

New bounds for almost universal hash functions

Abstract

Using combinatorial analysis and the pigeon-hole principle, we introduce a new lower (or combinatorial) bound for the key length in an almost (both pairwise and l -wise) universal hash function. The advantage of this bound is that the key length only grows in proportion to the logarithm of the message length as the collision probability moves away from its theoretical minimum by an arbitrarily small and positive value. Conventionally bounds for various families of universal hash functions have been derived by using the equivalence between pairwise universal hash functions and error-correcting codes, and indeed in this paper another (very similar) bound for almost universal hash functions can be calculated from the Singleton bound in coding theory. However, the latter, perhaps unexpectedly, will be shown to be not as tight as our combinatorial one. To the best of our knowledge, this is the first time that combinatorial analysis has been demonstrated to yield a better universal hash function bound than the use of the equivalence, and we will explain why there is such a mismatch. This work therefore potentially opens the way to re-examining many bounds for a number of families of universal hash functions, which have been solely derived from bounds in coding theory and/or other combinatorial objects.

1 Introduction and contribution

Universal hash function H with parameters (ϵ, K, M, b) was introduced by Carter and Wegman [9, 43]. Each family, which is indexed by a K -bit key k , consists of 2^K hash functions mapping a message representable by M bits into a b -bit hash output.

In this paper we use combinatorial analysis and the pigeon-hole principle to introduce a new bound, termed the *combinatorial* bound, for an *almost* pairwise universal hash function (AU). This result tells us the lower bound on the bitlength of the hash key in terms of a fixed amount of information we want to hash, the hash output bitlength and the hash collision probability ϵ . We also point out that this bound can be easily generalised to a l -wise AU -bound.

Although there has been much work in universal hash functions, most researchers concentrate on restricted versions of AU known as almost strongly and almost XOR universal hash functions (ASU and AXU), which are used to construct Message Authentication Codes (MAC) [1, 9, 12, 18, 19, 20, 36, 37, 43]. We however observe that the use of AU has been recently picked up by a number of new cryptographic applications and protocols. Most notably, there have been rapid developments in a *non-standard* authentication technology [2, 10, 13, 14, 23, 27, 28, 35, 41, 42] which make use of (short-output) AU and human interactions (or unspoofable authentication channels) to bootstrap secure communication to reduce the needs of PKI and passwords in pervasive computing environments. This new technology has been studied by many researchers to create secure communication among a group of devices [24, 27, 41], in financial transactions [32] (e.g. as in online or telephone banking, and CHIP&PIN technology) and telemedicine. In many protocols of this type, including Bluetooth v2.1 [25], MANA I [13, 14] and (S)HCBK [27], human owners of some

electronic devices would need to compare manually a digest or hash-value of the data on which these devices try to agree in combination with some fresh and random keys. Since the keys are always fresh and randomly generated in every protocol run, substitution and key-recovery attacks, which rely on the reuse of a single key in multiple sessions (as in MAC), are irrelevant. What we then require of such a universal hash function is a protection against collision attacks (*AU*) as opposed to substitution attacks (*ASU*), and hence *ASU* and *AXU* are not needed. Moreover in some of these schemes universal-hash keys need to be transmitted over a low-bandwidth authentication channel, e.g. this is the case of MANA I protocol introduced and standardised by Gehrman, Mitchell and Nyberg [13, 14], and consequently we want to reduce the key length as much as possible without compromising the security. Of course, to do this we have to know the lower bound of key length in an *AU* with respect to authenticated message length, the universal hash function output length, and the collision probability.

There is another problem with many universal hash function constructions in practice. As universal hash key length is usually of significant size, one tends to reuse a single secret (long) key for multiple messages over a long period of time. This opens the way for key recovery and universal forgery attacks which exploit weak key properties or partial information on a secret key; such attacks have been recently reported by Handschuh and Preneel [15]. Avoiding reusing keys would render most key recovery attacks useless, and so it is desirable to construct universal hash functions with short keys, which in turn generate the need to calculate the lower bound of universal hash key length.

An important contribution in this area was the discovery of an equivalence between pairwise universal hash functions and error-correcting codes (ECC) due to Johansson et al. [17]. This implies that every bound of coding theory potentially corresponds to another bound for (almost pairwise) universal hash functions, and vice versa. So far the only pairwise *AU*-bound, derived from the Plotkin bound in coding theory, is due to Stinson [36, 37]. However this is known to be tight in an extremely narrow range of ϵ [29] (i.e. arbitrarily close to the theoretical minimum of 2^{-b} as shown in Section 4.1)¹, and within that range the key length rises at least linearly with the message length, i.e. an unattractive prospect for the use of *AU* in practice. We therefore find it very strange that, apart from our *AU*-bounds introduced in this paper, there has not been any work finding a tight *AU*-bound when ϵ moves away from 2^{-b} by an arbitrarily small positive value.

In Section 3.2, we derive another pairwise *AU*-bound by using the *Singleton* bound [30] in coding theory. However, this bound turns out to be not as tight as our combinatorial one as seen by an elementary construction of an *AU*. Theoretically, we will prove that there exists a subclass of an *AU* which cannot be transformed into an equivalent code that satisfies the Singleton bound with equality, thus the Singleton bound does not give a tight result for this subclass of an *AU*. But perhaps paradoxically, we find that this does *not* imply that the Singleton bound can be improved.

To the best of our knowledge, this is the first time that combinatorial analysis can produce a tighter universal hash bound than the equivalence between ECC and universal hash functions which has been used by many researchers to derive bounds for not only *AU* but also *AXU* and *ASU*. We hope that by using combinatorial analysis one can further verify the accuracy of many existing bounds for other families of universal hash functions which have only been derived from bounds of error-correcting codes (ECC-bounds) [37, 38, 12, 18] or other combinatorial objects

¹In practice, the minimum collision probability of an *AU* is $\frac{2^M - 2^b}{2^{M+b} - 2^b}$, which is less than 2^{-b} . This occurs in an *optimally universal* hash scheme introduced by Sarwate [33].

such as difference matrices [37], orthogonal or perpendicular arrays [21, 37, 38, 39], and balanced incomplete block designs [21, 31, 36, 37, 39]. Another strength of combinatorial analysis is its ability to derive bounds for not only pairwise but also more general l -wise universal hash functions; this has not been the case with the equivalence which only applies to pairwise universal hash functions.

As we will see in theorems 1 and 4, the key lengths of both our AU -bounds grow in proportion to the logarithm of message length when ϵ is greater than 2^{-b} by an arbitrarily small positive value, and thus these results can lead to new AU constructions which are more usable in practice. Please note that the strange and asymptotic behaviour of the universal hash function key length with respect to the value of ϵ has previously been known as the ‘‘Wegman-Carter effect’’ [7, 22] that will be quantified by a threshold value of ϵ introduced in Section 4.1. In this respect, our contribution is therefore to nail down exactly the formula of the AU -bound which matches the asymptotic behaviour of universal hash functions previously known in the literature.

We end this paper by demonstrating the optimality of polynomial hashing over finite field [8, 17, 40] in building an AU , i.e. the construction meets the combinatorial AU -bound with equality. This therefore extends the work of Johansson, Kabatianskii and Smeets [17, 18], where the authors proved the asymptotic optimality of polynomial hashing as an ASU .

In our work, we also introduce a new bound for an AXU , which can be derived from Kabatianskii’s ASU -bound [18] and a connection between ASU and AXU [43, 11]. This bound is then rigorously analysed in relation to other known bounds and will be shown to be met with equality in the second version of polynomial hashing.

2 Notations and definitions of universal hash functions

In this paper, all formulas are expressed in terms of the generalised bitlengths of hash keys, input messages and hash output instead of the cardinalities of the sets of these parameters (2^K , 2^M and 2^b) as in other papers. Although the bitlengths are often integers in practice, our result reported here applies to both integer and non-integer bitlengths. The advantage of the notation will become clear when we explain why combinatorial analysis yields better bounds than the use of coding theory bounds in Section 3.2.

Let us recall the definitions of a number of families of pairwise universal hash functions. Here ϵ is the collision, differential or interpolation probability associated with ϵ - AU , ϵ - AXU or ϵ - ASU , respectively.² In all the following definitions, we look at the probability of some condition being met, e.g. hash collision, as the key k varies uniformly over its domain of $[1, 2^K]$: $\Pr_{\{k \in [1, 2^K]\}}[\]$.

An ϵ-almost universal hash function, ϵ-AU (K, M, b) [9, 36]
H is an ϵ - AU iff for every pair of distinct messages (m, \hat{m}) : $\Pr_{\{k \in [1, 2^K]\}}[h_k(m) = h_k(\hat{m})] \leq \epsilon$

An ϵ-almost XOR universal hash function, ϵ-AXU (K, M, b) [19, 20, 36]
H is an ϵ - AXU iff for every pair of distinct messages (m, \hat{m}) and any $\omega \in \{0, 1\}^b$: $\Pr_{\{k \in [1, 2^K]\}}[h_k(m) \oplus h_k(\hat{m}) = \omega] \leq \epsilon$

²The terms collision, differential and interpolation probabilities were introduced by Bernstein in the appendix of [4] to distinguish the differences between these families of universal hash functions.

<p>An ϵ-almost strongly universal hash function, ϵ-ASU (K, M, b) [43, 36]</p> <p>(a) For every pair of a message m and a hash output y: $\Pr_{\{k \in [1, 2^K]\}}[h_k(m) = y] \leq 2^{-b}$.</p> <p>(b) For every pair of distinct messages (m, \hat{m}) and for every pair of hash outputs (y, \hat{y}): $\Pr_{\{k \in [1, 2^K]\}}[h_k(m) = y, h_k(\hat{m}) = \hat{y}] \leq \epsilon 2^{-b}$</p>
--

All *universal* hash functions discussed to date are pairwise, since we look at their properties in relation to two different messages. We will see that the combinatorial bound, and its proof, can be easily adapted to a more general version of AU , termed a l -wise AU_l , and therefore we give the definition below. We argue that not only is this of theoretical interest to study AU_l , but also useful in many applications. For example, in many group protocols [27, 28, 41] of the non-standard authentication technology discussed in the introduction, the intruder always attempts to fool multiple parties into accepting different versions of a piece of data that the protocol seeks to ensure they agree on. It is therefore desirable that we consider the possibility of a hash collision with respect to more than two different input messages. However, unless indicated, our work presented in this paper always refers to pairwise universal hash functions.

<p>A l-wise ϵ-almost universal hash function, ϵ-AU_l (K, M, b)</p> <p>H is an ϵ-AU_l iff for any l distinct messages $\{m_1, \dots, m_l\}$:</p> <p>$\Pr_{\{k \in [1, 2^K]\}}[h_k(m_1) = \dots = h_k(m_l)] \leq \epsilon$</p>

We assume the input message bitlength M is significantly greater than the hash bitlength b . Whenever we use the term $\log X$, we refer to the base-2 logarithm to simplify the notation.

3 Bounds for almost universal hash functions

We first present a new bound for an AU using combinatorial analysis. Subsequently, the Singleton bound [30] in coding theory will be used to derive another AU -bound,³ which is not as tight as we had expected. In particular, when M is *not* an integer multiple of b , the combinatorial AU -bound is tighter (greater) than the latter. This result applies to both integer and non-integer values of M and b .

This therefore demonstrates that a universal hash bound derived from coding theory might only fit certain of universal hash parameters tightly, and for further parameter values the bound is based on the best conservative approximate that does fit coding theory. What we will discover is that combinatorial analysis can provide a tighter bound for this second class.

3.1 Combinatorial AU -bound

Theorem 1 *If there exists an ϵ - AU (K, M, b) then*

- *when M is an integer multiple of b : $K \geq \log(\epsilon^{-1} (\frac{M}{b} - 1))$*
- *when M is not an integer multiple of b : $K \geq \log(\epsilon^{-1} \lfloor \frac{M}{b} \rfloor)$*

³Although several bounds in coding theory have been transformed into equivalent bounds for universal hash functions, e.g. Plotkin [37] or Johnson bounds [18], to the best of our knowledge, the Singleton bound has never been used.

Proof We make use of the *pigeon-hole* principle: given two positive integers n and m , if n items are put into m holes then at least one hole must contain more than or equal to $\lceil n/m \rceil$ items.

Let assume $M = bt + b'$, where t is an integer and $0 \leq b' < b$.

For any key k_1 , there exists a hash value h_1 such that there are at least $\lceil 2^{M-b} \rceil$ distinct messages all hashing to h_1 under the same key k_1 , thanks to the pigeon-hole principle. For any choice of k_2 other than k_1 , there will also be a collection of at least $\lceil 2^{M-2b} \rceil$ of these messages mapping to some hash value h_2 under k_2 . Repeating this process $(t-1)$ times will result in at least $\lceil 2^{M-(t-1)b} \rceil = \lceil 2^{b+b'} \rceil$ distinct messages that hash to the same values under $(t-1)$ keys, leading to two possibilities.

- When $b' = 0$ or M is an integer multiple of b , we *cannot* repeat this process any further because at least 2 distinct messages must be left after these iterations. Thus a family of hash functions is ϵ -almost universal when $(t-1)$ is smaller than or equal to ϵ portion of the key space: $\epsilon 2^K \geq t-1 = M/b - 1$, which means that $K \geq \log(\epsilon^{-1}(M/b - 1))$
- When $b' > 0$ or M is *not* an integer multiple of b , we have $\lceil 2^{b+b'} \rceil \geq 2^b + 1$. Repeating this process for one more key will end up with at least 2 distinct messages that map to the same values under $t = \lfloor M/b \rfloor$ keys. As a result, we have $\epsilon 2^K \geq \lfloor M/b \rfloor$, which means that $K \geq \log(\epsilon^{-1} \lfloor M/b \rfloor)$ ■

The proof of the combinatorial bound for a pairwise AU_2 can be generalised to derive the corresponding bound for a l -wise AU_l , for any integer $l \geq 2$. Instead of leaving at least 2 distinct messages after these iterations as shown in the proof of Theorem 1, we need to leave at least l distinct messages. Similar analysis leads to the following theorem.

Theorem 2 *If there exists a l -wise ϵ - AU_l (K, M, b) and $M = bt + b'$, where t is an integer, then*

- when $0 \leq b' \leq \log(l-1)$: $K \geq \log(\epsilon^{-1} (\lfloor \frac{M}{b} \rfloor - 1))$
- when $\log(l-1) < b' < b$: $K \geq \log(\epsilon^{-1} \lfloor \frac{M}{b} \rfloor)$

Although there has been some study of l -wise *almost strongly* universal hash functions by Stinson [38] and Kurosawa et al. [21], as far as we are aware, this is the first result on l -wise *almost* universal hash functions.

We end this section with another observation: there is no restriction on any of the parameters, i.e. the generalised bitlengths (M, b, r) , in both our pairwise and l -wise combinatorial AU -bounds, which makes them more attractive than a similar ASU -bound introduced by Kabatianskii et al. [18], as will be discussed in the sections to come.

3.2 Error-correcting codes and almost universal hash functions

While the connection between almost universal hash functions and error-correcting codes (i.e. see Theorem 3), which was first observed by Johansson et al. [17], has often been used by researchers to derive tight bounds for universal hash functions [36, 37, 17, 18], the following comparative analysis will demonstrate that this strategy does not always give the best answer.

Let (n, T, d, q) be a q -ary error-correcting code, where n is the codeword length in symbols, T is the total number of codewords, and the minimum Hamming distance is d .

	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8
k_1	1	2	3	4	1	2	3	4
k_2	2	3	4	1	3	4	1	2

Table 1: A construction of an $(\epsilon = 1/2)$ - AU ($K = 1, M = 3, b = 2$), in which there are $2^M = 8$ input messages $\{m_1, \dots, m_8\}$ mapping onto $2^b = 4$ hash outputs $\{1, 2, 3, 4\}$ under $2^K = 2$ hash keys $\{k_1, k_2\}$. The elements in this table are the values of hash outputs, e.g. $h_{k_1}(m_3) = 3$.

Theorem 3 [17, 6, 37]. *If there exists an ϵ - AU (K, M, b), then there exists an $(n = 2^K, T = 2^M, d = n - \epsilon 2^K, q = 2^b)$ code. Conversely, if there exists an (n, T, d, q) code, then there exists an $(\epsilon = 1 - d/n)$ - AU ($K = \log n, M = \log T, b = \log q$).*

Using the connection, we can derive another AU -bound from the Singleton bound.

Singleton bound [30]: given an q -ary code of length n and minimum Hamming distance d , the number of codewords T is bounded above by q^{n-d+1} . In other words, given an (n, T, d, q) code then $q^{n-d+1} \geq T$.

Theorem 4 *Another bound for an ϵ - AU (K, M, b) is: $K \geq \log(\epsilon^{-1}(M/b - 1))$*

Proof Using Theorem 3, construct an $(n = 2^K, T = 2^M, d = n - \epsilon 2^K, q = 2^b)$ code from the universal hash function ϵ - AU (K, M, b). This code must satisfy the Singleton bound, so we obtain:

$$\begin{aligned} q^{n-d+1} &\geq T \\ 2^{b(\epsilon 2^K + 1)} &\geq 2^M \\ r &\geq \log(\epsilon^{-1}(M/b - 1)) \quad \blacksquare \end{aligned}$$

When M is an integer multiple of b , this is equivalent to the combinatorial AU -bound in Theorem 1. In contrast, when M is not an integer multiple of b , the combinatorial AU -bound is tighter (or greater) than the one derived from coding theory in Theorem 4, because $\lfloor \frac{M}{b} \rfloor > M/b - 1$.

To verify the above argument, we compute and compare the lower bounds of the key length for a simple construction of an AU , where $M = 3, b = 2$ and $\epsilon = 1/2$, by using formulas of theorems 4 and 1.

- The AU -bound of Theorem 4 gives $2^K \geq \epsilon^{-1}(M/b - 1) = 1$. This underestimates the key length because it is impossible to construct such an AU with a single key, i.e. an $(\epsilon = 1/2)$ - AU ($K = 0, M = 3, b = 2$) does not exist.⁴ In other words, the equality of this bound cannot be achieved.
- The combinatorial AU -bound of Theorem 1 gives $2^K \geq \epsilon^{-1} \lfloor M/b \rfloor = 2$ corresponding to an $(\epsilon = 1/2)$ - AU ($K = 1, M = 3, b = 2$) or an $(n = 2, T = 8, d = 1, q = 4)$ code. This bound is tight because such an AU can be constructed as seen in Table 1.

The following theorem and its proof will give the reader a better insight of why the Singleton bound when being converted into universal hashing parameters does not produce a tight AU -bound.

⁴We consider the probability of hash collision ϵ as the key varies uniformly over its domain. Since there are $2^M = 8$ different messages mapping onto $2^b = 4$ different hash outputs under a single key, $\epsilon = 1 > 1/2$.

Theorem 5 Given an ϵ - AU (K, M, b) and M is not an integer multiple of b . When this AU is converted into its corresponding code by using Theorem 3, the resulting code whose parameters are (n, T, d, q) never satisfies the Singleton bound with equality.

Proof This theorem is proved by contradiction. Assume that the resulting code (n, T, d, q) can satisfy the Singleton bound with equality, by using Theorem 3 we have

$$\begin{aligned} T &= q^{(n-d+1)} \\ 2^M &= 2^{b(n-d+1)} \\ M &= b(n-d+1) \end{aligned}$$

Since $n-d+1$ is integer, this means that M is an integer multiple of b , which leads to a contradiction. ■

This theorem therefore implies the Singleton bound when being converted into universal hashing parameters does not give a tight bound for this subclass of an AU where M is not an integer multiple of b , since equality is never satisfied.

One might note that any AU -bound is also a bound for error correcting codes. However when we convert the combinatorial bound into the parameters in coding, the result is, perhaps surprisingly, no better than the Singleton bound as demonstrated below.

- When M is an integer multiple of b , the two bounds are equivalent thanks to the above analysis, i.e. $q^{n-d+1} \geq T$.
- When $M = tb + b'$, where t is an integer and $0 < b' < b$. The combinatorial AU -bound is equivalent to: $n - d = \epsilon 2^K \geq \lfloor M/b \rfloor = t$, and so $q^{n-d} 2^{b'} \geq T = 2^{tb+b'}$. Since the number of codewords T must be an integer and $1 < 2^{b'} < q$, we have $q^{n-d+1} - 1 \geq T$.

Since the Singleton bound determines the maximum size of a q -ary code, and moreover there is no apparent benefit in distinguishing whether $M = \log T$ is an integer multiple of $b = \log q$ in coding theory, our combinatorial AU -bound when being transformed into coding theory parameters does not improve the Singleton bound.

4 The significance of the threshold value of ϵ

We are going to compare the combinatorial AU -bound introduced in Theorem 1 with other bounds for not only AU but also AXU and ASU to understand the accuracy and significance of our result. This comparative analysis also uncovers the importance of the value $\epsilon = (1 + \frac{b}{M-b})2^{-b}$ which represents a *threshold* in the behaviour of bounds, and therefore quantifies the asymptotic *Wegman-Carter* behaviour of any universal hash functions.

In addition, we introduce a new AXU -bound derived from the ASU -bound of Kabatianskii et al. [18] and a connection between AXU and ASU due to Wegman and Carter [43].

4.1 Comparison between the combinatorial and other AU -bounds

Stinson's AU -bound, which can be derived from the Plotkin bound in coding theory [37], is as follows: $2^K \geq \frac{2^M(2^b-1)}{2^{M(\epsilon 2^b-1)+2^{2b}(1-\epsilon)}}$. When $\epsilon = 2^{-b}$, this is much tighter than the combinatorial AU -bound for then it gives $K \geq M - b$, which means that the key bitlength grows at least linearly with the message bitlength. In contrast, as we increase ϵ to 2^{1-b} then setting $K = b$ satisfies the bound, i.e. the key needs be no longer than the bitlength of the hash.

It is known that as ϵ moves away from 2^{-b} the Stinson's AU -bound converges to the hash bitlength b as M increases (see Appendix A), whereas the combinatorial bound grows in proportion to $\log M$. Hence there comes a point as M and ϵ increase where Stinson's bound becomes weaker than the combinatorial one. In order to locate that point, we find the value of ϵ above which ours is greater than Stinson's. To simplify the calculation, we will round up the combinatorial AU -bound to $2^K \geq \frac{M}{\epsilon b}$. This gives a very good approximation to the crucial value.

$$\begin{aligned} \frac{M}{\epsilon b} &> \frac{2^M(2^b-1)}{2^{M(\epsilon 2^b-1)+2^{2b}(1-\epsilon)}} \\ \epsilon &> \frac{M2^M - M2^{2b}}{M2^{M+b} - M2^{2b} - b2^{M+b} + b2^M} \end{aligned}$$

Since $2^{2b} \ll 2^M \ll 2^{M+b}$, the above can be approximated as follows:

$$\epsilon > \frac{M2^M}{M2^{M+b} - b2^{M+b}} = \frac{M}{(M-b)2^b} = \left(1 + \frac{b}{M-b}\right) 2^{-b}$$

We therefore refer to $\left(1 + \frac{b}{M-b}\right) 2^{-b}$ as the *threshold* value of ϵ . Since M is always assumed to be significantly bigger than b , Stinson's AU -bound can only be tight within a very short range of ϵ . Moreover, the difference between the threshold value and 2^{-b} , which is $\frac{b}{(M-b)2^b}$, can be made as small positively as we want. This implies that if ϵ exceeds 2^{-b} by an arbitrarily small positive value the message bitlength grows at most exponentially with the key bitlength as demonstrated in the combinatorial AU -bound, but if $\epsilon = 2^{-b}$ it will grow at most linearly as shown in Stinson's AU -bound.

While the same asymptotic behaviour has also been derived from a relation between ASU and codes correcting independent errors by Johansson et al. [17, 18], it is not clear to us how we can derive the same threshold value of ϵ from the strategy used by Johansson et al. As a consequence, our approach of deriving the result quantitatively demonstrates three further important points:

- If we fix the bitlengths of an input message and a hash output then Stinson's AU -bound is still useful when $2^{-b} < \epsilon < \left(1 + \frac{b}{M-b}\right) 2^{-b}$. See Table 2 for more information.
- Given any value of ϵ which exceeds 2^{-b} by an arbitrarily small positive value, we can determine the threshold of input messages' bitlength ($M \geq b + \frac{b}{2^b\epsilon-1}$) above which the message bitlength can apparently start to grow exponentially with the key bitlength, i.e. the combinatorial AU -bound gives a better estimate than Stinson's AU -bound.
- The threshold value of ϵ , perhaps surprisingly, has the same theoretical importance when we visit different ASU - and AXU -bounds in Appendix C. See Table 2 for more information.

4.2 Comparison between the combinatorial AU -bound and known ASU - and AXU -bounds

Since ASU is more restrictive than AU , intuitively we would expect that the number of bits required for the key in AU should be smaller than in ASU with respect to the same set of parameters (ϵ, M, b) . This analysis is reflected by the following comparisons:

- When $\epsilon = 2^{-b}$, Stinson's AU -bound [37] ($K \geq M - b$) is smaller than Stinson's ASU -bound [36, 37]⁵ ($K \geq M + b - 1$) by $2b - 1$ bits. But when $\epsilon > 2^{-b}$, the gap gets closer as follows:
- The combinatorial AU -bound in Theorem 1 is smaller than Kabatianskii's ASU -bound [18], $K \geq b + \log(\epsilon^{-1} \lfloor M/b \rfloor)$,⁶ by at least b bits.
- The difference between the combinatorial AU -bound and Gemmell-Naor's ASU -bound [12],⁷ $K \geq \log M + 2 \log \epsilon^{-1} - \log \log \epsilon^{-1}$, gets very near to b when $\theta \ll b$: $\log \epsilon^{-1} + \log \frac{b}{\log \epsilon^{-1}} = b - \theta + \log \frac{b}{b-\theta}$

Coincidentally, it is known that if there exists an ϵ - AXU (K, M, b) then it can be used to construct an ϵ - ASU $(K + b, M, b)$, thanks to the work of Wegman and Carter [43], i.e. see Theorem 6.

Theorem 6 [43, 11]. *Let $H = \{h_k() \mid k \in [0, 2^K]\}$ be an ϵ - AXU (K, M, b) ,⁸ then $\hat{H} = \{\hat{h}_{k,s}() \mid k \in [0, 2^K), s \in [0, 2^b), \text{ and } \hat{h}_{k,s}() = h_k() \oplus s\}$ is an ϵ - ASU $(K + b, M, b)$.*

Proof of this theorem can be found in Appendix B. Applying Theorem 6 to Kabatianskii's ASU -bound, $K \geq b + \log(\epsilon^{-1} \lfloor M/b \rfloor)$, we can derive its AXU -variant as in the following theorem.

Theorem 7 *For any ϵ - AXU (K, M, b) : $K \geq \log(\epsilon^{-1} \lfloor M/b \rfloor)$, provided⁹ $M/b < \sqrt{2^{K+1}(1 - 2^{-b})} - 1/2$*

This theorem shows that AU -bound is strictly shorter than AXU -bound for some set of parameters (ϵ, M, b) , i.e. when M is an integer multiple of b . This argument is consistent with the formal definitions, since AXU is a stronger definition of AU .

For example, when we set $\epsilon = 2^{-b}$, Stinson's AU -bound yields $M - b$ bits compared to M , derived from Stinson's AXU -bound ($2^K \geq \frac{2^M(2^b-1)}{2^{b\epsilon(2^M-1)+2^b-2M}}$) [37].¹⁰ We will see again that this comparative analysis is justified for larger values of ϵ when we visit constructions based on *polynomial hashing* over finite fields in Section 5.

⁵Stinson's ASU -bound can be derived from the second Johnson bound for constant weight binary codes [37].

⁶Kabatianskii's ASU -bound, which is derived from the Johnson bound in Theorem 15 of [18], is valid when $M/b < \sqrt{2^{K-b+1}(1 - 2^{-b})} - 1/2$.

⁷We note that the bound was reported in the paper of Gemmell and Naor [12] (Section 5.1). However, it was noted there that the bound was actually introduced by Noga Alon through private communication.

⁸We note that the AXU in this theorem does not need to be uniformly distributed as argued by Etzel et al. [11].

⁹As pointed out in footnote 6 and [18], there is a condition for the validity of Kabatianskii's ASU -bound, and therefore the same condition should apply to the AXU -variant of Kabatianskii's ASU -bound.

¹⁰Stinson's AXU -bound is derived from the second Johnson bound for constant weight binary codes.

5 The optimality of polynomial hashing as an AU

Polynomial hashing over finite fields was independently introduced by Boer [8], Johansson et al. [17], and Taylor [40]. To the best of our knowledge, polynomial hashing as an authentication code (ASU) has only been proved to be *asymptotically optimal* by Johansson et al. [17].¹¹

Extending this result, we will show a different version of polynomial hashing which is designed as an AU is *optimal*, because it meets the combinatorial AU -bound in Theorem 1 with equality.

Fix some positive integer t . Let the set of all messages be $\{m = \langle m_1, \dots, m_t \rangle; m_i \in \mathbb{F}_q\}$, here $b = \log q$ and the message bitlength is $M = tb = t \log q$.

In the first version of polynomial hashing as an AU , each message m will form a polynomial $m(x)$ of degree less than t over \mathbb{F}_q . For any key $k \in \mathbb{F}_q$, the universal hash of message m under key k is equivalent to $m(k)$ over \mathbb{F}_q .

$$h_k(m) = m(k) = m_1 + m_2k + m_3k^2 + \dots + m_tk^{t-1}$$

If we fix two different messages A and $B = A + m$, then a hash collision is equivalent to: $0 = h_k(A) + h_k(B) = A(k) + B(k) = m(k)$. Since the polynomial $m(k)$ is of degree up to $(t-1)$, we have $\epsilon = (t-1)q^{-1} = (M/b-1)2^{-r}$, and so $K = \log(\epsilon^{-1}(M/b-1))$. The equality in the combinatorial AU -bound implies optimality of polynomial hashing as an AU for any $M/b = t \in [2, q]$.

The construction above is not an AXU because if we set $\omega = A_1 + B_1$ and for all $i \in (1, t]$: $A_i = B_i = 0$, then for all $k \in \mathbb{F}_q$ we have $h_k(A) + h_k(B) = A_1 + B_1 = \omega$. In contrast, letting message m form a polynomial of degree up to t can get around this problem completely:

$$h_k(m) = m(k) = m_1k + m_2k^2 + \dots + m_tk^t$$

Similar calculations show that this is an $(\epsilon = t/q)$ - AXU , which meets the AXU -variant of Kabatianskii's ASU -bound in Theorem 7 with equality: $\log(\epsilon^{-1}\lfloor M/b \rfloor) = \log q = r$. This, therefore, justifies the difference between our combinatorial AU -bound and the AXU -variant of Kabatianskii's ASU -bound, i.e. when M is an integer multiple of b , AXU -bound is greater than AU -bound with respect to the same set of parameters (ϵ, M, b) .

Using Theorem 6 and the above construction, we can build an $(\epsilon = \frac{t}{2^b})$ - ASU ($K = 2b, M = tb, b$), which was originally introduced by Johansson et al. [17]. For any pair of keys $(k, s) \in \mathbb{F}_q^2$:

$$h_{k,s}(m) = s + m(k) = s + m_1k + m_2k^2 + \dots + m_tk^t$$

This meets Kabatianskii's ASU -bound ($K \geq b + \log(\epsilon^{-1}\lfloor M/b \rfloor)$) with equality.¹² However, Kabatianskii's ASU -bound and its AXU -variant have only been proved to be valid when $t < \sqrt{2^{b+1}(1-2^{-b})} - 1/2 = \sqrt{2q(1-1/q)} - 1/2$, and so the two polynomial hashings as AXU and ASU can only be proved to be optimal under the condition as was also pointed out by Kabatianskii et al. [18].

¹¹Since Kabatianskii's ASU -bound has only been proved to be valid in a partial range of parameters (see footnote 6 or [18]), the optimality of polynomial hashing as an ASU remains to be proved. On the other hand, polynomial hashing as an ASU is known to be asymptotically optimal due to Johansson et al. [17], i.e. the authors used polynomials to construct an $(\epsilon = \frac{t}{2^b})$ - ASU ($K = 2b, M = tb, b$), where t is an integer, and they proved that for t fixed and $b \rightarrow \infty$ then $2^M = 2^{tb}$ is *asymptotically* the maximum number of messages that can be securely authenticated.

¹²There is another ASU -bound due to Gemmell and Noar (see Table 2 or [12]), however polynomial hashing as an ASU does not satisfy the bound with equality when $t \in [b, 2^{b-1}]$. This implies that Kabatianskii's ASU -bound is tighter than Gemmell-Noar's ASU -bound over the range of parameters where Kabatianskii's ASU -bound is valid.

6 Conclusions and future research

Since the rapid developments of a new and non-standard authentication technology which uses almost universal hash function, we believe that there is a need for other AU -constructions (e.g. better than polynomial hashing in terms of speed) and its theoretical bounds. In this paper, our first contribution is the introduction of a new (pairwise and l -wise) AU -bound which, as opposed to Stinson's AU -bound, gives a tight result when the collision probability moves away from its theoretical minimum by an arbitrarily small value. In addition, the key length only grows in proportion to the logarithm of message length, and thus follows the asymptotic Wegman-Carter behaviour.

Secondly we have demonstrated that the use of the equivalence between universal hash functions and error-correcting codes does not always give tight bounds for universal hash functions as conventionally expected. This work would open the way for re-examining existing bounds for universal hash functions which have been derived from bounds of error-correcting codes [37, 38, 12, 18] or other combinatorial objects such as difference matrices [37], orthogonal or perpendicular arrays [21, 37, 38, 39], and balanced incomplete block designs [21, 31, 36, 37, 39].

We hope that the same approach of using combinatorial analysis would lead to further bounds for other families of l -wise universal hash functions, because as far as we are aware the conventional approach based on existing results of other combinatorial objects only seems to work well with pairwise universal hash functions.

Finally we have quantified the asymptotic Wegman-Carter behaviour of any universal hash functions by introducing an important value of the hash collision probability ϵ that represents a threshold in behaviours of bounds for AU , AXU , and ASU ; the behaviour is summarised in Table 2.

References

- [1] *Bibliography on Authentication Codes*. (Up to 1998) Maintained by D.R. Stinson and R. Wei. See: <http://www.cacr.math.uwaterloo.ca/dstinson/acbib.html>
- [2] *Simple Pairing White Paper*. See: www.bluetooth.com/NR/rdonlyres/0A0B3F36-D15F-4470-85A6-F2CCFA26F70F/0/SimplePairing_WP_V10r00.pdf
- [3] S. Bakhtiari, R. Safavi-Naini, and J. Pieprzyk. *A message authentication code based on latin squares*. Australasian Conference on Information Security and Privacy, ACISP 1997, LNCS vol. 1270, 194-203.
- [4] D.J. Bernstein. *Stronger security bounds for Wegman-Carter-Shoup authenticators*. Advances in Cryptology, EUROCRYPT 2005, LNCS vol. 3497, 164-180.
- [5] D.J. Bernstein. *The Poly1305-AES message-authentication code*. Fast software encryption, FSE 2005, LNCS vol. 3557, pp. 32-49.
- [6] J. Bierbrauer, T. Johansson, G.A. Kabatianskii, and B.J.M. Smeets. *On Families of Hash Functions via Geometric Codes and Concatenation*. Advances in Cryptology, CRYPTO 1993, LNCS vol. 773, 331-342.
- [7] J. Bierbrauer. *Introduction to Coding Theory*. (pages 240-241). Published by CRC Press, 2004. ISBN 1584884215, 9781584884217.

	$\epsilon < \left(1 + \frac{b}{M-b}\right) 2^{-b}$	$\epsilon > \left(1 + \frac{b}{M-b}\right) 2^{-b}$
ϵ - <i>AU</i>	Stinson's bound [36, 37] $\log \left(\frac{2^M(2^b-1)}{2^M(\epsilon 2^b-1)+2^{2b}(1-\epsilon)} \right)$	M is an integer multiple of b <i>New</i> , Theorems 1 and 4 $\log \frac{M-b}{\epsilon b}$ M is <i>not</i> an integer multiple of b <i>New</i> , Theorem 1 $\log(\epsilon^{-1} \lfloor M/b \rfloor)$
ϵ - <i>AXU</i>	Stinson's bound [37] $\log \left(\frac{2^M(2^b-1)}{2^b \epsilon (2^M-1) + 2^{b-2^M}} \right)$	<i>AXU</i> -variant of Kabatianskii's <i>ASU</i> -bound <i>New</i> , Theorem 7 $\log(\epsilon^{-1} \lfloor M/b \rfloor)$ (provided $M/b < \sqrt{2^{K+1}(1-2^{-b})} - 1/2$)
ϵ - <i>ASU</i>	Stinson's bound [36, 37] $\log \left(1 + \frac{2^M(2^b-1)^2}{2^b \epsilon (2^M-1) + 2^{b-2^M}} \right)$	Kabatianskii's bound [18] $b + \log(\epsilon^{-1} \lfloor M/b \rfloor)$ (provided $M/b < \sqrt{2^{K-b+1}(1-2^{-b})} - 1/2$) Gemmell and Noar's bound [12] $\log M + 2 \log \epsilon^{-1} - \log \log \epsilon^{-1}$

Table 2: Classification of different lower bounds on the key length K for *AU*, *AXU* and *ASU* in relation to the threshold value of ϵ : $\left(1 + \frac{b}{M-b}\right) 2^{-b}$.

- [8] B. den Boer. *A simple and key-economical unconditional authentication scheme*. Journal of Computer Security 2 (1993), 65-71.
- [9] J.L. Carter and M.N. Wegman. *Universal Classes of Hash Functions*. Journal of Computer and System Sciences, 18 (1979), 143-154.
- [10] S.J. Creese, M.H. Goldsmith, A.W. Roscoe, and I. Zakiuddin. *The attacker in ubiquitous computing environments: Formalising the threat model*. Workshop on Formal Aspects in Security and Trust, Pisa, Italy, 2003.
- [11] M. Etzel, S. Patel, and Z. Ramzan. *SQUARE HASH : Fast message authentication via optimized universal hash functions*. Advances in Cryptology, CRYPTO 99, LNCS vol. 1666, 234-251.
- [12] P. Gemmell and M. Naor. *Codes for Interactive Authentication*. Advances in Cryptology, CRYPTO 93, LNCS vol. 773, 355-367.
- [13] C. Gehrman, C. Mitchell and K. Nyberg. *Manual Authentication for Wireless Devices*. RSA Cryptobytes, vol. 7, no. 1, pp. 29-37, 2004.
- [14] International Organisation for Standardisation, Geneva, Switzerland. *ISO/IEC 9798 Information technology - Security techniques - Entity authentication - Part 6: Mechanisms using manual data transfer*, 2003.
- [15] H. Handschuh and B. Preneel. *Key-Recovery Attacks on Universal Hash Function Based MAC Algorithms*. Advances in Cryptology, CRYPTO 2008, LNCS vol. 5157, 144-161.
- [16] S.-H. Heng and K. Kurosawa. *Square hash with a small key size*. Australasian Conference on Information Security and Privacy, ACISP 2003, LNCS vol. 2727, 522-531.
- [17] T. Johansson, G.A. Kabatianskii, and B. Smeets. *On the relation between A-Codes and Codes correcting independent errors*. Advances in Cryptology, EUROCRYPT 1993, LNCS vol. 765, 1-11.
- [18] G.A. Kabatianskii, B. Smeets, and T. Johansson. *On the cardinality of systematic authentication codes via error-correcting codes*. IEEE Transactions on Information Theory, IT-42 (1996), 566-578.
- [19] H. Krawczyk. *LFSR-based Hashing and Authentication*. Advances in Cryptology, CRYPTO 1994, LNCS vol. 839, 129-139.
- [20] H. Krawczyk. *New Hash Functions For Message Authentication*. Advances in Cryptology, EUROCRYPT 1995, LNCS vol. 921, 301-310.
- [21] K. Kurosawa, K. Okada, H. Saido, and D.R. Stinson. *New combinatorial bounds for authentication codes and key predistribution schemes*. Designs, Codes and Cryptography, 15 (1998), 87-100.
- [22] K. Kurosawa and S. Obana. *Combinatorial Bounds on Authentication Codes with Arbitration*. Design, Codes Cryptography 22 (3): 265-281 (2001).

- [23] S. Laur and S. Pasini. *SAS-Based Group Authentication and Key Agreement Protocols*. Public Key Cryptography, PKC, 197-213 (2008).
- [24] Y.-H. Lin, A. Studer, H.-C. Hsiao, J.M. McCune, K.-H. Wang, M. Krohn, P.-L. Lin, A. Perrig, H.-M. Sun, B.-Y. Yang. *SPATE: Small-group PKI-less Authenticated Trust Establishment*. Proceedings of the 7th international conference on Mobile systems, applications, and services, 2009, pp. 1-14.
- [25] A.Y. Lindell. *Comparison-Based Key Exchange and the Security of the Numeric Comparison Mode in Bluetooth v2.1*. Topics in Cryptology CT-RSA, LNCS vol. 5473, pp. 66-83, 2009.
- [26] W. Nevelsteen and B. Preneel. *Software performance of universal hash functions*. Advances in cryptology, EUROCRYPT 1999, LNCS vol. 1592, pp. 24-41.
- [27] L.H. Nguyen and A.W. Roscoe. *Authenticating ad hoc networks by comparison of short digests*. Information and Computation 206 (2008), 250-271.
- [28] L.H. Nguyen and A.W. Roscoe. *Authentication protocols based on low-bandwidth unspoofable channels: a comparative survey*. Journal of Computer Security (to appear).
- [29] B. Preneel. A lecture note on “Authentication Codes”.
- [30] V.S. Pless and W. Huffman. *Handbook of Coding Theory* (Chapter 4, Sec 2.2), published by Elsevier (1998). ISBN 0444500871. Or see: http://en.wikipedia.org/wiki/Singleton_bound
- [31] R.S. Rees and D.R. Stinson. *Combinatorial characterizations of authentication codes II*. Designs, Codes and Cryptography 7 (1996), 239-259.
- [32] A.W. Roscoe, B. Chen and L.H. Nguyen. *Reverse authentication in financial transactions*. Proceedings of IWSSI/SPMU 2010.
- [33] D.V. Sarwate. *A note on universal classes of hash functions*. Information Processing Letter, 10 (1): 41-45 (1980).
- [34] V. Shoup. *On Fast and Provably Secure Message Authentication Based on Universal Hashing*. Advances in Cryptology, CRYPTO 1996, LNCS vol. 1109, 313-328.
- [35] F. Stajano and R. Anderson. *The resurrecting duckling: Security issues for ad-hoc wireless networks*. Security Protocols 1999, LNCS vol. 1976, 172-194.
- [36] D.R. Stinson. *Universal Hashing and Authentication Codes*. Advances in Cryptology, CRYPTO 1991, LNCS vol. 576, 74-85.
- [37] D.R. Stinson. *On the Connections Between Universal Hashing, Combinatorial Designs and Error-Correcting Codes*. Congressus Numerantium, vol. 114 (1996), 7-27.
- [38] D.R. Stinson. *The combinatorics of authentication and secrecy codes*. Journal of Cryptology 2 (1990), 23-49.
- [39] D.R. Stinson. *Combinatorial techniques for universal hashing*. Journal of Computer and System Sciences 48 (1994), 337-346.

- [40] R. Taylor. *An Integrity Check Value Algorithm for Stream Ciphers*. Advances in Cryptology, CRYPTO 1993. LNCS vol. 773, Springer-Verlag, pp. 40-48, 1994.
- [41] J. Valkonen, N. Asokan, and K. Nyberg. *Ad Hoc Security Associations for Groups*. European Workshop on Security and Privacy in Ad hoc and Sensor Networks, 2006. LNCS vol. 4357, 150-164.
- [42] S. Vaudenay. *Secure Communications over Insecure Channels Based on Short Authenticated Strings*. Advances in Cryptology, CRYPTO 2005, LNCS vol. 3621, 309-326.
- [43] 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 (1981), 265-279.

A Stinson's AU -bound analysis

To understand the dramatic collapse of Stinson's AU -bound as ϵ moves away from 2^{-b} , we write $\epsilon = \gamma 2^{-b} > 2^{-b}$, which is the same as $\gamma > 1$. The bound is as follows.

$$2^K \geq \frac{2^M(2^b - 1)}{2^M(\gamma - 1) + 2^{2b}(1 - \gamma 2^{-b})} = \frac{2^b - 1}{(\gamma - 1) + 2^{2b-M}(1 - \gamma 2^{-b})}$$

Note that since both terms in the denominator of the right-hand form are positive for $\gamma > 1$, with the second one converging to 0 as M increases, no matter how big M gets it can never prove a stronger lower bound on K than

$$K > \log \frac{2^b}{\gamma - 1} = b + \log \frac{1}{\gamma - 1}$$

B Proof of a connection between AXU and ASU : Theorem 6

Proof For any message m and hash output y , we have

$$P_I = \Pr_{k,s} [\hat{h}_{k,s}(m) = y] = \Pr_{k,s} [h_k(m) \oplus s = y]$$

For any value of k , s is uniquely determined by $s = h_k(m) \oplus y$, and thus $P_I = \frac{2^K}{2^{K+b}} = 2^{-b}$.

For every pair of distinct messages (m, \hat{m}) and for every pair of hash outputs (y, \hat{y}) , we have

$$P_S = \Pr_{k,s} [\hat{h}_{k,s}(m) = y, \hat{h}_{k,s}(\hat{m}) = \hat{y}] = \Pr_{k,s} [h_k(m) \oplus s = y, h_k(\hat{m}) \oplus s = y \oplus \hat{y}]$$

For any value of k , s is uniquely determined by $s = h_k(m) \oplus y$. Since $h_k(\cdot)$ is an ϵ - AXU (K, M, b) there are at most $\epsilon 2^K$ keys satisfying $h_k(m) \oplus h_k(\hat{m}) = y \oplus \hat{y}$, and thus $P_S \leq \frac{\epsilon 2^K}{2^{K+b}} = \epsilon 2^{-b}$. ■

C The threshold value in relation to AXU and ASU

We note that Stinson's bounds for AXU and ASU [37] have similar forms to his AU -bound [37]. Furthermore, the same similarity in form holds between Kabatianskii's ASU -bound [18], the AXU -variant of Kabatianskii's ASU -bound in Theorem 7, and the combinatorial AU -bound in Theorem 1. We therefore assert that the threshold value of ϵ has the same significance in the relationships between the two versions of ASU -bound, and of AXU -bound respectively.

The following calculation locates the value of ϵ above which Kabatianskii's ASU -bound becomes better than Stinson's ASU -bound.¹³

$$\begin{aligned} \frac{M2^b}{\epsilon b} &\geq \frac{2^M(2^b - 1)^2}{2^b\epsilon(2^M - 1) + 2^b - 2^M} \\ \epsilon &\geq \frac{M2^{b+M} - M2^{2b}}{M2^{2b+M} - M2^{2b} - b2^{2b+M} + b2^{b+M+1} - b2^M} \end{aligned}$$

Since $2^{2b} \ll 2^M \ll 2^{M+b}$ the above can be approximated as follows:

$$\epsilon > \frac{M2^{b+M}}{M2^{2b+M} - b2^{2b+M}} = \frac{M}{(M - b)2^b} = \left(1 + \frac{b}{M - b}\right) 2^{-b}$$

A similar calculation also leads us to conclude that Stinson's AXU -bound is overtaken by the AXU -variant of Kabatianskii's ASU -bound at the threshold value of ϵ .

A summary of the relation between all these different bounds for AU , AXU and ASU in relation to the threshold value of ϵ is given in Table 2.

¹³Since the constant 1 in Stinson's ASU -bound ($2^K \geq 1 + \frac{2^M(2^b-1)^2}{2^b\epsilon(2^M-1)+2^b-2^M}$) is very small compared to 2^K , we will ignore it in subsequent analysis to simplify the calculation. In addition, we will round up Kabatianskii's ASU -bound from $2^K \geq \frac{2^b}{\epsilon} \lfloor M/b \rfloor$ to $2^K \geq \frac{2^b M}{\epsilon b}$.