New combinatorial bounds for universal hash functions

L. H. Nguyen and A. W. Roscoe
Oxford University Computing Laboratory
Email: {Long.Nguyen,Bill.Roscoe}@comlab.ox.ac.uk

Abstract

We introduce a new lower bound on the key length in an *almost* universal hash function by using combinatorial analysis. At the same time, we use the well-studied relation between almost universal hashes and error-correcting codes introduced by Johansson et al. in 1993 to derive another similar bound which turns out to be not as tight as we had expected. To the best of our knowledge, this is the first time when combinatorial analysis yields a better universal hash bound than the use of the relation, and we will explain the reason why. We then compare the new bound against known bounds of this and other families of universal hashes and discover an important value of hash collision probability, which not only represents a *threshold* in the behaviour of bounds but also quantifies the *Wegman-Carter* effect.

1 Introduction and contribution

Universal hash function H with parameters (ϵ, r, K, b) was introduced by Carter and Wegman [9, 35]. Each family, which is indexed by a r-bit key k, consists of 2^r hash functions mapping a message representable by K bits into a b-bit hash output: $H\ (r, K, b) = \{h_k() : \{0,1\}^K \to \{0,1\}^b | k \in [0, 2^r)\}$.

In this paper we use combinatorial analysis to introduce a new bound, termed the *combinatorial* bound, for an ϵ-*almost* universal hash function (ϵ-*AU*). This result tells us the lower bound on the bitlength of the hash key with respect to a fixed amount of information we want to hash, the hash output bitlength and the hash collision probability ϵ.

Although there has been much work in this area, most researchers concentrate on bounds for an ϵ-*almost strongly* universal hash function (ϵ-*ASU*, a more restrictive version of ϵ-*AU* as can be seen in their definitions below). This is because a much-used mechanism in practice, called MAC, make use of an ϵ-*ASU* [29, 30, 17, 18, 12, 16, 35, 9, 2]. We however believe that there is a similar potential for *AU*. For example, a new class of authentication schemes, based on new concepts of trust derived from human actions and interactions, has been recently proposed to replace PKI and passwords in pervasive computing environments [28, 34, 10, 23, 24, 25, 33, 21]. Some of these protocols make use of a new cryptographic *digest* function introduced in [23, 24], with similar security properties and purposes to an *AU*. In these protocols, digest or hash keys are always random and fresh in each protocol session, and so a substitution attack, which relies of the reuse of a hash key for multiple messages, is irrelevant. Hence, what we require is a protection against hash collision attacks (*AU*) as opposed to substitution attacks (*ASU*).

Moreover, since universal hash keys in MAC are often large, one reuses a single secret key for multiple messages as mentioned above. 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 [13]. Avoiding reusing keys would render most key recovery attacks useless, and so it is desirable to construct universal hashes with short keys, which in turn generate the need to calculate the lower bound of universal hash key length.

We are aware of a relation between *almost* universal hash functions and error-correcting codes discovered by Johansson et al. [15], which implies that every bound of coding theory potentially corresponds to another bound on universal hashes, and vice versa. Consequently, we will show how to use the *Singleton* bound to derive a different AU-bound which turns out to be not as tight as we had expected. To the best of our knowledge, this is the first time when combinatorial analysis yields a better universal hash bound than the use of the relation. However, when we convert the combinatorial AU-bound into parameters in coding theory it is perhaps surprising to discover that the result is no better than Singleton bound.

In comparing the combinatorial AU-bound to Stinson's AU-bound [29, 30], we discover the significance of the value $(1+\frac{b}{K-b})2^{-b}$: as ϵ increases beyond the threshold, our bound is tighter than Stinson's AU-bound. Subsequently this threshold value will be shown to have the same theoretical significance in relationships between known bounds for *almost XOR* and *almost strongly* universal hash functions. What this illustrates is a behaviour of any universal hash functions, known as the "Wegman-Carter effect" in the literature [7, 20], previously reported in [15, 16] by Johansson, Kabatianskii and Smeets: if ϵ exceeds 2^{-b} (the theoretical minimum[1]) by an arbitrarily small positive value, then the total number of messages, that can be authenticated, grows exponentially with the number of keys provided, but if $\epsilon = 2^{-b}$ it only grows linearly. However, while these authors only demonstrate this behaviour asymptotically, we are able to *quantify* it using the threshold value.

We end this paper by proving the *optimality* of polynomial hashing over finite field [8, 15, 32] in building AU, AXU and ASU, i.e. they meet the combinatorial AU-bound, AXU- and ASU-bounds with equality. This therefore improves on the proof of *asymptotic optimality* of polynomial hashing as an ASU given by Johansson et al. [15].

In our work, we also introduce a new bound for an ϵ-AXU. The bound is derived from Kabatianskii's ASU-bound [16] and a connection between ASU and AXU [35, 11]. For this reason, we suspect that this has been known to the community. However, the bound has never been published, and moreover rigorously analysed in relation to other known bounds. We will show that the bound is 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 bitlengths[2] of hash keys, input messages and hash output instead of the cardinalities of the sets of these parameters (2^r, 2^K and 2^b) as in other papers. 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 universal hash functions. Here ϵ, which is sometimes written as $2^{\theta-b} = \gamma 2^{-b}$, is referred to as the collision, differential or interpolation probability associated with ϵ-AU, ϵ-AXU or ϵ-ASU, respectively.[3] In all following definitions, we

[1] In practice, the minimum collision probability of an AU is $\frac{2^K-2^b}{2^{K+b}-2^b}$, which is less than 2^{-b}. This occurs in an *optimally universal* hash scheme introduced by Sarwate [26].

[2] In practice, it is often the case that r, K and b are integers.

[3] The 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.

look at the probability of some condition being met, e.g. hash collision, as the key k varies uniformly over its domain: $\mathbf{Pr}_k[]$.

An ϵ-*almost* universal hash function, ϵ-*AU* (r, K, b) [9, 29]

H is an ϵ-*AU* iff for all different messages m and m':
$\mathbf{Pr}_k[h_k(m) = h_k(m')] \leq \epsilon$

An ϵ-*almost XOR* universal hash function, ϵ-*AXU* (r, K, b) [17, 18, 29]

H is an ϵ-*AXU* iff for every pair of distinct messages (m, m') and
any $\omega \in \{0, 1\}^b$: $\mathbf{Pr}_k[h_k(m) \oplus h_k(m') = \omega] \leq \epsilon$

An ϵ-*almost strongly* universal hash function, ϵ-*ASU* (r, K, b) [35, 29]

(a) For every message m and hash output y: $\mathbf{Pr}_k[h_k(m) = y] \leq 2^{-b}$.
(b) For every pair of distinct messages (m, m') and for every pair
of hash outputs (y, y'): $\mathbf{Pr}_k[h_k(m) = y, h_k(m') = y'] \leq \epsilon 2^{-b}$

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 ϵ-AU_l, and therefore we give the definition below. We argue that not only is this of theoretical interest to study ϵ-AU_l, but also useful in many applications, such as in the new family of authentication protocols discussed in the introduction, where the intruder attempts to fool 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 w.r.t more than two different input messages. However, unless indicated, our work presented in this paper always refers to pairwise universal hash functions.

A l-wise ϵ-*almost* universal hash function, ϵ-AU_l (r, K, b)

H is an ϵ-AU_l iff for any l different messages $\{m_1, \ldots, m_l\}$:
$\mathbf{Pr}_k[h_k(m_1) = \cdots = h_k(m_l)] \leq \epsilon$

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

3 Bounds for almost universal hash functions

In this section, we first derive a new bound of an *AU* using combinatorial analysis.

We then use *Singleton* bound [1] to derive another *AU*-bound. Although several bounds in coding theory have been converted into equivalent bounds for universal hashes, e.g. Plotkin bound [30] or Johnson bound [16], to the best of our knowledge, Singleton bound has not been used.

It is perhaps interesting to discover the combinatorial *AU*-bound is tighter (greater) than the one derived from Singleton bound when K is *not* a multiple of b, and both K and b are integers. When K is a multiple of b they are equivalent.

3.1 Combinatorial *AU*-bound

Theorem 1 *If there exists an ϵ-AU (r, K, b) then $r \geq \log\left(\epsilon^{-1} \lfloor (K-1)/b \rfloor\right)$*

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

Proof For any key k_1, there exists a hash value h_1 such that there are at least 2^{K-b} different 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 2^{K-2b} of these messages that all map to some hash value h_2, which can be equal to h_1, under k_2. And if we continue this process repeatedly, in the end, this will result in at least two distinct messages mapping to the same values under $c = \lfloor (K-1)/b \rfloor$ different keys[4] out of 2^r all possible key-values.

We now can deduce that if a family of hash functions is ϵ-*almost universal* then $\lfloor (K-1)/b \rfloor$ must be smaller than or equal to ϵ portion of the key space: $\epsilon 2^r \geq \lfloor (K-1)/b \rfloor$, which means that $r \geq \log(\epsilon^{-1} \lfloor (K-1)/b \rfloor)$ ▮

The distinction between this formula and what one gets by removing the -1 will become important in distinguishing AU from AXU and ASU in the sections to come. This result can be interpreted alternatively as follows: given the security parameter ϵ, the bitlengths of the key and the hash output, it yields an upper bound on the length of the information we are hashing: $K < b + 1 + \epsilon b 2^r$.

The proof of the combinatorial bound for a pairwise ϵ-AU_2 can be adapted to derive the corresponding bound for a l-wise ϵ-AU_l. Instead of leaving 2 different messages after c iterations as shown in the proof of Theorem 1, we need to leave l messages, and hence number of iterations c is upgraded to $\lfloor (K - \log l)/b \rfloor$. This leads to the following theorem.

Theorem 2 *If there exists a l-wise ϵ-AU_l (r, K, b) then* $r \geq \log\left(\epsilon^{-1} \lfloor (K - \log l)/b \rfloor\right)$

This is slightly lower than the combinatorial bound, since the likelihood of l different messages hashing to the same value is smaller than pairwise. Although there has been some study of l-wise *almost strongly* universal hash functions by Stinson [31] and Kurosawa et al. [19], 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 limit on message length K relative to b and r in both our pairwise and l-wise combinatorial AU-bounds, which makes them more attractive than a similar ASU-bound proposed by Kabatianskii et al. [16], 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 hashes and error-correcting codes (i.e. see Theorem 3), which was first observed by Johansson et al. [15], has been used by many reseachers to derive tight bounds on universal hashes [29, 30, 16], the following comparative analysis will demonstrate that the strategy does not always give the best answer.

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

Theorem 3 [15, 6, 30]. *If there exists an ϵ-AU (r, K, b), then there exists an $(n = 2^r, T = 2^K, d = 2^r - 2^r\epsilon, q = 2^b)$ code. Conversely, if there exists an (n, T, d, q) code, then there exists an $(\epsilon = 1 - d/n)$-AU $(r = \log n, K = \log T, b = \log q)$.*

[4] The reason why we use $K-1$ instead of K is because we want to have at least $2^{K - b((K-1)/b)} = 2^1 = 2$ different messages left after c such iterations.

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

Singleton bound [1]: given an (n, T, d, q) code then $q^{n-d+1} \geq T$.

Theorem 4 *Another bound on an ϵ-AU (r, K, b), which is derived from Singleton bound, is:* $r \geq \log\left(\epsilon^{-1}\left(K/b - 1\right)\right)$

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

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

When K is a multiple of b, this is equivalent to the combinatorial AU-bound in Theorem 1.

In contrast, when K is not a multiple of b, and both K and b are integers, then the combinatorial AU-bound is tighter (or greater) than the one derived in Theorem 4, since $\lfloor (K-1)/b \rfloor > K/b - 1$.

We also discover that, a set of parameters (ϵ, r, K, b), where K is not a multiple of b, which achieves equality in the bound derived in Theorem 4 cannot be converted into an (n, T, d, q) code whose values of both n and d are integers.[5] Hence, it is impossible to construct an AU with the set of parameters, i.e. the bound derived from Singleton bound in Theorem 4 is not tight as shown in the following example and Table 1.

Let $K = 3$, $b = 2$ and $\epsilon = 1/2$, the AU-bound defined in Theorem 4 gives $r \geq \log\left(\epsilon^{-1}\left(K/b-1\right)\right) = 0$, which is not tight because it is impossible to construct such an AU with a single key, i.e. zero bit. The combinatorial AU-bound, on the other hand, gives $r \geq \log\left(\epsilon^{-1}\lfloor (K-1)/b \rfloor\right) = 1$ corresponding to an $(\epsilon = 1/2)$-AU $(r = 1, K = 3, b = 2)$ or an $(n = 2, T = 8, d = 1, q = 4)$ code.

Since any AU-bound is also a bound on error correcting codes, one might question: does the combinatorial AU-bound give rise to a new bound in coding theory which is tighter than Singleton bound? It is however perhaphs suprising to discover when we convert the combinatorial bound into parameters in coding, it becomes Singleton bound as demonstrated below.

- When $K = tb + b'$ and $b' \in [1, b]$, where t is an integer. The combinatorial AU-bound is equivalent to: $t = \lfloor (K-1)/b \rfloor \leq \epsilon 2^r = n - d$, and so $T = 2^{tb+b'} \leq q^{n-d+1}$.
- When $K = tb + b'$ and $0 < b' < 1$: $\lfloor (K-1)/b \rfloor = t - 1$. The combinatorial AU-bound is equivalent to $T \leq 2^{b'} q^{n-d+1}$, which is not as tight as Singleton bound.

4 The significance of the threshold value of ϵ

Having discovered the combinatorial AU-bound, we are going to compare it with other bounds for not only ϵ-AU but also ϵ-AXU and ϵ-ASU to understand the significance and contribution of our result. This comparative analysis will be given in the following order: Stinson's AU-bound in

[5] Assume $K = tb + b'$ where $0 < b' < b$ and t is an integer. If equality in the bound derived in Theorem 4 is achieved, we have: $n - d = 2^r\epsilon = t - 1 + b'/b$. Since b'/b is not an integer, both n and d cannot be integer at the same time.

	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_1	2	3	4	1	3	4	1	2

Table 1: A construction of an ($\epsilon = 1/2$)-*AU* $(1,3,2)$, in which there are $2^r = 2$ hash keys $\{k_1, k_2\}$ and $2^K = 8$ input messages $\{m_1, \ldots, m_8\}$. The range of a hash output is [1,$2^b = 4$].

Section 4.1, and in Section 4.2 we study two *ASU*-bounds of Gemmell and Naor, and Kabatianskii et al., and *AXU*-bounds.

This study leads us to discover the significance of the value $\epsilon = (1 + \frac{b}{K-b})2^{-b}$ which represents an important *threshold* in the behaviour of bounds, quantifying the *Wegman-Carter* effect. We also introduce a new *AXU*-bound derived from the *ASU*-bound of Kabatianskii et al. [16] and a connection between *AXU* and *ASU* that was introduced by Wegman and Carter [35].

We end this section with Table 2 that captures the interesting relationships w.r.t the threshold value between the combinatorial *AU*-bound, Kabatianskii's *ASU*-bound, the *AXU*-variant of Kabatianskii's bound, and Stinson's bounds for *AU*, *AXU* and *ASU*.

4.1 Comparison between the combinatorial and other *AU*-bounds

Stinson's *AU*-bound [29] is as follows: $2^r \geq \frac{2^K(2^b-1)}{2^K(\epsilon 2^b - 1) + 2^{2b}(1-\epsilon)}$. When $\epsilon = 2^{-b}$, this is much tighter than ours for then it gives $r \geq K - b$, 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 $r = b$ satisfies the bound, i.e. the key needs be no longer than the bitlength of the hash.

To explain the reason for the dramatic collapse, we present a different way to interpret the formula when $\epsilon = \gamma 2^{-b} > 2^{-b}$, which is the same as $\gamma > 1$.

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

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 K increases, no matter how big K gets it can never prove a stronger lower abound on r than

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

In other words, while the combinatorial bound grows in proportion to *log* K, this bound is essentially constant as K increases. Hence there comes a point as K 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 our bound from ($2^r \geq \epsilon^{-1}\lfloor (K-1)/b \rfloor$) to ($2^r \geq \frac{K}{\epsilon b}$). This gives a very good approximation to the crucial value.

$$\begin{array}{rcl} \frac{K}{\epsilon b} & > & \frac{2^K\left(2^b - 1\right)}{2^K\left(\epsilon 2^b - 1\right) + 2^{2b}(1-\epsilon)} \\ \epsilon & > & \frac{K2^K - K2^{2b}}{K2^{K+b} - K2^{2b} - b2^{K+b} + b2^K} \end{array}$$

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

$$\epsilon > \frac{K2^K}{K2^{K+b} - b2^{K+b}} = \frac{K}{(K-b)2^b} = \left(1 + \frac{b}{K-b}\right) 2^{-b}$$

From now on, we will refer to this value of ϵ as the *threshold* value. The result demonstrates that Stinson's AU-bound can only be tight within a very short range of ϵ, since K is always assumed to be significantly bigger than b. Moreover, the difference between the threshold value and 2^{-b}, i.e. $\frac{b}{(K-b)2^b}$, can be made as small as we want. This leads us to conclude that if ϵ exceeds 2^{-b} by an arbitrarily small positive value, then 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.

This conclusion has also been derived from a relation between *almost strongly* universal hash functions and codes correcting independent errors in the work of Johansson et al. [15, 16]. However, it is not clear to us how we can derive the same threshold value of ϵ from the asymptotic behaviour. As a consequence, our approach of deriving the result quantitatively demonstrates three further important points:

1. If we fix the bitlengths of an input message and a hash output, then Stinson's AU-bound is still useful when $2^{-b} < \epsilon < \left(1 + \frac{b}{K-b}\right) 2^{-b}$, more information can be found in Table 2.
2. Given any value of ϵ which exceeds 2^{-b} by an arbitrarily small positive value, we can determine the threshold of input messages' bitlength ($K \geq b + \frac{b}{2^b\epsilon - 1}$) 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.
3. The threshold value of ϵ, perhaps surprisingly, has the same theoretical importance when we visit different ASU- and AXU-bounds in Appendix A. See Table 2 for more information.

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

Having compared our result to AU-bounds, we turn our attention to ASU- and AXU-bounds. There are a number of existing ASU-bounds, introduced by Gemmell and Naor [12], and Kabatianskii et al. [16], that have similar form to our AU-bound. 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 w.r.t the same set of parameters (ϵ, K, b). This analysis is reflected by the following two comparisons:

- Our AU-bound, $r \geq \log\left(\epsilon^{-1}\lfloor (K-1)/b \rfloor\right)$, is smaller than Kabatianskii's ASU-bound, $r \geq b + \log(\epsilon^{-1}\lfloor K/b \rfloor)$, by at least b bits.[6]

[6] Kabatianskii's ASU-bound, Theorem 15 of [16], is valid when $K < b\sqrt{2^{r-b+1}(1-2^{-b})} - b/2$, which is equivalent to: $r > b + 2\log(K/b + 1/2) + \log\frac{2^b}{2(2^b-1)}$. In order for the bound to be met with equality, the bound must be itself greater than $b + 2\log(K/b + 1/2) + \log\frac{2^b}{2(2^b-1)}$. This is satisfied when $K < \frac{2b}{\epsilon} - b$, yielding a very large K in practice when the hash length is in the range from 80 to 160 bits.

- The difference between our *AU*-bound and Gemmell-Naor's *ASU*-bound,[7] $r \geq \log K + 2\log\epsilon^{-1} - \log\log\epsilon^{-1}$, gets very near to b when $\theta \ll b$: $\log\epsilon^{-1} + \log\frac{b}{\log\epsilon^{-1}} = b - \theta + \log\frac{b}{b-\theta}$

The above comparisons imply that the difference between *AU*- and *ASU*-bounds on the key length may be *very near*, or equal, to b bits w.r.t the same set of parameters (ϵ, K, b). Coincidentally, it is known that if there exists an ϵ-*AXU* (r, K, b) which is uniformly distributed,[8] then it can be used to construct an ϵ-*ASU* $(r + b, K, b)$, thanks to the work of Wegman and Carter [35]. This can be summarised by the following theorem, adapted from Lemma 1 of [11] by Etzel, Patel and Ramzan.

Theorem 5 [35, 11]. *Let $H = \{h_k() : \{0,1\}^K \longrightarrow \{0,1\}^b | k \in [0, 2^r)\}$ be an ϵ-almost XOR universal hash function. Moreover, suppose H is also uniformly distributed. Then $H' = \{h'_{k,b'}() : \{0,1\}^K \longrightarrow \{0,1\}^b | k \in [0, 2^r), b' \in [0, 2^b)\}$, defined by $h'_{k,b'}(m) = h_k(m) \oplus b'$ where b' is a b-bit random number, is an ϵ-almost strongly universal hash function.*

This means that if we apply Theorem 5 to Kabatianskii's *ASU*-bound, $r \geq b + \log(\epsilon^{-1}\lfloor K/b \rfloor)$, the corresponding *AXU*-bound will be $r \geq \log(\epsilon^{-1}\lfloor K/b \rfloor)$. We therefore term this the *AXU*-variant of Kabatianskii's bound, illustrated by the following theorem.

Theorem 6 *If there exists an ϵ-AXU (r, K, b) then $r \geq \log(\epsilon^{-1}\lfloor K/b \rfloor)$*

As pointed out in footnote 6 and [16], 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 bound:[9] $K < b\sqrt{2^{r+1}(1 - 2^{-b})} - b/2$.

The theorem also leads us to believe that ϵ-*AU*-bound may be shorter than ϵ-*AXU*-bound for some set of parameters (ϵ, K, b), i.e. when K is a multiple of b. This argument is consistent with the formal definitions, since ϵ-*AXU* is a stronger definition of ϵ-*AU*.

An example, showing the correctness of the argument, is given when we set $\epsilon = 2^{-b}$, Stinson's *AU*-bound yields $K - b$ bits compared to K, derived from Stinson's *AXU*-bound ($2^r \geq \frac{2^K(2^b-1)}{2^b\epsilon(2^K-1)+2^b-2^K}$) [30]. 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.

We note that Stinson's bounds for *AXU* and *ASU* have similar forms to his *AU*-bound. Furthermore, the same similarity in form holds between Kabatianskii's *ASU*-bound, the *AXU*-variant of Kabatianskii's bound and our *AU*-bound. Owing to this symmetry, we 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, as can be demonstrated in Appendix A.

5 The optimality of *polynomial hashing* as *AU*, *AXU* and *ASU*

Polynomial hashing over finite fields was independently introduced by Boer [8], Johansson et al. [15], and Taylor [32]. Subsequently, many research authors such as Shoup [27], Nevelsteen and Preneel [22], and Bernstein [5], report on several efficient implementations of polynomial hashing.

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

[8] A universal class H (r, K, b) of hash functions is uniformly distributed iff for every pair of a message and a hash value (m, y), as the key k varies uniformly over its range: $\mathbf{Pr}_k[h_k(m) = y] \leq 2^{-b}$.

[9] The exponent inside the square root operator is $r + 1$ instead of $r - b + 1$ as in the original formula because the key bitlength of an ϵ-*AXU* in this case is exactly b bits shorter than in an ϵ-*ASU*.

	$\epsilon < \left(1+\frac{b}{K-b}\right)2^{-b}$	$\epsilon > \left(1+\frac{b}{K-b}\right)2^{-b}$
ϵ-*AU*	Stinson's bound [29, 30] $\log\left(\frac{2^K(2^b-1)}{2^K(\epsilon 2^b-1)+2^{2b}(1-\epsilon)}\right)$	Combinatorial bound *New*, Theorem 1, Section 3.1 $\log\left(\epsilon^{-1}\lfloor (K-1)/b\rfloor\right)$ *New*, Theorems 4, Section 3.2 (from Singleton bound in coding theory) $\log\frac{K-b}{\epsilon b}$
ϵ-*AXU*	Stinson's bound [30] $\log\left(\frac{2^K(2^b-1)}{2^b\epsilon(2^K-1)+2^b-2^K}\right)$	*AXU*-variant of Kabatianskii's bound *New*, Theorem 6, Section 4.1 $\log\left(\epsilon^{-1}\lfloor K/b\rfloor\right)$
ϵ-*ASU*	Stinson's bound [29, 30] $\log\left(1+\frac{2^K(2^b-1)^2}{2^b\epsilon(2^K-1)+2^b-2^K}\right)$	Kabatianskii's bound [16] $b+\log\left(\epsilon^{-1}\lfloor K/b\rfloor\right)$ Gemmell and Noar's bound [12] $\log K + 2\log\epsilon^{-1} - \log\log\epsilon^{-1}$

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

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. [15]. In their paper, the authors used polynomials to construct an $(\epsilon = \frac{t}{2^b})$-$ASU(r = 2b, K = tb, b)$, where t is an integer, and they proved that for t fixed and $b \to \infty$ then $2^K = 2^{tb}$ is *asymptotically* the upper bound on the number of messages that can be securely authenticated.

Improving on the result, we will show that three familiar and slightly different versions of polynomial hashing (AU, AXU, and ASU) are *optimal*, because they meet the combinatorial AU-bound, the AXU-variant of Kabatianskii's bound, and respectively Kabatianskii's ASU-bound with equality. The last of these three constructions is also the one introduced by Johansson et al. [15].

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

In the first version of polynomial hashing, 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 hash of the message m with respect to the key k is equivalent to $m(k)$ over $\mathbb{F}_q$. This implies that bitlengths of the key and the hash output are equal to each other, i.e. $\log q = b = r$.

$$h_k(m) = m(k) = m_1 + m_2 k + m_3 k^2 + \cdots + m_t k^{t-1}$$

If we fix two different messages A and $B = A + m$,[10] 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 $(t-1)$, there are at most $t-1$ different roots out of total q possible values of key k causing a hash collision. This therefore implies that $\epsilon = (t-1)q^{-1} = \lfloor (K-1)/b \rfloor 2^{-r}$. Here the equality between $t-1$ and $\lfloor (K-1)/b \rfloor$ holds because K is a multiple of b. From this, we derive that $r = \log\left(\epsilon^{-1} \lfloor (K-1)/b \rfloor\right)$.

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 even though A and B are different messages, we always have:

$$\mathbf{Pr}_k[h_k(A) + h_k(B) = \omega] = \mathbf{Pr}_k[A_1 + B_1 = \omega] = 1$$

On the other hand, if we let the message m form a polynomial $m(x)$ of degree up to t over $\mathbb{F}_q$, then we can get around this problem completely. In the second version of polynomial hashing, we have

$$h_k(m) = m(k) = m_1 k + m_2 k^2 + \cdots + m_t k^t$$

Since the degree of this polynomial is up to t, a similar calculation leads us to conclude that this forms an $(\epsilon = t/q)$-AXU. And if we substitute this value of ϵ into the AXU-variant of Kabatianskii's bound, we obtain equality: $\log(\epsilon^{-1} \lfloor K/b \rfloor) = \log q = r$.

As pointed out in Section 4.2 and [16], the AXU-variant of Kabatianskii's bound has been only proved to be valid when $K = bt < b\sqrt{2^{r+1}(1-2^{-b})} - b/2$, which can be approximated to $t < 2^{(b+1)/2} - 1/2$ when $r = b$ in polynomial hashing. We note, however, that constructions based on polynomial hashing can meet this bound for all integer values of t over the wider range $[1, q = 2^b)$.

Since we can construct an $(\epsilon = \frac{t}{2^b})$-$AXU$ $(r = b, K = tb, b)$ that meets the AXU-variant of Kabatianskii's bound with equality, using Theorem 5 we can construct an $(\epsilon = \frac{t}{2^b})$-$ASU$ $(r = 2b, K = tb, b)$, which was originally introduced by Johansson et al. [15]. For any pair of keys $(k, s) \in \mathbb{F}_q^2$:

$$h_{k,s}(m) = s + m(k) = s + m_1 k + m_2 k^2 + \cdots + m_t k^t$$

This meets Kabatianskii's ASU-bound $(r \geq b + \log\left(\epsilon^{-1} \lfloor K/b \rfloor\right))$ with equality, and therefore is optimal.

[10] Addition and subtraction are the same thing in the finite field $\mathbb{F}_q$.

6 Conclusions and future research

In this paper, we have derived a new AU-bound combinatorically, which is tighter than another AU-bound derived from Singleton bound. To the best of our knowledge, this is the first time when one demonstrates that: the use of the connection between universal hash functions and error-correcting codes does not always give a tight bound on universal hashes.

This work hopefully will open the ways for re-examming many existing bounds on universal hashes which have been derived from bounds of error-correcting codes (ECC-bounds) or other combinatorial objects such as difference matrices, orthogonal arrays, and balanced incomplete block design [30]. As we have shown, there are subclasses of some universal hashes which cannot be transformed into equivalent codes that achieve equality in the ECC-bounds from which AU or ASU-bounds are derived. As a consequence, the AU or ASU-bounds are not tight since equality is not achievable in these subclasses of universal hashes. In the cases, combinatorial analysis might produce better bounds.

In addition, we quantify the (asymptotic) Wegman-Carter effect with respect to the threshold value of ϵ that represents a threshold in behaviours of bounds of AU, AXU, and ASU.

We have illustrated, in the l-wise variant of the combinatorial bound, how the inclusion of further parameters can capture wider range of security properties. It is therefore of interest to construct universal hash functions that meet our combinatorial bound on ϵ-AU_l for $l > 2$ with equality.

References

[1] See: `http://en.wikipedia.org/wiki/Singleton_bound`

[2] *Bibliography on Authentication Codes.* Maintained by D. Stinson and R. Wei.
See: `http://www.cacr.math.uwaterloo.ca/ dstinson/acbib.html`

[3] S. Bakhtiari, R. Safavi-Naini and J. Pieprzyk. *A message authentication code based on latin squares.* In 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.

[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] 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.

[14] S.-H. Heng and K. Kurosawa. *Square hash with a small key size.* Eighth Australasian Conference on Information Security and Privacy, ACISP 2003, LNCS vol. 2727, 522-531.

[15] T. Johansson, G.A. Kabatianskii, and B. Smeets. *On the relation between A-Codes and Codes correcting independent errors.* EUROCRYPT 1993, LNCS vol. 765, 1-11.

[16] 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.

[17] H. Krawczyk. *LFSR-based Hashing and Authentication.* CRYPTO 1994, LNCS vol. 839, 129-139.

[18] H. Krawczyk. *New Hash Functions For Message Authentication.* EUROCRYPT 1995, LNCS vol. 921, 301-310.

[19] 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.

[20] K. Kurosawa, S. Obana. *Combinatorial Bounds on Authentication Codes with Arbitration.* Des. Codes Cryptography 22 (3): 265-281 (2001).

[21] S. Laur and S. Pasini. *SAS-Based Group Authentication and Key Agreement Protocols.* In Public Key Cryptography - PKC, 197-213 (2008).

[22] W. Nevelsteen and B. Preneel. *Software performance of universal hash functions.* Advances in cryptology: EUROCRYPT 1999, LNCS vol. 1592, pp. 24-41.

[23] L.H. Nguyen and A.W. Roscoe. *Efficient group authentication protocol based on human interaction.* Proceedings of Workshop on Foundation of Computer Security and Automated Reasoning Protocol Security Analysis, 9-31 (2006).

[24] L.H. Nguyen and A.W. Roscoe. *Authenticating ad hoc networks by comparison of short digests.* Information and Computation 206 (2008), 250-271.

[25] L.H. Nguyen and A.W. Roscoe. *Separating two roles of hashing in one-way message authentication.* Proceedings of FCS-ARSPA-WITS 2008, 195-210.

[26] D.V. Sarwate. *A note on universal classes of hash functions.* Inf. Process. Lett., 10 (1): 41-45 (1980).

[27] V. Shoup. *On Fast and Provably Secure Message Authentication Based on Universal Hashing.* Advances in Cryptology - CRYPTO 1996, LNCS vol. 1109, 313-328.

[28] F. Stajano and R. Anderson. *The resurrecting duckling: Security issues for ad-hoc wireless networks.* Security Protocols 1999, LNCS vol. 1976, 172-194.

[29] D.R. Stinson. *Universal Hashing and Authentication Codes.* Advances in Cryptology - CRYPTO 1991, LNCS vol. 576, 74-85.

[30] D.R. Stinson. *On the Connections Between Universal Hashing, Combinatorial Designs and Error-Correcting Codes.* Congressus Numerantium, vol. 114 (1996), 7-27.

[31] D.R. Stinson. *The combinatorics of authentication and secrecy codes.* Journal of Cryptology 2 (1990), 23-49.

[32] R. Taylor. *An Integrity Check Value Algorithm for Stream Ciphers.* Advances in Cryptology - Crypto 1993. LNCS vol. 773, Springer-Verlag, pp. 40-48, 1994.

[33] J. Valkonen, N. Asokan, and K. Nyberg. *Ad Hoc Security Associations for Groups.* In Proceedings of the Third European Workshop on Security and Privacy in Ad hoc and Sensor Networks, Hamburg, Germany, 2006. LNCS vol. 4357, 150-164.

[34] S. Vaudenay. *Secure Communications over Insecure Channels Based on Short Authenticated Strings.* Advances in Cryptology - CRYPTO 2005, LNCS vol. 3621, 309-326.

[35] 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 The threshold value in relation to AXU and ASU

The following calculation, which will locate the value of ϵ above which Kabatianskii's ASU-bound becomes better than Stinson's ASU-bound, will demonstrate that the threshold value of ϵ has the same significance in the relationships between Stinson's and Kabatianskii's ASU-bound.[11]

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

$$\begin{aligned}\frac{K2^b}{\epsilon b} &\geq \frac{2^K(2^b-1)^2}{2^b\epsilon(2^K-1)+2^b-2^K} \\ \epsilon &\geq \frac{K2^{b+K}-K2^{2b}}{K2^{2b+K}-K2^{2b}-b2^{2b+K}+b2^{b+K+1}-b2^K}\end{aligned}$$

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

$$\epsilon > \frac{K2^{b+K}}{K2^{2b+K}-b2^{2b+K}} = \frac{K}{(K-b)2^b} = \left(1+\frac{b}{K-b}\right)2^{-b}$$

A similar calculation also leads us to conclude that Stinson's AXU-bound is overtaken by the AXU-variant of Kabatianskii's bound at the threshold value of ϵ.[12]

[12] On the one hand, $\epsilon > \left(1+\frac{b}{K-b}\right)2^{-b}$ is the same as $K > \frac{b}{\epsilon 2^b - 1} + b$. On the other hand, Kabatianskii's bound and its AXU-variant are valid when $K < \frac{2b}{\epsilon} - b$. Consequently, in order for these to make sense, we require $\frac{2b}{\epsilon} - b > \frac{b}{\epsilon 2^b - 1} + b$, which is the same as $\epsilon > \frac{2}{2^{b+1}-1}$. This is true, since $\epsilon > \left(1+\frac{b}{K-b}\right)2^{-b} > \frac{2}{2^{b+1}-1}$, which is equivalent to $b2^{b+1} > K$, derived from the condition of Kabatianskii's ASU-bound.