound as the WCBK paradigm when nonces are never reused, and is secure up to the birthday bound when nonces are reused [52].

Cogliati and Seurin [52] slightly modify this construction by replacing f_2 with an n-bit block cipher E, and f_1 with the Encrypted–Davies–Meyer construction based on E, which then gives the tag:

$$\mathbf{t} = \langle r, E_{k_1}(h(m) \oplus E_{k_2}(r) \oplus r)\rangle.$$

They call this scheme Encrypted–Wegman–Carter with Davies–Meyer (EWCDM), and show that it enjoys beyond birthday bound security when nonces are not reused, and birthday bound security when they are reused.

Theorem 3.3. *In the* EWCDM *paradigm, when nonces are not reused, the probability that an efficient adversary $\mathcal{A}$ who performs q_m black-box MAC queries and q_v verification queries to successfully forge the MAC is at most*

$$2\mathrm{PRPadv}[\mathcal{A}, E] + \frac{5q_m^{3/2}}{2^n} + \frac{\varepsilon q_m}{2} + \frac{6q_v}{2^n} + \varepsilon q_v.$$

Furthermore, when nonces are reused, then the probability of a successful forgery is at most

$$2\mathrm{PRPadv}[\mathcal{A}, E] + \frac{2(q_m + q_v)^2}{2^n} + \frac{(q_m + q_v)^2\varepsilon}{2}.$$

Later, Mennink and Neves [53] proved an improved bound on EWCDM using Patarin's Mirror Theory [54, 55] and H-coefficients technique [56], showing that, in the nonce-respecting setting, the MAC is secure up to approximately 2^n MAC and verification queries. In the nonce-misuse setting, the MAC remains secure up to roughly $2^{2n/3}$ MAC queries, and 2^n verification queries.

Datta *et al.* [12] introduced a variant of EWCDM called Decrypted–Wegman–Carter with Davies–Meyer (DWCDM). The goal of this variant is to use less key material than EWCDM, which uses three keys, two for the PRPs, and one for the hash function. In DWCDM, the outer encryption is replaced with decryption, so we only need one PRP key. It is important to note that DWCDM cannot use a full n-bit nonce, instead it uses a $2n/3$-bit nonce, padded with 0s. This is to avoid a birthday bound forging attack (see [12] Sec. 4.1). The tag for DWCDM is then calculated as

$$\mathbf{t} = \langle r, D_{k_1}(h(m) \oplus E_{k_1}(r) \oplus r)\rangle.$$

DWCDM is secure up to roughly $2^{2n/3}$ MAC queries, and 2^n verification queries in the nonce-respecting setting, and it is secure roughly up to $2^{n/2}$ MAC queries, and 2^n verification queries in the nonce-misuse setting.

Theorem 3.4. *Suppose that H is a ε_1 regular, ε_2-AΔU, and ε_3 3-way regular hash function family. Then in the* DWCDM *paradigm, when nonces are not reused, the probability that an efficient adversary $\mathcal{A}$ who performs q_m black-box MAC queries and q_v verification queries to successfully forge the MAC is at most*

$$\begin{aligned} &\mathrm{PRPadv}[\mathcal{A}, E] + \frac{2q_m}{2^{2n/3}} + q_m\varepsilon_1 + \frac{2q_m\varepsilon_2}{2^{n/3}} \\ &+ \max\{q_v\varepsilon_1, 2q_v\varepsilon_2, 2q_v\varepsilon_3, \frac{q_m}{2^{2n/3}}\} + \frac{q_m+q_v}{2^n} + \frac{5q_m^3}{2^{2n}}. \end{aligned} \tag{3.1}$$

Furthermore, when nonces are reused, then the probability of a successful forgery is at most

$$\mathrm{PRPadv}[\mathcal{A}, E] + q_m^2\varepsilon_2 + \frac{4q_m^2}{2^n} + q_m\varepsilon_1 + \frac{q_m+q_v}{2^n}.$$

1K-DWCDM, also introduced by Datta *et al.* [12], further reduces key usage. It uses the same structure, but instead of picking the hash function key k_h randomly, it is derived from the PRP as $k_h = E_{k_1}(1)$. 1K-DWCDM is secure up to roughly $2^{2n/3}$ MAC queries, and 2^n verification queries in the nonce-respecting setting.

Theorem 3.5. *Suppose that H is a ε_1 regular, ε_2-AΔU, and ε_3 3-way regular hash function family. Then in the* 1K-DWCDM *paradigm, when nonces are not reused, the probability that an efficient adversary $\mathcal{A}$ who performs q_m black-box MAC queries and q_v verification queries to successfully forge the MAC is at most*

$$\begin{aligned} &\mathrm{PRPadv}[\mathcal{A}, E] + \frac{3q_m}{2^{n/3}} + \frac{\varepsilon_2 q_m^2}{2^n} + \frac{q_v}{2^{n-1}} \\ &+ \max\{q_v\varepsilon_1, 2q_v\varepsilon_2, 2q_v\varepsilon_3, \frac{q_m}{2^{2n/3}}\} + q_v\varepsilon_1 + \frac{q_m}{2^n} + \frac{5q_m^3}{2^{2n}}. \end{aligned} \tag{3.2}$$

The security of 1K-DWCDM *in the nonce-misuse setting is similar to that of* DWCDM.

The regularity and 3-way regularity requirements for DWCDM and 1K-DWCDM seem to be restrictive, as not many universal hash functions have been analysed with these properties in mind, but we have shown, in Theorem 2.3, that these properties are directly implied by ε-almost-Δ-universality. Using Theorem 2.3, we can simplify the bounds given above, with relaxed requirements.

Theorem 3.6. *Suppose that H is an ε-AΔU hash function family, and E is a block cipher. Then in the nonce-respecting setting, the probability of forging* DWCDM *is at most*

$$\begin{aligned} &\mathrm{PRPadv}[\mathcal{A}, E] + \frac{2q_m}{2^{2n/3}} + \left(q_m + \frac{2q_m}{2^{n/3}}\right)\varepsilon + \\ &\max\{2q_v\varepsilon, \frac{q_m}{2^{2n/3}}\} + \frac{q_m+q_v}{2^n} + \frac{5q_m^3}{2^{2n}}, \end{aligned} \tag{3.3}$$

and the probability of forging 1K-DWCDM *is at most*

$$\begin{aligned}&\mathrm{PRPadv}[\mathcal{A}, E] + \frac{3q_m}{2^{n/3}} + \left(q_v + \frac{q_m^2}{2^n}\right)\varepsilon + \frac{q_v}{2^{n-1}} + \\ &\max\{2q_v\varepsilon, \frac{q_m}{2^{2n/3}}\} + \frac{q_m}{2^n} + \frac{5q_m^3}{2^{2n}}.\end{aligned} \tag{3.4}$$

Furthermore, in the nonce-misuse setting, the probability of forging DWCDM *is at most*

$$\mathrm{PRPadv}[\mathcal{A}, E] + (q_m + q_m^2)\varepsilon + \frac{4q_m^2}{2^n} + \frac{q_m + q_v}{2^n}.$$

It is important to note that 1K-DWCDM uses significantly less key material than the original EWCDM, which is of particular importance in QKD, because the less key material is used, the more material is available for further communication. One reason Wegman–Carter MAC is often used in QKD is because of its low key usage, but the 1K-DWCDM paradigm recycles one key to authenticate multiple messages, making it even more appealing in this regard. Furthermore, as mentioned earlier, so long as authentication is not broken during the QKD protocol execution, the generated key remains information-theoretically secure, so the beyond birthday bound security makes it a good candidate for use in QKD. For example, AES-256 is believed to be secure in the quantum setting, and even a quantum adversary utilizing Grover's algorithm could defeat AES-256 only with probability roughly 2^{-128}. So choosing AES-256 would make this MAC secure up to roughly $2^{512/3}$ MAC queries and 2^{256} verification queries, which is much better than the WCBK paradigm previously suggested for use in QKD.

3.4 Benefits of using universal hashing

Constructing MACs based on universal hash functions is stunning from several points of view. Such MACs have desirable security properties, because the properties of universal hashing and the encryption step complement one another. Universal hashing compresses the message to a short string with mathematically proven collision bounds, unlike cryptographic primitives which rely on hardness assumptions. Furthermore, compressing the message to a short string means that the encryption step is fast, because it only needs to be performed on a short input. For the same reason, a strong encryption method can be chosen without much of a cost in performance. Finally, as Black *et al.* [15] put it, "the underlying cryptographic primitive is used only on short and secret messages, eliminating the typical avenues of attack. Under this approach security and efficiency are not conflicting requirements—quite the contrary, they go hand in hand." Because universal hashing is usually cheaper to implement, it is appealing for devices with limited power. For these reasons, universal hashing based MACs have been utilized extensively in the rising field of lightweight cryptography [57], for which NIST has published a call for algorithms, since many existing cryptography schemes do not perform well on constrained devices. In conclusion, universal hashing based MACs are among the most highly secure and fastest MAC algorithms.

4 (Odd) Quadratic Hash

Let $\mathbb{Z}_n$ be the ring of integers modulo n defined as $\mathbb{Z}_n = \{0, \ldots, n-1\}$. An almost universal hash functions family which has received much attention is Polynomial Hash (PH). In this family, each message block m_i and the key x are in $\mathbb{Z}_p$ (p is prime), and all operations are performed in $\mathbb{Z}_p$. Formally,

Definition 4.1. (PH) *Given a prime p,*

$$\mathrm{PH} := \{h_x : \mathbb{Z}_p^{d+1} \to \mathbb{Z}_p \,|\, x \in \mathbb{Z}_p\},$$

where

$$h_x(\boldsymbol{m}) := \sum_{i=0}^{d} m_i x^i \pmod{p},$$

for every message $\mathbf{m} = \langle m_0, m_1, \ldots, m_d\rangle \in \mathbb{Z}_p^{d+1}$ and every key $x \in \mathbb{Z}_p$.

It is well-known that the family PH is $\frac{d}{p}$-almost-Δ-universal. Polynomial Hash is widely attributed to Wegman and Carter [5], Dietzfelbinger et. al. [58], den Boer [59], Bierbrauer et. al. [60], and Taylor [61]. But we have discovered that it has been already introduced by Mehlhorn and Vishkin [62] back in 1984 (of course, Wegman and Carter [5] already studied the degree one case). Polynomial Hash has been used in GCM [46] and Poly1305 [47].

In this section, we define two other families of universal hash functions that we call **Quadratic Hash** (QH) and **Odd Quadratic Hash** (OQH), and investigate their universality. It turns out that these new families give much better collision bounds than the Polynomial Hash.

Definition 4.2. Let p be an odd prime and k be a positive integer. We define the family **Quadratic Hash** (QH) as follows:

$$\mathrm{QH} := \{h_{\mathbf{x}} \,:\, \mathbb{Z}_p^k \to \mathbb{Z}_p \,\ |\mathbf{x} \in \mathbb{Z}_p^k\},$$

where

$$h_{\mathbf{x}}(\mathbf{m}) := \sum_{i=1}^{k} m_i x_i^2 \pmod{p},$$

for any $\mathbf{x} = \langle x_1, x_2, \ldots, x_k\rangle \in \mathbb{Z}_p^k$ and any $\mathbf{m} = \langle m_1, m_2, \ldots, m_k\rangle \in \mathbb{Z}_p^k$.

The *Hamming distance* between two strings (vectors) of equal length is the number of positions at which the corresponding symbols (coordinates) are different. In our case, our strings are vectors in $\mathbb{Z}_p^k$, and the Hamming distance between vectors $\mathbf{x}, \mathbf{y} \in \mathbb{Z}_p^k$ is the number of coordinates where $x_i \neq y_i$. Now we define a variant of Quadratic Hash (QH) where the input is from a subset O of $\mathbb{Z}_p^k$ with the property that the Hamming distance between any two distinct vectors in O is an odd integer.

Definition 4.3. Let p be an odd prime and k be a positive integer. Also, let O be a subset of $\mathbb{Z}_p^k$ where the Hamming distance between any two distinct vectors in O is an odd integer. We define the family **Odd Quadratic Hash** (OQH) as follows:

$$\mathrm{OQH} := \{h_{\mathbf{x}} \,:\, O \to \mathbb{Z}_p \,\ |\mathbf{x} \in \mathbb{Z}_p^k\},$$

where

$$h_{\mathbf{x}}(\mathbf{m}) := \sum_{i=1}^{k} m_i x_i^2 \pmod{p},$$

for any $\mathbf{x} = \langle x_1, x_2, \ldots, x_k\rangle \in \mathbb{Z}_p^k$ and any $\mathbf{m} = \langle m_1, m_2, \ldots, m_k\rangle \in O$.

In order to study the universality of QH and OQH we need some results on the number of solutions of quadratic congruences which are discussed in the next subsection.

4.1 Quadratic congruences

Let us first review some definitions that are needed in the rest of the paper.

Definition 4.4. Let $a, n \in \mathbb{Z}$. We say that a is a *quadratic residue* modulo n, if a is congruent to a perfect square mod n. That is, there exists $x \in \mathbb{Z}_n$ such that

$$a \equiv x^2 \pmod{n}.$$

If a is not a quadratic residue, it is called a *quadratic nonresidue.*

Definition 4.5. Let p be an odd prime and a an integer. The *Legendre Symbol* $\left(\frac{a}{p}\right)$ is a quadratic character defined as follows.

$$\left(\frac{a}{p}\right) = \begin{cases} 0, & \text{if } a \equiv 0 \pmod{p}; \\ 1, & \text{if } a \text{ is a non-zero quadratic residue modulo } n; \\ -1, & \text{if } a \text{ is a quadratic nonresidue modulo } n. \end{cases}$$

Definition 4.6. Let a be an integer, and n a positive integer with prime factorization $n = p_1^{\alpha_1} p_2^{\alpha_2} \cdots p_r^{\alpha_r}$, where p_i is prime and $\alpha_i \geq 1$ for all $1 \leq i \leq r$. The *Jacobi Symbol* $\left(\frac{a}{n}\right)$ is a generalization of the Legendre symbol to composite moduli:

$$\left(\frac{a}{n}\right) = \left(\frac{a}{p_1}\right)^{\alpha_1} \left(\frac{a}{p_2}\right)^{\alpha_2} \cdots \left(\frac{a}{p_r}\right)^{\alpha_r}.$$

Note that if $\left(\frac{a}{n}\right) = -1$, then a is a quadratic nonresidue modulo n, but if $\left(\frac{a}{n}\right) = 1$, a is not necessarily a quadratic residue modulo n.

Two integers are said to be *coprime* (*relatively prime*) if their greatest common divisor (gcd) is 1. We use $\mathbf{0}$ to denote the vector of all zeroes. We say that an integer n is *square-free*, if it is divisible by no perfect squares other than 1. That is, if n has prime factorization $p_1^{\alpha_1} \cdots p_r^{\alpha_r}$, then $\alpha_i = 1$ for all $1 \leq i \leq r$.

The *Euler's totient function* $\varphi(n)$ is defined as the number of positive integers up to n that are coprime to n. The *Möbius function* $\mu : \mathbb{N} \setminus \{\mathbf{0}\} \to \{-1, 0, 1\}$ is defined as follows:

$$\mu(n) = \begin{cases} -1, & \text{if } n \text{ is square-free with an odd number of prime factors;} \\ 1, & \text{if } n \text{ is square-free with an even number of prime factors;} \\ 0, & \text{if } n \text{ is not square-free.} \end{cases}$$

Denote $N_k(b, n)$ as the number of solutions to the quadratic congruence

$$a_1 x_1^2 + a_2 x_2^2 + \cdots + a_k x_k^2 \equiv b \pmod{n},$$

where $a_1, a_2, \ldots, a_k, b$ are integers and $n \geq 1$ is an odd integer.

Cohen [11] proved the following interesting and deep results for the number of solutions of the above congruence (see also [63]).

Theorem 4.7. ([11]) *If* $b \equiv 0 \pmod{n}$ *and all of the coefficients* a_i *are coprime to the modulus* n*, then*

$$N_k(0, n) = \begin{cases} n^{2m-1} \sum_{d \mid n} \left(\frac{(-1)^m a_1 \cdots a_k}{d}\right) \frac{\varphi(d)}{d^m}, & k = 2m; \\ n^{2m} \sum_{q^2 \mid n} \frac{\varphi(q^2)}{q^{2m+1}}, & k = 2m+1. \end{cases}$$

Theorem 4.8. ([11]) *If* $\gcd(b,n)=1$ *and all of the coefficients* a_i *are coprime to the modulus* n*, then*

$$N_k(b,n)=\begin{cases} n^{2m-1}\sum_{d\ \mid\ n}\left(\frac{(-1)^m a_1\cdots a_k}{d}\right)\frac{\mu(d)}{d^m}, & k=2m; \\ n^{2m}\sum_{q\ \mid\ n}\left(\frac{(-1)^{m+1}a_1\cdots a_k\cdot b}{q}\right)\frac{\mu^2(q)}{q^m}, & k=2m+1. \end{cases}$$

What happens when some of the coefficients a_i are zero mod n (and so not coprime to n)? This simple but important case can be addressed as follows.

Theorem 4.9. *Suppose* j *of the coefficients* a_i *are non-zero and coprime to* n *for some* $1\leq j\leq k$*. Then*

$$N_k(b,n)=n^{k-j}N_j(b,n).$$

Proof. Clearly the values of the $k-j$ variables x_i with zero-coefficients do not impact the solutions, since they add nothing to the sum, hence there are $|\mathbb{Z}_n|^{k-j}=n^{k-j}$ choices for the values of these variables. Because the non-zero coefficients are coprime to n, there are $N_j(b,n)$ solutions for these remaining variables. □

4.2 Universality of (Odd) Quadratic Hash

In this subsection, using the above results, we investigate the universality of OQH and QH.

Theorem 4.10. OQH *is a universal family of hash functions.*

Proof. Let $\mathbf{m}=\langle m_1,m_2,\ldots,m_k\rangle\in O$ and $\mathbf{m}'=\langle m_1',m_2',\ldots,m_k'\rangle\in O$ with $\mathbf{m}\neq\mathbf{m}'$. Define $\mathbf{a}=\langle a_1,a_2,\ldots,a_k\rangle=\mathbf{m}-\mathbf{m}'$. We have

$$\begin{aligned} & h_{\mathbf{x}}(\mathbf{m})-h_{\mathbf{x}}(\mathbf{m}')=0 \\ \Leftrightarrow & \sum_{i=1}^{k}m_ix_i^2-\sum_{i=1}^{k}m_i'x_i^2\equiv 0 \pmod{p} \\ \Leftrightarrow & \sum_{i=1}^{k}a_ix_i^2\equiv 0 \pmod{p}. \end{aligned}$$

Since $\mathbf{m},\mathbf{m}'\in O$, there are an odd number non-zero coordinates in $\mathbf{a}$. Let $\mathbf{a}$ have j non-zero coordinates, where $1\leq j\leq k$ and j is odd. Denote these non-zero coordinates as $a_{i_1},\ldots,a_{i_j}$. Since j is odd and $1\leq j\leq k$, we have $j=2m+1$ for some integer $m\geq 0$.

To bound the collision probability, we must find the maximum number of solutions $\mathbf{x}\in\mathbb{Z}_p^k$ to the congruence above over all choices of $\mathbf{a}\in\mathbb{Z}_p^k\setminus\{\mathbf{0}\}$, where $\mathbf{a}$ has an odd number of non-zero coordinates. All non-zero coefficients are coprime to p, so by Theorem 4.9, the number of solutions to the above congruence is exactly

$$N_k(0,p)=p^{k-j}N_j(0,p).$$

Applying Theorem 4.7,

$$\begin{aligned} N_j(0,p) &= p^{2m} \sum_{q^2 \mid p} \frac{\varphi(q^2)}{q^{2m+1}} \\ &= p^{2m} \left(\frac{\varphi(1^2)}{1^{2m+1}} \right) \\ &= p^{2m}. \end{aligned}$$

Since $|\mathbb{Z}_p^k| = p^k$, the probability of two distinct messages colliding over a randomly chosen $\mathbf{x}$ is

$$\frac{p^{k-j}p^{2m}}{p^k} = \frac{p^{2m}}{p^j} = \frac{p^{2m}}{p^{2m+1}} = \frac{1}{p}.$$

□

Theorem 4.11. QH *is* $\left(\frac{2}{p} - \frac{1}{p^2}\right)$*-almost-universal.*

Proof. Let $\mathbf{m} = \langle m_1, m_2, \ldots, m_k \rangle \in \mathbb{Z}_p^k$ and $\mathbf{m}' = \langle m_1', m_2', \ldots, m_k' \rangle \in \mathbb{Z}_p^k$ with $\mathbf{m} \neq \mathbf{m}'$. Define $\mathbf{a} = \langle a_1, a_2, \ldots, a_k \rangle = \mathbf{m} - \mathbf{m}'$. We have

$$\begin{aligned} & h_{\mathbf{x}}(\mathbf{m}) - h_{\mathbf{x}}(\mathbf{m}') = 0 \\ \Leftrightarrow & \sum_{i=1}^{k} m_i x_i^2 - \sum_{i=1}^{k} m_i' x_i^2 \equiv 0 \pmod{p} \\ \Leftrightarrow & \sum_{i=1}^{k} a_i x_i^2 \equiv 0 \pmod{p}. \end{aligned}$$

Since $\mathbf{m} \neq \mathbf{m}'$, not all coordinates of $\mathbf{a}$ are zero. Let $\mathbf{a}$ have j non-zero coordinates, where $1 \leq j \leq k$. Denote these non-zero coordinates as $a_{i_1}, \ldots, a_{i_j}$.

To bound the collision probability, we must find the maximum number of solutions $\mathbf{x} \in \mathbb{Z}_p^k$ to the congruence above over all choices of $\mathbf{a} \in \mathbb{Z}_p^k \setminus \{\mathbf{0}\}$. All non-zero coefficients are coprime to p, so by Theorem 4.9, the number of solutions to the above congruence is exactly

$$N_k(0,p) = p^{k-j} N_j(0,p).$$

So all that remains is to bound $N_j(0,p)$. Using Theorem 4.7, we can split the proof into the case where j is even and the case where j is odd. First, suppose that j is even. Since j is even and $1 \leq j \leq k$, we have $j = 2m$ for some integer $m \geq 1$. By Theorem 4.7, we have the following bound:

$$\begin{aligned} N_j(0,p) &= p^{2m-1} \sum_{d \mid p} \left(\frac{(-1)^m a_{i_1} \cdots a_{i_j}}{d} \right) \frac{\varphi(d)}{d^m} \\ &= p^{2m-1} \left[\left(\frac{(-1)^m a_{i_1} \cdots a_{i_j}}{1} \right) \frac{\varphi(1)}{1^m} + \left(\frac{(-1)^m a_{i_1} \cdots a_{i_j}}{p} \right) \frac{\varphi(p)}{p^m} \right] \\ &= p^{2m-1} \left[1 + \left(\frac{(-1)^m a_{i_1} \cdots a_{i_j}}{p} \right) \frac{p-1}{p^m} \right] \\ &\le p^{2m-1} \left[1 + \frac{p-1}{p^m} \right] \\ &= p^{2m-1} + p^m - p^{m-1}. \end{aligned}$$

Since $|\mathbb{Z}_p^k| = p^k$, the probability of two distinct messages colliding over a randomly chosen $\mathbf{x}$ is

$$\begin{aligned} \frac{p^{k-j}(p^{2m-1} + p^m - p^{m-1})}{p^k} &= \frac{p^{2m-1} + p^m - p^{m-1}}{p^j} \\ &= \frac{p^{2m-1} + p^m - p^{m-1}}{p^{2m}} \\ &= \frac{1}{p} + \frac{p-1}{p^{m+1}}, \end{aligned}$$

which is maximized when $m = 1$. Thus the probability of collision in this case is bounded above by

$$\frac{1}{p} + \frac{p-1}{p^2} = \frac{2}{p} - \frac{1}{p^2}.$$

Now suppose that j is odd, and thus $j = 2m+1$ for some integer $m \ge 0$. This is precisely the proof of Theorem 4.10, where we got a collision bound of $\frac{1}{p}$.

The proof is now complete. □

Interestingly, QH is also almost Δ-universal.

Theorem 4.12. QH *is* $\frac{2}{p}$*-almost-*Δ*-universal.*

Proof. Let $\mathbf{m} = \langle m_1, m_2, \ldots, m_k \rangle \in \mathbb{Z}_p^k$ and $\mathbf{m}' = \langle m'_1, m'_2, \ldots, m'_k \rangle \in \mathbb{Z}_p^k$ with $\mathbf{m} \neq \mathbf{m}'$. Define $\mathbf{a} = \langle a_1, a_2, \ldots, a_k \rangle = \mathbf{m} - \mathbf{m}'$. For every $b \in \mathbb{Z}_p$, we have

$$\begin{aligned} & h_{\mathbf{x}}(\mathbf{m}) - h_{\mathbf{x}}(\mathbf{m}') = b \\ \Leftrightarrow & \sum_{i=1}^{k} m_i x_i^2 - \sum_{i=1}^{k} m'_i x_i^2 \equiv b \pmod{p} \\ \Leftrightarrow & \sum_{i=1}^{k} a_i x_i^2 \equiv b \pmod{p}. \end{aligned}$$

Since $\mathbf{m} \neq \mathbf{m}'$, not all coordinates of $\mathbf{a}$ are zero. Let $\mathbf{a}$ have j non-zero coordinates, where $1 \le j \le k$. Denote these non-zero coordinates as $a_{i_1}, \ldots, a_{i_j}$.

To bound the collision probability, we must find the maximum number of solutions $\mathbf{x} \in \mathbb{Z}_p^k$ to the congruence above over all choices of $\mathbf{a} \in \mathbb{Z}_p^k \setminus \{\mathbf{0}\}$ and all $b \in \mathbb{Z}_p$. Note that the case $b = 0$ was proved earlier in Theorem 4.11, which implies that $\left(\frac{2}{p} - \frac{1}{p^2}\right)$ is an upper bound on the collision probability in this case. So assume that $b \in \mathbb{Z}_p \setminus \{\mathbf{0}\}$, which means that $\gcd(b, p) = 1$. Then by Theorem 4.9, we have that the number of solutions is

$$p^{k-j} N_j(b, p).$$

We must once again split the proof into the case where j is odd and the case where j is even. First suppose that j is even. Since j is even and $1 \leq j \leq k$, we have $j = 2m$ for some integer $m \geq 1$. Then applying Theorem 4.8 we see that

$$\begin{aligned}
N_j(b,p) &= p^{2m-1} \sum_{d \mid p} \left(\frac{(-1)^m a_{i_1} \cdots a_{i_j}}{d}\right) \frac{\mu(d)}{d^m} \\
&= p^{2m-1} \left[\left(\frac{(-1)^m a_{i_1} \cdots a_{i_j}}{1}\right) \frac{\mu(1)}{1^m} + \left(\frac{(-1)^m a_{i_1} \cdots a_{i_j}}{p}\right) \frac{\mu(p)}{p^m}\right] \\
&= p^{2m-1} \left[1 + \left(\frac{(-1)^m a_{i_1} \cdots a_{i_j}}{p}\right) \frac{-1}{p^m}\right] \\
&\leq p^{2m-1} \left[1 + \frac{1}{p^m}\right] \\
&= p^{2m-1} + p^{m-1}.
\end{aligned}$$

Since $|\mathbb{Z}_p^k| = p^k$, the probability of two distinct messages colliding over a randomly chosen $\mathbf{x}$ is

$$\begin{aligned}
\frac{p^{k-j}(p^{2m-1} + p^{m-1})}{p^k} &= \frac{p^{2m-1} + p^{m-1}}{p^j} \\
&= \frac{p^{2m-1}}{p^{2m}} + \frac{p^{m-1}}{p^{2m}} \\
&= \frac{1}{p} + \frac{1}{p^{m+1}},
\end{aligned}$$

which is maximized when $m = 1$. So $\frac{1}{p} + \frac{1}{p^2}$ is an upper bound on the collision probability in this case.

Now suppose that j is odd. Since j is odd and $1 \leq j \leq k$, we have $j = 2m + 1$ for some integer $m \geq 0$. Applying Theorem 4.8 we get

$$\begin{aligned}
N_j(b,n) &= p^{2m} \sum_{e \mid p} \left(\frac{(-1)^{m+1} a_{i_1} \cdots a_{i_j} \cdot b}{e}\right) \frac{\mu^2(e)}{e^m} \\
&= p^{2m} \left[\left(\frac{(-1)^{m+1} a_{i_1} \cdots a_{i_j} \cdot b}{1}\right) \frac{\mu^2(1)}{1^m} + \left(\frac{(-1)^{m+1} a_{i_1} \cdots a_{i_j} \cdot b}{p}\right) \frac{\mu^2(p)}{p^m}\right] \\
&= p^{2m} \left[1 + \left(\frac{(-1)^{m+1} a_{i_1} \cdots a_{i_j} \cdot b}{p}\right) \frac{1}{p^m}\right] \\
&\leq p^{2m} \left[1 + \frac{1}{p^m}\right] \\
&= p^{2m} + p^m.
\end{aligned}$$

Now, the probability of two distinct messages colliding over a randomly chosen $\mathbf{x}$ is

$$\begin{aligned}\frac{p^{k-j}(p^{2m}+p^{m})}{p^{k}} &= \frac{p^{2m}+p^{m}}{p^{j}}\\ &= \frac{p^{2m}+p^{m}}{p^{2m+1}}\\ &= \frac{1}{p}+\frac{1}{p^{m+1}},\end{aligned}$$

which is maximized when $m = 0$. So $\frac{2}{p}$ is an upper bound on the collision probability in this case.

Now comparing the three upper bounds we have found, we get an upper bound for all cases of $\max\{\frac{2}{p}-\frac{1}{p^2},\frac{1}{p},\frac{2}{p}\}=\frac{2}{p}$. □

4.3 Comparing QH and OQH with PH

A hash function in QH hashes the message $\mathbf{m}$ as $\sum_{i=1}^{k} m_i x_i^2 \pmod{p}$, but the values x_i^2 only need to be computed the first time $\mathbf{x}$ is used, and can be stored for subsequent evaluations of the hash function. Furthermore, while $\frac{2}{p}$ is an absolute bound on the differential probability of QH, as the Hamming distance between messages increases, the differential probability approaches $\frac{1}{p}$, so in practice, we can expect the differential probability to be somewhere between $\frac{1}{p}$ and $\frac{2}{p}$. Additionally, when the Hamming distance is odd, which should occur roughly half of the time, collision probability is exactly $\frac{1}{p}$. So the collision probability of QH will be close to $\frac{1}{p}$ in practice, while for Polynomial Hash (PH) the collision probability is $\frac{d}{p}$, where d can be quite large depending on the length of the message.

5 Converting ε-AΔU to ε-ASU

In this section, we prove a general result using which we can convert any ε-AΔU family to an ε-ASU family. Our result is a generalization of the following result by Etzel *et al.* [64] which seems to have remained underappreciated.

Theorem 5.1. *Let the family*

$$H = \{h_k \,:\, D \to R \,|\, k \in K\}$$

be a Δ-universal family of hash functions, where K is the key space and R is a finite additive Abelian group. Then the family

$$H' = \{h'_{k,w} \,:\, D \to R \,|\, k \in K, w \in R\},$$

where

$$h'_{k,w}(x) = h_k(x) + w,$$

and '+' denotes the group addition operation, is strongly universal.

In order to generalize the above result, we also need the following result (see [65]):

Theorem 5.2. *Let G be an Abelian group, and let $\xi_1, \xi_2, \ldots, \xi_t$ be independent random variables which take on values in G. If one of ξ_i is uniformly distributed in G, then the sum $\xi_1 + \xi_2 + \cdots + \xi_t$ is also uniformly distributed in G.*

More generally, Sherstnev [65] gave necessary and sufficient conditions on the distributions of independent random variables $\xi_1, \xi_2, \ldots, \xi_t$, taking on values in an Abelian group G, under which the sum $\xi_1 + \xi_2 + \cdots + \xi_t$ is uniformly distributed in G.

Now, we are ready to prove our result.

Theorem 5.3. *Let the family*

$$H = \{h_k : D \to R \mid k \in K\}$$

be an ε-almost-Δ-universal family of hash functions, where K is the key space and R is a finite additive Abelian group. Then the family

$$H' = \{h'_{k,w} : D \to R \mid k \in K, w \in R\},$$

where

$$h'_{k,w}(x) = h_k(x) + w,$$

and '+' denotes the group addition operation, is ε-almost-strongly universal.

Proof. For any two distinct $x, y \in D$, and all $a, b \in R$, we have

$$\begin{aligned}
&\mathrm{Pr}_{h'_{k,w} \leftarrow H'}[h'_{k,w}(x) = a,\ h'_{k,w}(y) = b] \\
&= \mathrm{Pr}_{h'_{k,w} \leftarrow H'}[h_k(x) + w = a,\ h_k(y) + w = b] \\
&= \mathrm{Pr}_{h'_{k,w} \leftarrow H'}[h_k(x) - h_k(y) = a - b,\ w = a - h_k(x)] \\
&= \mathrm{Pr}_{h'_{k,w} \leftarrow H'}[h_k(x) - h_k(y) = a - b] \cdot \mathrm{Pr}_{h'_{k,w} \leftarrow H'}[a = w + h_k(x)].
\end{aligned}$$

Since H is ε-almost-Δ-universal, we have

$$\mathrm{Pr}_{h'_{k,w} \leftarrow H'}[h_k(x) - h_k(y) = a - b] \le \varepsilon.$$

Also, by Theorem 5.2 we have

$$\mathrm{Pr}_{h'_{k,w} \leftarrow H'}[a = w + h_k(x)] = \frac{1}{|R|}.$$

Consequently,

$$\mathrm{Pr}_{h'_{k,w} \leftarrow H'}[h'_{k,w}(x) = a,\ h'_{k,w}(y) = b] \le \frac{\varepsilon}{|R|}.$$

Hence, the result follows. □

Corollary 5.4. *Using Theorem 5.3, we can convert any ε-almost-Δ-universal family, in particular the ε-almost-Δ-universal families studied in this paper, to ε-almost-strongly universal families and so make them useful for applications in QKD and many other areas. This can be done by adding a uniform value $w \leftarrow R$ to the hash functions, where R is the range of the corresponding hash functions.*

Conclusion

In contrast to the key agreement protocols in public-key cryptography which their security relies on the computational difficulty of certain mathematical problems, QKD relies on the foundations of quantum mechanics and so provides a higher level of security. Authentication schemes play a critical role in QKD, as secure communication is impossible without it, since otherwise an adversary could stand in the middle and intercept all communications without the legitimate parties realizing it. The information-theoretically secure Wegman–Carter construction is of particular importance, as it is completely secure even against an adversary with unbounded computational power. However, the authentication scheme must only remain unbroken during the execution of the QKD protocol to guarantee *everlasting security of the generated key.* Price *et al.* [6] suggest using the computationally secure WCBK paradigm for authentication in QKD. This not only increases the efficiency of key material used, but also significantly increases resilience against certain DoS attacks. We propose using the 1K-DWCDM scheme and its variants for this purpose instead, as they offer increased security over WCBK for a minimal cost in performance.

We also introduced QH and OQH and analyzed their collision properties using results from Cohen [11]. The family QH is $\frac{2}{p}$-AΔU, where p is the prime modulus. While $\frac{2}{p}$ is an upper bound on the differential probability, it is based on the worst case, when the Hamming distance between strings is minimal. Thus, in practice, the differential probability should be lower on average. Interestingly, in the case of OQH, when all message pairs have an odd Hamming distance (which should occur roughly half of the time), collision probability is exactly $\frac{1}{p}$. This should also lower the collision probability of QH in practice.

We also generalize the method of Etzel *et al.* [64] to convert ε-AΔU families to ε-ASU families. The latter families can be suitably applied wherever almost strongly universal hashing is needed, such as authentication schemes in QKD.

Finally, as stated by Price *et al.* [6], we also believe that the intersection of modern and quantum cryptography should be explored more thoroughly, as it is largely untapped, and has the potential to offer improvements to real-world algorithms in both fields.

Acknowledgements

The authors would like to thank the editor and the referees for carefully reading the paper, and for their useful comments which helped improve the paper.

References

1. R. Allaume, C. Branciard, J. Bouda, T. Debuisschert, M. Dianati, N. Gisin, M. Godfrey, P. Grangier, T. Lnger, N. Lütkenhaus, C. Monyk, P. Painchault, M. Peev, A. Poppe, T. Pornin, J. Rarity, R. Renner, G. Ribordy, M. Riguidel, L. Salvail, A. Shields, H. Weinfurter, and A. Zeilinger. Using quantum key distribution for cryptographic purposes: A survey. *Theoretical Computer Science*, 560:62 – 81, 2014. Theoretical Aspects of Quantum Cryptography celebrating 30 years of BB84.
2. D. Stebila, M. Mosca, and N. Lütkenhaus. The case for quantum key distribution. In A. Sergienko, S. Pascazio, and P. Villoresi, editors, *Quantum Communication and Quantum Networking*, pages 283–296, 2010.
3. D. Unruh. Everlasting multi-party computation. In R. Canetti and J. A. Garay, editors, *Advances in Cryptology – CRYPTO 2013*, pages 380–397, 2013.

4. M. Sasaki. Quantum key distribution and its applications. *IEEE Security & Privacy*, 16(5):42–48, 2018.
5. M. N. Wegman and J. L. Carter. New hash functions and their use in authentication and set equality. *Journal of Computer and System Sciences*, 22(3):265–279, June 1981.
6. A.B. Price, J.G. Rarity, and C. Erven. A quantum key distribution protocol for rapid denial of service detection. *EPJ Quantum Technology*, 7(8), 2020.
7. M. Sasaki, M. Fujiwara, H. Ishizuka, W. Klaus, K. Wakui, M. Takeoka, S. Miki, T. Yamashita, Z. Wang, A. Tanaka, K. Yoshino, Y. Nambu, S. Takahashi, A. Tajima, A. Tomita, T. Domeki, T. Hasegawa, Y. Sakai, H. Kobayashi, T. Asai, K. Shimizu, T. Tokura, T. Tsurumaru, M. Matsui, T. Honjo, K. Tamaki, H. Takesue, Y. Tokura, J. F. Dynes, A. R. Dixon, A. W. Sharpe, Z. L. Yuan, A. J. Shields, S. Uchikoga, M. Legré, S. Robyr, P. Trinkler, L. Monat, J.-B. Page, G. Ribordy, A. Poppe, A. Allacher, O. Maurhart, T. Länger, M. Peev, and A. Zeilinger. Field test of quantum key distribution in the tokyo QKD network. *Opt. Express*, 19(11):10387–10409, May 2011.
8. D. Stucki, M. Legré, F. Buntschu, B. Clausen, N. Felber, N. Gisin, L. Henzen, P. Junod, G. Litzistorf, P. Monbaron, L. Monat, J-B. Page, D. Perroud, G. Ribordy, A. Rochas, S. Robyr, J. Tavares, R. Thew, P. Trinkler, S. Ventura, R. Voirol, N. Walenta, and H. Zbinden. Long-term performance of the SwissQuantum quantum key distribution network in a field environment. *New Journal of Physics*, 13(12):123001, Dec 2011.
9. C. Portmann. Key recycling in authentication. *IEEE Transactions on Information Theory*, 60(7):4383–4396, 2014.
10. R. Canetti. Universally composable security: a new paradigm for cryptographic protocols. In *Proceedings 42nd IEEE Symposium on Foundations of Computer Science*, pages 136–145, 2001.
11. E. Cohen. Rings of arithmetic functions. II: The number of solutions of quadratic congruences. *Duke Mathematical Journal*, 21(1):9–28, March 1954.
12. N. Datta, A. Dutta, M. Nandi, and K. Yasuda. Encrypt or decrypt? to make a single-key beyond birthday secure nonce-based MAC. In H. Shacham and A. Boldyreva, editors, *Advances in Cryptology – CRYPTO 2018*, pages 631–661, 2018.
13. J. L. Carter and M. N. Wegman. Universal classes of hash functions. *Journal of Computer and System Sciences*, 18(2):143–154, April 1979.
14. C. H. Bennett and G. Brassard. Quantum cryptography: Public key distribution and coin tossing. *Theoretical Computer Science*, 560:7–11, 2014.
15. J. Black, S. Halevi, H. Krawczyk, T. Krovetz, and P. Rogaway. UMAC: Fast and secure message authentication. In *Advances in Cryptology – CRYPTO'99*, volume 1666 of *Lecture Notes in Computer Science*, pages 216–233, 1999.
16. Y. Dodis, L. Reyzin, and A. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In *Advances in Cryptology - EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 523–540, 2004.
17. S. Halevi and H. Krawczyk. MMH: software message authentication in the Gbit/second rates. In E. Biham, editor, *Fast Software Encryption – FSE'97*, volume 1267 of *Lecture Notes in Computer Science*, pages 172–189, 1997.
18. E. N. Gilbert, F. J. Macwilliams, and N. J. A. Sloane. Codes which detect deception. *The Bell System Technical Journal*, 53(3):405–424, March 1974.
19. H. Handschuh and B. Preneel. Key-recovery attacks on universal hash function based MAC algorithms. In D. Wagner, editor, *Advances in Cryptology CRYPTO'08*, Lecture Notes in Computer Science, pages 144–161, 2008.
20. M. Hayashi. General nonasymptotic and asymptotic formulas in channel resolvability and identification capacity and their application to the wiretap channel. *IEEE Transactions on Information Theory*, 52(4):1562–1575, April 2006.
21. M. Hayashi. Exponential decreasing rate of leaked information in universal random privacy amplification. *IEEE Transactions on Information Theory*, 57(6):3989–4001, June 2011.
22. R. Renner and S. Wolf. Simple and tight bounds for information reconciliation and privacy amplification. In *Advances in Cryptology – ASIACRYPT 2005*, Lecture Notes in Computer Science,

pages 199–216, 2005.

23. P. Rogaway. Bucket hashing and its application to fast message authentication. In D. Coppersmith, editor, *Advances in Cryptology CRYPTO 95*, volume 12 of *Lecture Notes in Computer Science*, pages 29–42, 1995.
24. H. Tyagi and A. Vardy. Universal hashing for information-theoretic security. *Proceedings of the IEEE*, 103(10):1781–1795, October 2015.
25. R. Cramer, I. B. Damgård, N. Döttling, S. Fehr, and G. Spini. Linear secret sharing schemes from error correcting codes and universal hash functions. In E. Oswald and M. Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part II*, volume 9057 of *Lecture Notes in Computer Science*, pages 313–336. Springer, 2015.
26. D. R. Stinson. On the connections between universal hashing, combinatorial designs and error-correcting codes. *Congressus Numerantium*, 114:7–27, 1996.
27. J. Håstad, R. Impagliazzo, L. Levin, and M. Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, January 1999.
28. N. Nisan. Pseudorandom generators for space-bounded computations. *Combinatorica*, 12(4):449–461, 1992.
29. S. Rudich and A. Wigderson. *Computational Complexity Theory*, volume 10 of *IAS/Park City Mathematics Series.* American Mathematical Society, August 2004. ISSN: 1079-5634, 2472-5064.
30. M. Sipser. A complexity theoretic approach to randomness. In *ACM Symposium on Theory of Computing - STOC'83*, STOC 83, page 330335, 1983.
31. R. Impagliazzo and D. Zuckerman. How to recycle random bits. In *Symposium on Foundations of Computer Science – SFCS'89*, pages 248–253, October 1989.
32. R. Motwani and P. Raghavan. *Randomized Algorithms.* Cambridge University Press, 1995.
33. A. Pagh and R. Pagh. Uniform hashing in constant time and optimal space. *SIAM Journal on Computing*, 38(1):85–96, January 2008.
34. A. Siegel. On universal classes of extremely random constant-time hash functions. *SIAM Journal on Computing*, 33(3):505–543, January 2004.
35. H. Karloff, S. Suri, and S. Vassilvitskii. A model of computation for mapreduce. In *ACM-SIAM Symposium on Discrete Algorithms – SODA 10*, page 938948, 2010.
36. C. E. Leiserson, T. B. Schardl, and J. Sukha. Deterministic parallel random-number generation for dynamic-multithreading platforms. In *ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming – PPoPP'12*, pages 193–204, 2012.
37. R. Ritchie and K. Bibak. SQUAREMIX: A faster pseudorandom number generator for dynamic-multithreading platforms. In *2020 Data Compression Conference (DCC)*, pages 391–391, 2020.
38. H. Krawczyk. LFSR-based hashing and authentication. In Y. G. Desmedt, editor, *Advances in Cryptology CRYPTO 94*, Lecture Notes in Computer Science, pages 129–139, 1994.
39. D. R. Stinson. Universal hashing and authentication codes. *Designs, Codes and Cryptography*, 4(3):369–380, July 1994.
40. ISO/IEC 9797-1:2011. Information technology – security techniques – message authentication codes (MACs) – part 1: Mechanisms using a block cipher, 2011.
41. J. Black and P. Rogaway. CBC MACs for arbitrary-length messages: The three-key constructions. In *Advances in Cryptology – CRYPTO'00*, volume 1880 of *Lecture Notes in Computer Science*, pages 197–215, 2000.
42. T. Iwata and K. Kurosawa. OMAC: One-key CBC MAC. In T. Johansson, editor, *Fast Software Encryption – FSE'03*, volume 2887 of *Lecture Notes in Computer Science*, pages 129–153, 2003.
43. T. Iwata and K. Kurosawa. OMAC: One-key CBC MAC – addendum. 04 2003.
44. J. Black and P. Rogaway. A block-cipher mode of operation for parallelizable message authentication. In *Advances in Cryptology – EUROCRYPT'02*, volume 2332 of *Lecture Notes in Computer Science*, pages 384–397, 2002.
45. M. Bellare, R. Canetti, and H. Krawczyk. Keying hash functions for message authentication. In N. Koblitz, editor, *Advances in Cryptology – CRYPTO 96*, Lecture Notes in Computer Science,

page 115, 1996.

46. D. A. McGrew and J. Viega. The security and performance of the Galois/Counter mode (GCM) of operation. In A. Canteaut and K. Viswanathan, editors, *Progress in Cryptology – INDOCRYPT 2004*, Lecture Notes in Computer Science, pages 343–355, 2005.
47. D.J. Bernstein. The Poly1305-AES message-authentication code. In *Fast Software Encryption – FSE'05*, volume 3557 of *Lecture Notes in Computer Science*, page 3249, 2005.
48. D. Boneh and V. Shoup. *A Graduate Course in Applied Cryptography.* 0.5 edition, 2020.
49. G. Brassard. On computationally secure authentication tags requiring short secret shared keys. In *Advances in Cryptology – CRYPTO '82*, pages 79–86, 1982.
50. V. Shoup. On fast and provably secure message authentication based on universal hashing. In N. Koblitz, editor, *Advances in Cryptology CRYPTO 96*, Lecture Notes in Computer Science, pages 313–328, 1996.
51. A. Joux. Authentication failures in NIST version of GCM. Comments submitted to NIST Modes of Operation Process, 2006.
52. B. Cogliati and Y. Seurin. EWCDM: An efficient, beyond-birthday secure, nonce-misuse resistant MAC. In M. Robshaw and J. Katz, editors, *Advances in Cryptology – CRYPTO 2016*, pages 121–149, 2016.
53. B. Mennink and S. Neves. Encrypted davies-meyer and its dual: Towards optimal security using mirror theory. In J. Katz and Hovav Shacham, editors, *Advances in Cryptology – CRYPTO 2017*, pages 556–583, 2017.
54. J. Patarin. Introduction to mirror theory: Analysis of systems of linear equalities and linear non equalities for cryptography. *IACR Cryptol. ePrint Arch.*, 2010:287, 2010.
55. J. Patarin. Mirror theory and cryptography. *Applicable Algebra in Engineering, Communication and Computing*, 28(4):321–338, 2017.
56. J. Patarin. The "coefficients H" technique. In R. M. Avanzi, L. Keliher, and F. Sica, editors, *Selected Areas in Cryptography*, pages 328–345, 2009.
57. S. Duval and G. Leurent. Lightweight MACs from universal hash functions. In *Smart Card Research and Advanced Applications*, volume 11833 of *Lecture Notes in Computer Science*, pages 195–215, 2020.
58. M. Dietzfelbinger, J. Gil, Y. Matias, and N. Pippenger. Polynomial hash functions are reliable. In W. Kuich, editor, *International Colloquium on Automata, Languages and Programming – ICALP'92*, pages 235–246, 1992.
59. B. den Boer. A simple and key-economical unconditional authentication scheme. *Journal of Computer Security*, 2:65–72, 1993.
60. J. Bierbrauer, T. Johansson, G. Kabatianskii, and B. Smeets. On families of hash functions via geometric codes and concatenation. In *Advances in Cryptology – CRYPTO'93*, volume 5665 of *Lecture Notes in Computer Science*, page 331342, 1993.
61. R. Taylor. An integrity check value algorithm for stream ciphers. In D. R. Stinson, editor, *Advances in Cryptology CRYPTO' 93*, volume 773 of *Lecture Notes in Computer Science*, pages 40–48, 1994.
62. K. Mehlhorn and U. Vishkin. Randomized and deterministic simulations of PRAMs by parallel machines with restricted granularity of parallel memories. *Acta Informatica*, 21(4):339–374, November 1984.
63. L. Tóth. Counting solutions of quadratic congruences in several variables revisited. *Journal of Integer Sequences*, 17:Article 14.11.6, 2014.
64. M. Etzel, S. Patel, and Z. Ramzan. Square hash: fast message authentication via optimized universal hash functions. In M. Wiener, editor, *Advances in Cryptology CRYPTO 99*, volume 1666 of *Lecture Notes in Computer Science*, pages 234–251, 1999.
65. V. I. Sherstnev. A random variable uniformly distributed on a finite Abelian group as a sum of independent summands. *Rossiĭskaya Akademiya Nauk. Teoriya Veroyatnosteĭ i ee Primeneniya*, 43(2):397–403, 1998.