

New combinatorial bounds for universal families of hash functions

Abstract. We introduce a new lower bound on the key length, termed the *combinatorial* bound, in an ϵ -almost universal family of hash functions. This result gives us a lower bound on the bit-length of the hash key r with respect to the message bit-length K , the hash output bit-length b , and the hash collision probability ϵ . We derive the bound using combinatorial analysis, and also show how it corresponds to bounds for subsets of parameter values derived from the theory of error-correcting codes, and finite field arithmetic. We compare the bound against other well-known bounds of this and other universal families of hash functions. We discover that the value $\epsilon = (1 + \frac{b}{K-b})2^{-b}$ represents an important *threshold* in the behaviour of bounds, quantifying the *Wegman-Carter* effect. We then move on to analyse how tightly our new bound is satisfied by many known ways of computing universal families of hash functions, including error-correcting codes, polynomial hashing over finite fields, and square hash with small key size. In addition to the combinatorial bound, a new lower bound on the key length of an *almost XOR universal* family of hash function is introduced.

Keywords: universal families of hash functions, combinatorial analysis.

1 Introduction

Universal families of hash functions H with parameters (r, K, b) were proposed by Carter and Wegman [7, 30]. Each family consists of hash functions mapping a message representable by K bits into a hash output of b bits. In total, there are 2^r hash functions, indexed by a r -bit key k : $h_k(\cdot)$.

In this paper we introduce a new bound, termed the *combinatorial* bound, for an ϵ -almost universal family of hash functions (ϵ -*AU*). This result tells us the lower bound on the bit-length of the hash key with respect to a fixed amount of information we want to hash, the hash output bit-length, and the hash collision probability ϵ . We derive the bound using combinatorial analysis, and we are able to show how it corresponds to bounds for subsets of parameter values derived from the theory of error-correcting codes, and finite field arithmetic.

Although the topic has been studied extensively to date [25, 26, 10, 13–15, 30, 7, 10, 16, 2], most of these papers concentrate on theoretical bounds for an ϵ -almost *strongly-universal* family of hash functions (ϵ -*ASU*, a more restrictive version of ϵ -*AU*, as can be seen in their definitions below) because a much-used mechanism in practice, called Message Authentication Codes, make use of an ϵ -*ASU* [25, 26, 14, 15, 10, 13, 30, 7, 2]. We however believe that there is a similar potential for *AU* in practice. 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 traditional security infrastructures, such as PKI, in pervasive computing environments [24, 29, 8, 19–21, 28, 18]. Some of these protocols make use of a new cryptographic primitive, termed a *digest* function introduced in [19, 20], with similar security properties and purposes to an *AU*. In these mechanisms: (1) no party or any intruder can predict what the digest value will be until the digest key, which is fresh in each session of the protocol, is revealed, thanks to the principle of *commitment before knowledge* introduced in [19, 20], and hence a substitution attack (i.e. one in which the intruder can observe one or more authenticated messages and their hash values w.r.t the same key before launching an attack) is irrelevant; (2) input messages are influenced by the adversary. As a result, what we require is a protection against collision attacks (*AU*) as opposed to substitution (or interpolation) attacks (*ASU*).

The only published lower bound on the key length in an ϵ - AU , as far as we are aware, is due to Stinson [25, 26]. We compare our results against this bound, and will discover that Stinson’s bound is tight only in a short range of ϵ . As ϵ increases beyond the threshold value $(1 + \frac{b}{K-b})2^{-b}$, our results gives a tighter bound. Subsequently in Section 2.4, this threshold value will be shown to have the same theoretical significance in relationships between known bounds of *almost XOR universal* and *almost strongly universal* families of hash functions. What this threshold value illustrates is a behaviour of any universal family of hash functions, known as the “Wegman-Carter effect” in the literature [5, 17], previously reported in the work of Johansson, Kabatianskii and Smeets [12, 13]: if ϵ exceeds 2^{-b} (the theoretical minimum¹) 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.

We then move on to analyse how tightly our new bound is satisfied by many known ways of computing universal hash functions, including methods based on *error-correcting codes* [4], and *square hash* (SQH) with small key size [11]. We will further show in Sections 3.1 and 3.3 that a special case of our combinatorial bound can be derived from the theory of coding as well as being met with equality in the first version of *polynomial hashing* over finite fields [6, 23] for some values of (ϵ, K, b) .

It is worth to mention that in our work, we also introduce a new lower bound on the key length for an *almost XOR universal* family of hash functions (ϵ - AXU). The bound is derived from Kabatianskii’s ASU -bound [13] and a well-known relation between ASU and AXU [30, 9]. For this reason, we suspect that this has been known to the community. However, as far as we are aware, the bound has never been published, and moreover rigorously analysed in relative to other known bounds. We will show that the bound is met with equality in two different constructions: the second version of polynomial hashing, and square hash with small key size.

2 Theoretical bounds for *almost universal* families of hash functions: a combinatorial approach

In this section, we will present, and then give a proof for, a new bound of an ϵ - AU . This proof is based on a combinatorial approach. We therefore term this the *combinatorial* bound.

The following are the definitions of a number of well-studied universal families of hash functions, namely *almost universal* and *almost strongly universal* families of hash functions introduced by Carter and Wegman [7, 30], and an *almost XOR universal* family of hash functions introduced by Krawczyk [14, 15]. 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.² In all 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: $\Pr_k[\cdot]$.

An ϵ -almost universal family hash functions, ϵ - AU (r, K, b)
 H is an ϵ - AU iff for all different messages m and m' :
 $\Pr_k[h_k(m) = h_k(m')] \leq \epsilon$

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 [22].

² The terms collision, differential and interpolation probabilities were proposed by Bernstein in the appendix of [3] to distinguish the differences between these classes of universal hash functions.

<p>An ϵ-almost XOR universal family of hash functions, ϵ-AXU (r, K, b) H is an ϵ-AXU iff for every pair of distinct messages (m, m') and any $\omega \in \{0, 1\}^b$: $\Pr_k[h_k(m) \oplus h_k(m') = \omega] \leq \epsilon$</p>
--

<p>An ϵ-almost strongly universal family of hash functions, ϵ-ASU (r, K, b) (a) For every message m and hash output y: $\Pr_k[h_k(m) = y] \leq 2^{-b}$. (b) For every pair of distinct messages (m, m') and for every pair of hash outputs (y, y'): $\Pr_k[h_k(m) = y, h_k(m') = y'] \leq \epsilon 2^{-b}$</p>
--

All of the *universal* families of hash functions discussed to date are pairwise, since we look at their properties in relation to two different messages. We will see that our 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 the protocols of Laur and Pasini [18], Valkonen et al. [28], and Nguyen and Roscoe [19–21] (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 families of hash functions.

<p>A l-wise ϵ-almost universal family hash functions, ϵ-AU_l (r, K, b) H is an ϵ-AU_l iff for any l different messages $\{m_1, \dots, m_l\}$: $\Pr_k[h_k(m_1) = \dots = h_k(m_l)] \leq \epsilon$</p>

Although these parameters (r, K, b) do not have to be integers, the powers 2^r , 2^K and 2^b representing the cardinalities of the sets of all keys, input messages and hash outputs are. We assume the input message bit-length K is significantly greater than the hash bit-length b . Furthermore, whenever we use the term $\log X$, we refer to the logarithm of base 2 to simplify the notation.

2.1 Combinatorial bound

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

The following proof makes use of the *pigeon-hole* principle, which states that, 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. If we pick any key k_1 , then there will exist 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,³ 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 $r \geq \log(\epsilon^{-1} \lfloor (K-1)/b \rfloor)$ \square

³ 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.

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

We will see later that a special case of the bound, i.e. when K is a multiple of b , can be derived in Section 3.1, by looking at a construction based on error-correcting codes and then applying the well-known *singleton bound* of coding theory [1]. Moreover, when r is equal to b and K is a multiple of b , the exactness of our bound can be derived using finite field arithmetic in Section 3.3.

It is interesting to realise that the proof of our 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 $r \geq \log \left(\epsilon^{-1} \left\lfloor \frac{K - \log l}{b} \right\rfloor \right)$*

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* families of hash functions by Stinson [27], and Kurosawa et al. [16], as far as we are aware, this is the first result on l -wise *almost universal* families of 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. [13], as will be discussed in the sections to come.

2.2 Comparison between the combinatorial bound and known AU -bounds

Having discovered a new bound, it is essential to compare it with other bounds of not only ϵ - AU but also ϵ - AXU and ϵ - ASU to find out the significance and contribution of our result. We will give the comparative analysis in the following order: Stinson's AU -bound in this section, and in the next section we will study two ASU -bounds of Gemmell and Naor, and Kabatianskii et al., and then AXU -bounds.

Stinson's AU -bound [25] 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 (i.e. proving r to be much longer than ours does) for then it gives $r \geq K - b$, which means that the key bit-length grows at least linearly with the message bit-length.

In contrast, if one looks more closely into the formula for Stinson's bound above, one quickly realises that it gets large for $\epsilon = 2^{-b}$ because the part $2^K(\epsilon 2^b - 1)$ of the denominator of the fraction is eliminated. An example, showing how it can dramatically decrease as we increase ϵ , is given when we set $\epsilon = 2^{1-b}$, then setting $r = b$ satisfies the bound. In other words, as far as this bound is concerned, the key needs be no longer than the bit-length of the hash.

In order to explain the reason for the dramatic collapse, we will 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 bound 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 $(2^r \geq \epsilon^{-1} \lfloor \frac{K-1}{b} \rfloor)$ to $(2^r \geq \frac{K}{\epsilon b})$. This gives a very good approximation to the crucial value.

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

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

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

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 bit-length grows at most exponentially with the key bit-length as demonstrated in our 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 authentication codes or *almost strongly universal* families of hash functions and codes correcting independent errors in the work of Johansson, Kabatianskii, and Smeets [12, 13]. 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 bit-lengths of an input message and a hash output, then Stinson's *AU*-bound is still useful when $2^{-b} < \epsilon < \left(1 + \frac{b}{K-b}\right) 2^{-b}$.
2. More importantly, given any value of ϵ , which exceeds 2^{-b} by an arbitrarily small positive value, we will be able to determine the lower bound on the bit-length of input messages in relative to the hash bit-length and ϵ : $K \geq b + \frac{b}{2^b \epsilon - 1}$, above which the message bit-length can apparently start to grow exponentially with the key bit-length, i.e. our 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 -bounds and AXU -bounds in Section 2.4.

2.3 Comparison between the combinatorial bound and known ASU - and AXU -bounds

Having compared our result to AU -bounds, we turn our attention to ASU - and AXU -bounds. We note that there a number of existing ASU -bounds, introduced by Gemmell and Naor [10], and Kabatianskii et al. [13], 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 ϵ - AU should be smaller than in ϵ - ASU w.r.t the same set of parameters (ϵ, K, b) . This analysis is reflected by the following two comparisons:

- Our AU -bound, $r \geq \log(\epsilon^{-1} \lfloor \frac{K-1}{b} \rfloor)$, is smaller than Kabatianskii’s ASU -bound, $r \geq b + \log(\epsilon^{-1} \lfloor K/b \rfloor)$, by at least b bits.⁴
- The difference between our AU -bound and Gemmell-Naor’s ASU -bound,⁵ $r \geq \log K + 2 \log \epsilon^{-1} - \log \log \epsilon^{-1}$, 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,⁶ then it can be used to construct an ϵ - ASU $(r + b, K, b)$, thanks to the work of Wegman and Carter [30]. This can be summarised by the following theorem, adapted from Lemma 1 of [9] by Etzel, Patel and Ramzan.

Theorem 3. [30, 9]. *Let $H = \{h_k() : \{0, 1\}^K \rightarrow \{0, 1\}^b | k \in [0, 2^r]\}$ be an ϵ -almost XOR universal family of hash functions. Moreover, suppose H is also uniformly distributed. Then $H' = \{h'_{k,b'}() : \{0, 1\}^K \rightarrow \{0, 1\}^b | k \in [0, 2^r], b' \in [0, 2^b]\}$, 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 family of hash functions.*

This means that if we apply Theorem 3 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 4. *If there exists an ϵ - AXU (r, K, b) then $r \geq \log(\epsilon^{-1} \lfloor K/b \rfloor)$*

As pointed out in footnote 4 and [13], 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:⁷ $K < b\sqrt{2^{r+1}(1 - 2^{-b})} - b/2$.

⁴ We note that Kabatianskii’s ASU -bound, Theorem 15 of [13], 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 $K < \frac{2^b}{\epsilon} - b$, yielding a very large K in practice when the hash length is in the range from 80 to 160 bits.

⁵ We note that the bound was reported in the paper of Gemmell and Naor [10] (Section 5.1). However, it was noted there that the bound was actually introduced by Noga Alon through private communication.

⁶ 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: $\Pr_k[h_k(m) = y] \leq 2^{-b}$.

⁷ The exponent inside the square root operator is $r + 1$ instead of $r - b + 1$ as in the original formula because the key bit-length of an ϵ - AXU in this case is exactly b bits shorter than in an ϵ - ASU .

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 ($2^r \geq \frac{2^K(2^b-1)}{2^b\epsilon(2^K-1)+2^b-2^K}$) [26]. 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 3.3 and *square hash* with small key size in Section 3.4.

2.4 The threshold value in relation to AXU and ASU

Stinson's bounds for AXU and ASU have similar forms to his AU -bound. It is therefore reasonable to suspect that they are only tight in a short range of ϵ . 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.

This will be justified by the following calculation, which will locate the value of ϵ above which Kabatianskii's ASU -bound becomes better than Stinson's ASU -bound.⁸

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

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 .⁹ We end this section with Table 1 that captures the interesting relationships between our combinatorial AU -bound, Kabatianskii's ASU -bound and the AXU -variant of Kabatianskii's bound, and Stinson's bounds on AU , AXU and ASU .

3 Verification of the combinatorial bound

What we have shown in the previous section is that the combination of the *combinatorial* and Stinson's bounds gives a better result than either individually over the range of values of ϵ in an

⁸ 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 $2^r \geq \frac{2^b}{\epsilon} \lfloor \frac{K}{b} \rfloor$ to $2^r \geq \frac{2^b K}{\epsilon b}$.

⁹ 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.

	$\epsilon < \left(1 + \frac{b}{K-b}\right) 2^{-b}$	$\epsilon > \left(1 + \frac{b}{K-b}\right) 2^{-b}$
ϵ - <i>AU</i>	Stinson's bound [25, 26] $\log \left(\frac{2^K(2^b-1)}{2^K(\epsilon 2^b-1)+2^{2b}(1-\epsilon)} \right)$	Combinatorial bound <i>New</i> , Theorem 1, Section 2.1 $\log \left(\epsilon^{-1} \lfloor \frac{K-1}{b} \rfloor \right)$ A special case of the combinatorial bound (When K is a multiple of b) <i>New</i> , Theorem 6, Section 3.1 $\log \frac{K-b}{eb}$
ϵ - <i>AXU</i>	Stinson's bound [26] $\log \left(\frac{2^K(2^b-1)}{2^b\epsilon(2^K-1)+2^b-2^K} \right)$	<i>AXU</i> -variant of Kabatianskii's bound <i>New</i> , Theorem 4, Section 2.2 $\log \left(\epsilon^{-1} \lfloor \frac{K}{b} \rfloor \right)$
ϵ - <i>ASU</i>	Stinson's bound [25, 26] $\log \left(1 + \frac{2^K(2^b-1)^2}{2^b\epsilon(2^K-1)+2^b-2^K} \right)$	Kabatianskii's bound [13] $b + \log \left(\epsilon^{-1} \lfloor \frac{K}{b} \rfloor \right)$ Gemmell and Noar's bound [10] (introduced by Noga Alon through private communication) $\log K + 2 \log \epsilon^{-1} - \log \log \epsilon^{-1}$

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

AU . However, it is not clear whether this is the tightest possible bound. It is known that when $\epsilon = 2^{-b}$, Stinson's AU -bound is met with equality by a construction based on the first order Reed-Muller code [26]. What remains to be discovered is how tightly our combinatorial bound is satisfied when ϵ moves away from 2^{-b} in a number of established algorithms for computing an ϵ - AU taken from the literature.

We first present how to derive a special case of our combinatorial AU -bound from the use of error-correcting codes (ECC) and the *singleton* bound of coding theory. We then turn into *polynomial hashing* over finite fields because, using this algorithm, it is possible to build an ϵ - AU that meets our combinatorial bound with equality for any value of ϵ of the form $\frac{k}{q}$, where q is a power of a prime number, and k is a positive integer less than q .

Every construction presented after the first version of polynomial hashing will produce an AXU . For this reason, we analyse how tightly these constructions satisfy the AXU -variant of Kabatianskii's bound. We argue that this does not devalue our goal of verifying the combinatorial bound, since the AXU -bound is only slightly bigger than our combinatorial AU -bound, as pointed out earlier. We are able to demonstrate that the AXU -bound is met with equality by the second version of polynomial hashing as well as *square hash* with small key size.

3.1 Error-Correcting Codes

In this section, we use a known relation between error-correcting codes and universal hash functions to give a different way of deriving a special case of our combinatorial AU -bound, where K is a multiple of b .

Let $(n, t, d_H(V), q)$ be a q -ary error-correcting code, whose code-word and data-word are n and t in symbols, and the minimum Hamming distance is $d_H(V)$.

The following equivalence between error-correcting codes and *almost universal* families of hash functions was first observed by Bierbrauer, Johansson, Kabatianskii and Smeets [4].

Theorem 5. [4, 26]. *If there exists an ϵ - AU (r, K, b) , then there exists an $(n = 2^r, t = K/b, d_H(V) = 2^r - 2^r \epsilon, q = 2^b)$ code. Conversely, if there exists an $(n, t, d_H(V), q)$ code, then there exists an $(\epsilon = 1 - d_H(V)/n)$ - AU $(r = \log n, K = t \log q, b = \log q)$.*

We therefore can derive the following theorem.

Theorem 6. *A special case of our combinatorial bound on an ϵ - AU (r, K, b) , where K is a multiple of b , is: $r \geq \log \frac{K-b}{\epsilon b} = \log \left(\epsilon^{-1} \left(\frac{K}{b} - 1 \right) \right)$*

Proof. Using Theorem 5, construct an $(n = 2^r, t = \frac{K}{b}, d_H(V) = 2^r - 2^r \epsilon, q = 2^b)$ code from the universal hash family. This code must meet the singleton bound [1], so we obtain.

$$\begin{aligned} n - t + 1 &\geq d_H(V) \\ 2^r - \frac{K}{b} + 1 &\geq 2^r(1 - \epsilon) \\ r &\geq \log \left(\epsilon^{-1} \left(\frac{K}{b} - 1 \right) \right) = \log \left(\epsilon^{-1} \left\lfloor \frac{K-1}{b} \right\rfloor \right) \end{aligned}$$

Note that the equality between $\frac{K}{b} - 1$ and $\left\lfloor \frac{K-1}{b} \right\rfloor$ holds because K is a multiple of b . □

A strong connection between universal families of hash functions and coding theory has been extensively investigated to date, for example [25, 26, 13, 4, 12].

In order to meet the bound with equality under this construction, the underlying codes must meet the singleton bound with equality. As far as we are aware, the only well-studied codes meeting the bound are the Reed-Solomon codes that will be discussed in the next section.

3.2 The use of the Reed-Solomon code

We now look into constructing an $(\epsilon = 2^{-b})$ - AU based on the Reed-Solomon (RS) error-correcting codes. Since RS codes meet the singleton bound with equality, the corresponding 2^{-b} - AU meets its combinatorial bound with equality, i.e. $r = b + \log\left(\frac{K}{b} - 1\right)$. This construction, perhaps surprisingly, is a special case where the equality between our combinatorial AU -bound and Stinson's AU -bound occurs, as when $\epsilon = 2^{-b}$ Stinson's AU -bound ($r \geq K - b$) must be satisfied.

We recall two main properties of RS codes to analyse how they affect the choice of parameters in the constructed AU .

- The number of symbols in a codeword is equivalent to the range of a symbol. In other words, $n = 2^b$, which is the same as $r = b$, since $n = 2^r$.
- As RS codes meet the singleton bound with equality, we have: $d_H(V) = n - t + 1$.

These properties and our combinatorial AU -bound suggest that $r = b + \log(K/b - 1)$, which means that $K = tb = 2b$.

This result demonstrates that if one uses RS codes to build an 2^{-b} - AU , then only messages, which are representable by twice the bit-length of the hash output, can be hashed. This means that Stinson's AU -bound is met with equality, since $r = b = K - b$. This also shows that these constructions based on RS codes are of limited use in many applications, where $K \gg b$.

In practice, one constructs RS codes by using polynomials over finite fields, and therefore the construction has many similarities to polynomial hashing, discussed in the following section.

3.3 Evaluation hash function or Polynomial hashing over finite fields

Polynomial hashing over finite fields was introduced by Boer [6], and was analysed further by Shoup [23]. Our discussion here shows that two slightly different versions of polynomial hashing meet our combinatorial AU -bound and the AXU -variant of Kabatianskii's bound with equality, respectively.

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 bit-length 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 , $h_k(m)$, is equivalent to $m(k)$ over \mathbb{F}_q . This implies that bit-lengths of the key and the hash output are equal to each other, i.e. $\log q = b = r$.

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

If we fix two different messages A and B , where $B = A \oplus m$,¹⁰ then a hash collision is equivalent to: $0 = h_k(A) \oplus h_k(B) = A(k) \oplus B(k) = m(k)$. Since the polynomial $m(k)$ is of degree up to $(t-1)$, there

¹⁰ \oplus denotes subtraction over the finite field \mathbb{F}_q .

are at most $t - 1$ different roots out of total q possible values of key k , causing a hash collision. This therefore implies that $\epsilon = (t - 1)q^{-1} = \lfloor \frac{K-1}{b} \rfloor 2^{-r}$, which is equivalent to $r = \log(\epsilon^{-1} \lfloor \frac{K-1}{b} \rfloor)$, here the equality between $t - 1$ and $\lfloor \frac{K-1}{b} \rfloor$ holds because K is a multiple of b . This result demonstrates that the construction meets the combinatorial bound with equality whenever ϵ is of the form $\frac{t-1}{q}$, which justifies our claim made at the end of the second paragraph of Section 3.

We note that the construction above is not an AXU because if we set $\omega = A_1 \oplus B_1$, and for all $i \in (1, t]$: $A_i = B_i = 0$, then even though A and B are different messages, we always have:

$$\Pr_k[h_k(A) \oplus h_k(B) = \omega] = \Pr_k[A_1 \oplus B_1 = \omega] = 1$$

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

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

Since the degree of this polynomial is up to t , a similar calculation leads us to conclude that this forms an $(\epsilon = \frac{t}{q})$ - 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 2.3 and [13], 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 $[1, q = 2^b)$.

It is interesting to note that the AXU -variant of Kabatianskii's bound can also be met with equality in a variant of polynomial hashing over finite fields, as will be introduced in the next section.

3.4 Square hash (SQH) with small key size

Heng and Kurosawa [11] introduced another construction for an *almost XOR universal* family of hash functions, namely *square hash* with small key size, which is based on the idea of *Square Hash* of Etzel et al. [9].

The input K -bit message is divided into t blocks, where each block is of $b = \log q$ bits, i.e. $m = \langle m_1, \dots, m_t \rangle$, here $m_i \in \mathbb{F}_q$. In this scheme, the bit-lengths of the key and the hash output are equal to each other: $r = b = \log q$. Furthermore, we use the key to derive $\langle x_1, \dots, x_k \rangle$, where $x_i = k^i$. The SQH , forming an $(\epsilon = \frac{t}{q})$ - AXU , is defined as follows:

$$h_k(m) = \sum_{i=1}^t (m_i \oplus x_i)^2$$

When we substitute the set of parameters $(\epsilon = t/q, r = \log q, K = tb, b = \log q)$ into the AXU -variant of Kabatianskii's bound, it is again met with equality: $\log(\epsilon^{-1} \lfloor K/b \rfloor) = \log q = r$.

4 Conclusions and future research

In this paper, we have discovered a new combinatorial AU -bound using combinatorial analysis. We then showed how it corresponds to bounds for subsets of parameter values derived from coding theory, and finite field arithmetic. 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 .

For each universal family of hash functions, there are several lower bounds on the key length, working in different ranges of ϵ , one should therefore hope to be able to unify these bounds into a single one that captures the Wegman-Carter effect. Equally, these classes of universal hash functions are strongly related to one another, i.e. ASU is a stronger version of AXU , which is, in turns, a generalisation of AU . Hence, we plan to explore the possibility of unifying the various bounds of these classes. We note that works reported in [25, 26, 13, 12, 4] make use of known bounds on classical combinatorial structures, such as *error-correcting code*, *difference matrices*, *orthogonal arrays*, and *balanced incomplete block design*. One should hope to be able to find other ways to transfer between other combinatorial designs, such as *Latin Squares*, and combinations of different families of universal hash functions. Possibly, better bounds can be obtained by that way. In particular we hope to find AU - and other families of universal hash functions which satisfy the known bounds and where we are able to choose r significantly larger than the hash output width, thereby bringing ϵ much closer to 2^{-b} .

We have also illustrated, in the l -wise variant of the combinatorial bound, how the inclusion of further parameters can capture wider range of security properties.

In practice we believe that one of the most important problems to solve is how to compute efficient digest functions highly efficiently, thanks to applications such as those in [21].

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. D.J. Bernstein. *Stronger security bounds for Wegman-Carter-Shoup authenticators*. Advances in Cryptology - Eurocrypt 2005, LNCS vol. 3497, Springer-Verlag, pp. 164-180, 2005.
4. 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'93, LNCS vol. 773.
5. J. Bierbrauer. *Introduction to Coding Theory*. (pages 240-241). Published by CRC Press, 2004. ISBN 1584884215, 9781584884217.
6. B. den Boer. *A simple and key-economical unconditional authentication scheme*. Journal of Computer Security 2 (1993), 65-71.
7. J.L. Carter and M.N. Wegman. *Universal Classes of Hash Functions*. Journal of Computer and System Sciences, 18 (1979), pp. 143-154.
8. 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, September 2003.
9. M. Etzel, S. Patel, and Z. Ramzan. *SQUARE HASH : Fast message authentication via optimized universal hash functions* Advances in Cryptology - CRYPTO '99, vol. 1666, LNCS.
10. P. Gemmell and M. Naor. *Codes for Interactive Authentication*. Advances in Cryptology - CRYPTO '93, vol. 773, LNCS.
11. S.-H. Heng and K. Kurosawa. *Square hash with a small key size*. Eighth Australasian Conference on Information Security and Privacy, ACISP '03, LNCS 2727, pp. 522-531, Springer-Verlag, 2003.
12. T. Johansson, G.A. Kabatianskii, and B. Smeets. *On the relation between A-Codes and Codes correcting independent errors*. EUROCRYPT 1993, LNCS 765, pp. 1-11.
13. 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), pp. 566-578.
14. H. Krawczyk. *LFSR-based Hashing and Authentication*. CRYPTO 1994, LNCS vol. 839, pp. 129-139.
15. H. Krawczyk. *New Hash Functions For Message Authentication*. EUROCRYPT 1995, LNCS vol. 921, pp. 301-310.
16. K. Kurosawa, K. Okada, H. Saido, and D.R. Stinson. *New combinatorial bounds for authentication codes and key redistribution schemes*. Designs, Codes and Cryptography, 15 (1998), 87-100.

17. K. Kurosawa, S. Obana. *Combinatorial Bounds on Authentication Codes with Arbitration*. Des. Codes Cryptography 22 (3): 265-281 (2001).
18. S. Laur and S. Pasini. *SAS-Based Group Authentication and Key Agreement Protocols*. In Public Key Cryptography - PKC, pages 197-213, 2008.
19. 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, pp. 9-31, August 2006.
20. L.H. Nguyen and A.W. Roscoe. *Authenticating ad hoc networks by comparison of short digests*. Journal of Information and Computation 206 (2008), 250-271.
21. L.H. Nguyen and A.W. Roscoe. *Separating two roles of hashing in one-way message authentication*. Proceedings of FCS-ARSPA-WITS 2008.
22. D.V. Sarwate. *A note on universal classes of hash functions*. Inf. Process. Lett., 10(1):41-45, 1980.
23. V. Shoup. *On Fast and Provably Secure Message Authentication Based on Universal Hashing*. Advances in Cryptology - CRYPTO '96, Vol. 1109, LNCS.
24. F. Stajano and R. Anderson. *The resurrecting duckling: Security issues for ad-hoc wireless networks*. Security Protocols 1999, LNCS vol. 1976, Springer-Verlag, pp. 172-194, 1999.
25. D.R. Stinson. *Universal Hashing and Authentication Codes*. Advances in Cryptology - Crypto 1991, LNCS vol. 576, Springer-Verlag, pp. 74-85, 1992.
26. D.R. Stinson. *On the Connections Between Universal Hashing, Combinatorial Designs and Error-Correcting Codes*. Congressus Numerantium, vol. 114, pp. 7-27, 1996.
27. D.R. Stinson. *The combinatorics of authentication and secrecy codes*. Journal of Cryptology 2 (1990), 23-49.
28. 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, September 2006. Vol. 4357 of LNCS, Springer.
29. S. Vaudenay. *Secure Communications over Insecure Channels Based on Short Authenticated Strings*. Advances in Cryptology - Crypto 2005, LNCS vol. 3621, Springer-Verlag, pp. 309-326, 2005.
30. 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, pp. 265-279, 1981.