On Quantifying the Resistance of Concrete Hash Functions to Generic Multi-Collision Attacks

Somindu C. Ramanna and Palash Sarkar

Applied Statistics Unit,
Indian Statistical Institute,
203, B.T. Road, Kolkata,
India 700108.
email: somindu_r@isical.ac.in, palash@isical.ac.in

October 31, 2009

Abstract

Bellare and Kohno (2004) introduced the notion of balance to quantify the resistance of a hash function h to a generic collision attack. Motivated by their work, we consider the problem of quantifying the resistance of h to a generic multi-collision attack. To this end, we introduce the notion of r-balance $\mu_r(h)$ of h and obtain bounds on the success probability of finding an r-collision in terms of $\mu_r(h)$. These bounds show that for a hash function with m image points, if the number of trials q is $\Theta(rm^{(\frac{r-1}{r})\mu_r(h)})$, then it is possible to find r-collisions with a significant probability of success. It is further shown that compared to regular functions, random functions offer somewhat lesser resistance to a generic multi-collision attack. These results extend and complete the earlier results obtained by Bellare and Kohno (2004) for collisions (i.e., $r = 2$).

1 Introduction

An (n, m)-hash function is a map $h : X \to Y$, where $|X| = n$, $|Y| = m$ and $n > m > 0$. A *collision* for h is a pair of *distinct* points $x, x' \in X$ such that $h(x) = h(x')$. Since $n > m$, collisions necessarily exist. For cryptographic applications, h should be designed such that it is infeasible for a resource-bounded adversary to find a collision for h. Such a function is called *collision resistant*. The notion of a collision has been generalized to that of a multi-collision. An r-way collision (or r-*collision*) consists of r *distinct* domain points $x_1, x_2, \cdots, x_r$ such that, $h(x_1) = h(x_2) = \cdots = h(x_r)$. Again, for certain cryptographic applications, the design goal is to ensure that for some suitable range of r, r-collisions are hard to find for a resource-bounded adversary.

Given a hash function h, an algorithm to find an r-collision for h is called an attack. A generic attack does not consider the manner in which the function h is defined, i.e., it does not consider the "internal structure" of h. Instead, some points are picked from the domain and h is applied to them with the hope that a subset of the points will yield an r-collision. Suppose that q points $x_1, x_2, \cdots, x_q$ are picked. Then the probability of obtaining an r-collision increases monotonically with q. The domain points on which to apply h can be chosen in different ways.

1. **Sampling without replacement.** An r-collision by definition requires the domain points to be distinct. Hence, one would like to use uniform random sampling without replacement to select the

domain points. In particular, x_i is selected uniformly at random from $X \setminus \{x_1, \ldots, x_{i-1}\}$. Since it has to be ensured that x_i is distinct from $x_1, \ldots, x_{i-1}$, this method is not very convenient to implement. Also, the lack of independence among the x_i's makes it more difficult to analyse this scenario.

2. **Sampling with replacement.** In this method the domain points are independent and uniformly distributed, i.e., x_i is distributed uniformly over X and is independent of the previous choices. From an algorithmic point of view, this is much more simpler to implement than sampling without replacement.

3. **Picking distinct points without sampling.** Suppose that h is a uniform random function from X to Y. Then it is pointless to use a sampling strategy for picking the domain points. One can simply pick any q distinct points, apply h to them and look for a collision. The probability of success does not depend on the particular set of q points that has been picked. This can also be considered to be the uniform random distribution of q balls to m bins and then looking for a bin with at least r balls.

   In this formulation, the problem has been studied in the literature. McKinney [McK66] gives an exact formula for the probability of finding r-collisions in q trials. But this formula gets more difficult to evaluate as r grows. One can also express this probability using a multinomial cumulative distribution function. Levin [Lev81] provides an efficient way to compute a multinomial distribution function by expressing it as the conditional distribution of independent Poisson random variables given fixed sum. These approximations, however, provide little intuition on the asymptotic behaviour of the complexity of finding an r-collision. It is well known that this complexity is $\Theta(rm^{(r-1)/r})$. (See [Pre93] for a proof.) For $r = 2$, the complexity is $\Theta(m^{1/2})$ and the attack is usually called the *birthday attack*.

A measure of collision resistance of a hash function is the success probability of a generic attack of the above kinds in finding a collision. Most works in the literature ignore the actual hash function and instead analyse a random function. It is then (implicitly) implied that the results for a random function also hold for the actual hash function.

This approach has been eloquently criticised by Bellare and Kohno [BK04]. They argue that, given a concrete hash function h, one cannot assume that h has "random behaviour", since then, one ends up "not analyzing the given h, but rather analyzing an abstract and ideal object which ultimately has no connection to h, regardless of the design principle underlying h".

The specific case of $r = 2$ (i.e., collisions) is considered in [BK04]. Suppose that the domain points $x_1, \ldots, x_q$ are chosen using sampling with replacements as explained above. Then, it is usually assumed that the birthday attack applies to the hash function h. Bellare and Kohno [BK04] explain the drawback of this argument. Suppose that a point x is drawn uniformly at random from X. Then it does not follow that the point $h(x)$ is uniformly distributed over Y. Instead, the probability that a $h(x)$ equals a particular $y \in Y$ is $|h^{-1}(y)|/|X|$, where $h^{-1}(y)$ is the set of all pre-images of y under h. So the points $h(x_1), \ldots, h(x_q)$ are uniformly distributed over Y if and only if h is *regular*, i.e., every range point has the same number of pre-images under h. This need not be true for the particular hash function under consideration. In fact, Bellare and Kohno [BK04] comprehensively cover textbook discussions of birthday attacks on hash functions and point out the inadequate and sometimes incorrect viewpoints been provided.

Having exposed the fallacy in the analysis of collision resistance of a *concrete* hash function h, Bellare and Kohno [BK04] turn to the problem of quantifying the collision resistance of h. They introduce an important measure $\mu(h)$, called the *balance* of a hash function h. This is defined to be $\mu(h) = -\log_m((n_1^2 + \cdots + n_m^2)/n^2)$, where $Y = \{y_1, \ldots, y_m\}$ and n_i is the number of pre-images of y_i. In other words, $-\mu(h)$ is the logarithm of the probability that $h(x_i) = h(x_j)$ for $i \neq j$. Note that this includes the possibility that $x_i = x_j$ which is a trivial collision, i.e., $-\mu(h)$ is the logarithm of the probability of obtaining a possibly trivial collision. The rationale for considering possibly trivial collisions in the definition of balance is that if n is large, then with high probability it is a proper collision.

An extensive analysis is carried out to quantify the collision resistance of h in terms of the balance. To this end, two quantities are introduced: $C_h(q)$ and $Q_h(c)$, where $C_h(q)$ is the probability of finding a collision in q trials and $Q_h(c) = \min\{q : C_h(q) \geq c\}$ is the minimum number of queries required to find a collision with probability c. Bounds on $C_h(q)$ are obtained in terms of the balance $\mu(h)$ and these bounds are then translated to obtain bounds on $Q_h(c)$. Section 2 summarizes the bounds that they obtain. They further show that regular functions offer (slightly) better collision resistance compared to random functions.

1.1 Our Contributions

The work done by Bellare and Kohno in [BK04] is for $r = 2$. We continue and to a certain extent complete the work started in [BK04] by considering r-collisions for arbitrary $r \geq 2$. As noted above, like [BK04], we also work in the setting where the domain points are chosen according to uniform random sampling with replacement. We call this the generic multi-collision attack. The first question that we consider is the following.

> What is the notion of balance of an (n, m)-hash function h in the context of r-collisions?

To answer this question, we introduce $\mu_r(h)$ which we call the r-balance of the function h. This is defined to be $-(\log_m p_r)/(r-1)$, where p_r is the probability that r points chosen indepedently and uniformly at random from the domain form an r-collision. As in [BK04], this notion then leads to the following question.

> How is the performance of the generic multi-collision attack for finding r-collisions related to the notion of r-balance?

Similar to [BK04], we study two quantities.

1. $C_h^{(r)}(q)$. This is the probability of finding an r-collision in q trials.

2. $Q_h^{(r)}(c)$. This is the number of queries required to find an r-collision with probability c.

Upper and lower bounds are obtained on $C_h^{(r)}(q)$. These bounds on $C_h^{(r)}(q)$ are translated to obtain upper and lower bounds on $Q_h^{(r)}(c)$. From this it follows that for an (n, m)-hash function, the number of queries required to find an r-collision with significant probability is $\Theta(rm^{\frac{r-1}{r}\mu_r(h)})$.

Following the agenda set out in [BK04], we next consider a uniform random (n, m)-hash function and introduce $C_{n,m}^{\$(r)}(q)$ (resp. $Q_{n,m}^{\$(r)}(c)$), which is the probability (resp. number of queries) for finding an r-collision with q queries (resp. probability c). Again bounds on $C_{n,m}^{\$(r)}(q)$ are obtained which are used to obtain bounds on $Q_{n,m}^{\$(r)}(c)$. It is shown that if h is a regular (n, m)-hash function, then for a certain range of q, the upper bound on $C_h^{(r)}(q)$ is lesser than a lower bound on $C_{n,m}^{\$(r)}(q)$. As a consequence, using the same number of queries, the probability of finding an r-collision for a regular function is lesser than that of a uniform random function. This shows that compared to random functions, regular functions provide better resistance to the generic multi-collision attack.

In Section 5 we provide bounds on the expected number of trials to obtain an r-collision. For collisions, this was done by Bellare and Kohno and we simply adapt their general arguments with the bounds obtained in this paper.

Textbook discussion. Most textbooks analyse collisions obtained by the birthday attack. As mentioned earlier, such analysis is sometimes inadequate and sometimes incorrect as discussed in [BK04]. On the other hand, to the best of our knowledge, no textbook analyses r-collisions with respect to the generic multi-collision attack. The only analysis available in the literature is using the "balls and bins" approach as discussed above.

Applicability to actual hash functions. This point has already been discussed in detail by Bellare and Kohno [BK04]. We only note the following point. Hash functions such as SHA-2 can take as input, strings of arbitrary length. Like Bellare and Kohno, we work with hash functions with a finite domain. So, to apply our results, one would have to restrict the domain of SHA-2 to strings of some maximum length. This makes sense, since r-collisions for the restricted domain hash function are also r-collisions for the unrestricted hash function.

Relation to the work of Bellare and Kohno [BK04]. At a general level, we follow the path set out in [BK04]. The results that we obtain for general r are in a way already anticipated by the results for $r = 2$ in [BK04]. Having said this, we would also like to note that our analysis and proofs are not straightforward extensions of [BK04]. Some of the important differences are noted below.

Definition of balance. A straightforward extension of the Bellare and Kohno's definition of balance will be based on the logarithm of $(n_1^r + \cdots + n_m^r)/n^r$. The quantity $(n_1^r + \cdots + n_m^r)/n^r$ is the probability that $h(x_1) = \cdots = h(x_r)$ when $x_1, \ldots, x_r$ are sampled with replacement from the domain. This would include possibly trivial r-collisions, i.e., it would include the possibility that $x_i = x_j$ for some $i \neq j$.

The definition of r-balance that we define is based on the probability of actual r-collisions and not possibly trivial r-collisions. As we show later, this probability is $((n_1)_r + \cdots + (n_m)_r)/n^r$, where $(n_i)_r = n_i(n_i - 1) \cdots (n_i - r + 1)$. This expression is somewhat more complicated, but, we are able to satisfactorily analyse it. The advantage is that our bounds are better than what would be obtained otherwise.

Lower bound on the success probability. In [BK04], the lower bound on $C_h(q)$ is shown to hold only for a certain range of q.

In contrast, the lower bound on $C_h^{(r)}(q)$ that we obtain holds for all q. This is a consequence of the fact that $C_h^{(r)}(q)$ is monotone increasing in q. (Similarly, $C_h(q)$ is also monotone increasing in q, but, [BK04] do not consider the consequences of this fact.)

Upper bound on the number of queries. The lower bound on success probability translates into an upper bound on the number of queries. This the upper bound holds for a larger range of probability.

We note an issue of interpretation. In [BK04], it is mentioned that the bounds on $Q_h(c)$ are meaningful only for a certain range of c. But, more precisely, as we point out later, the lower bound on $Q_h^{(r)}(c)$ holds for all c, while the upper bound holds only for a certain range of c. This means that for a value of c outside this range, we cannot upper bound the number of queries required to obtain success probability c. But, we still can say that at least a certain number of queries will be required to obtain success probability c.

1.2 Related Work

The property of r-collision freeness has been suggested as a useful tool in building cryptographic protocols. It has been used for the micropayment scheme Micromint of Rivest and Shamir [RS96], for identification schemes by Girault and Stern [GS94] and for signature schemes by Brickell *et. al.* [BPVY00].

The intuition behind relying on r-collision freeness is that finding multi-collisions is harder than finding collisions. This is true when the function is truly random. But concrete hash functions mostly lack "random behaviour". For the case of hash functions based on an iterated construction, Joux [Jou04] has demonstrated that r-collisions in iterated hash functions are not much harder to find than ordinary collisions, even for very large values of r. Following Joux's attack, several works [NS07, HS06] have extended the attack to more general classes of constructions.

There are several space efficient algorithms that find cycles in random graphs. These methods can be used to find collisions in a hash function. It would be interesting to find space efficient algorithms to find multi-collisions. This problem has been addressed recently by Joux and Lucks in [JL09]. They give an algorithm to find 3-collisions that roughly uses m^{δ} storage and whose running time is $m^{1-\delta}$ for $\delta \leq 3$. This shows that finding 3-collisions in time $m^{2/3}$ would require $m^{1/3}$ units of storage.

2 Bounds Obtained by Bellare and Kohno [BK04]

The following results summarize the bounds on $C_h(q)$ and $Q_h(c)$ obtained in [BK04].

Theorem 2.1. *[BK04] Let h be an (n, m)-hash function and $m \geq 2$. Let $\alpha \geq 0$ be any real number. Then for any integer $q \geq 2$*

$$(1 - \alpha^2/4 - \alpha) \cdot \binom{q}{2} \cdot \left(\frac{1}{m^{\mu(h)}} - \frac{1}{n} \right) \leq C_h(q) \leq \binom{q}{2} \cdot \left(\frac{1}{m^{\mu(h)}} - \frac{1}{n} \right), \tag{1}$$

the lower bound being true under the additional assumption that

$$q \leq \alpha \cdot \left(1 - \frac{m}{n}\right) \cdot m^{\mu(h)/2}. \tag{2}$$

Theorem 2.2. *[BK04] Let h be an (n, m)-hash function and $n \geq 2m \geq 4$. Let $\alpha \geq 0$ be any real number such that $\beta = 1 - \alpha^2/4 - \alpha > 0$. Let c be a real number in the interval $0 \leq c < 1$. Then*

$$\sqrt{2c} \cdot m^{\mu(h)/2} \leq Q_h(c) \leq 1 + \sqrt{\frac{4c}{\beta}} \cdot m^{\mu(h)/2}, \tag{3}$$

the upper bound being true under the additional assumption that

$$c \leq (\alpha \cdot (1 - m/n) - m^{-\mu(h)/2})^2 \cdot \frac{\beta}{4}. \tag{4}$$

3 Balance-Based Analysis of the Generic Multi-Collision Attack

The *generic multi-collision attack* that we consider is the following. Given an (n, m)-hash function $h : X \to Y$ do the following.

1. Pick $x_1, \ldots, x_q$ independently and uniform at random from X.
2. Compute $y_i = h(x_i)$ for $1 \leq i \leq q$.

An r-collision is found if there are indices $i_1, \ldots, i_r$ with $1 \le i_1 < i_2 < \cdots < i_r \le q$ such that $y_{i_1} = \cdots = y_{i_r}$ *and* the domain points $x_{i_1}, \ldots, x_{i_r}$ are distinct. To find an r-collision we certainly need $q \ge r$.

Our goal here is to analyse the performance of the generic multi-collision attack in terms of what we call the r-balance of h. Equivalently, we want to analyse how the following quantities vary with r-balance.

- $C_h^{(r)}(q)$: probability that an r-collision for h is found in q trials ($q \ge r$). This function is monotonically increasing in q since the probability of finding collisions cannot decrease as the number of trials increases.
- $Q_h^{(r)}(c)$: the minimum number of trials required to obtain an r-collision with probability greater than or equal to c. That is,
$$Q_h^{(r)}(c) = \min\{q : C_h^{(r)}(q) \ge c\}. \tag{5}$$
Higher the value of c, more is the number of trials needed to find an r-collision. Hence $Q_h^{(r)}(c)$ is monotonically increasing in c.

Note that, for a balance-based analysis of the generic multi-collision attack, the definition of balance given in [BK04] will not be useful. We need to define balance in the context of r-collisions. From the definition, it follows that $C_h^{(2)}(q) = C_h(q)$ and $Q_h^{(2)}(c) = Q_h(c)$.

3.1 Notation

If d is a non-negative integer, then $[d] = \{1, 2, \cdots, d\}$. For an integer $r \ge 2$, $[d]_r$ denotes the set of all r-element subsets of $[d]$. $[d]_{r,2}$ denotes the set of all 2-element subsets of $[d]_r$. Let $r \ge 2$ and $d \ge 0$ be integers. Then $(d)_r$ is defined as follows.

$$(d)_r = \begin{cases} d(d-1)\cdots(d-r+1) & \text{if } d \ge r \\ 0 & \text{otherwise} \end{cases}$$

Let $h : X \to Y$ be an (n, m)-hash function. For any $y \in Y$, $h^{-1}(y) = \{x \in X : h(x) = y\}$. Let $Y = \{y_1, y_2, \cdots, y_m\}$. Then for $i \in [m]$, $n_i = |h^{-1}(y_i)|$ denotes the size of the set of pre-images of y_i under h.

3.2 Definition of r-Balance

A natural way to define the r-balance of h would be in terms of the probability of finding r-collisions for h. To this end, we first prove the following result.

Proposition 3.1. *Let $h : X \to Y$ be a hash function whose domain X and range $Y = \{y_1, y_2, \cdots, y_m\}$ have sizes $n, m \ge r$, respectively. For $i \in [m]$, let $n_i = |h^{-1}(y_i)|$ denote the size of the pre-image of y_i under h. Let r elements be chosen independently and uniformly at random from the domain X. The probability that they form an r-collision is*
$$p_r = \frac{\sum_{i=1}^{m} (n_i)_r}{n^r}.$$

Proof. Let r elements $w_1, w_2, \cdots, w_r$ be picked independently and uniformly at random from the domain X. Let E be the event that these elements form an r-collision. Let A denote the event that these are distinct and for $1 \le i \le m$, let B_i be the event that $h(w_1) = \cdots = h(w_r) = y_i$. Then $E = AB_1 \cup AB_2 \cup \cdots \cup AB_m$.

Since B_i's are mutually exclusive events, we have

$$\begin{aligned}\Pr[E] = \sum_{i=1}^{m}\Pr[AB_i] = \sum_{i=1}^{m}\Pr[A|B_i]\cdot\Pr[B_i] &= \sum_{i=1}^{m}\frac{n_i(n_i-1)\cdots(n_i-r+1)}{n_i^r}\cdot\frac{n_i^r}{n^r}\\ &= \sum_{i=1}^{m}\frac{n_i(n_i-1)\cdots(n_i-r+1)}{n^r}\end{aligned}$$

Since $p_r = \Pr[E]$, the proposition follows. □

Definition 3.1. Let $h : X \to Y$ be a hash function with $|X| = n$ and $Y = \{y_1, y_2, \cdots, y_m\}$. Let $n \geq r$ and $p_r > 0$. The *r-balance of h*, denoted $\mu_r(h)$, is defined as

$$\mu_r(h) = \frac{1}{r-1}\cdot\log_m\left(\frac{1}{p_r}\right). \tag{6}$$

If $n_i < r$ for all i, then there cannot be any r-collisions, that is, $p_r = 0$. A necessary condition for the existence of an r-collision is that $n_i \geq r$ for at least one i. If $n \geq rm$, then an r-collision will certainly exist but there could be an r-collision even if $n < rm$. We only require the condition that $p_r > 0$.

Consider the case $r = 2$. From the definition of $\mu(h)$, we have

$$m^{-\mu_2(h)} = \frac{\sum_{i=1}^{m} n_i(n_i-1)}{n^2} = \frac{\sum_{i=1}^{m} {n_i}^2}{n^2} - \frac{\sum_{i=1}^{m} n_i}{n} = m^{-\mu(h)} - \frac{1}{n}.$$

This shows that $\mu_2(h)$ is always greater than $\mu(h)$. The difference gets smaller as n grows larger.

The following lemma will be useful in obtaining bounds on the r-balance of a hash function.

Lemma 3.2. *Let $r \geq 2$ be an integer. Let $n_1, n_2, \cdots, n_m$ be non-negative integers such that $\sum_{i=1}^{m} n_i = n$. Then*

$$m\cdot\left(\frac{n}{m}\right)_r \leq \sum_{i=1}^{m}(n_i)_r \leq (n)_r.$$

The upper bound is attained when exactly one of the n_i equals n and all others are zero, while the lower bound is attained when all the n_is are equal.

Proof. We will prove the bounds using a counting argument. Let $S(n_i)$ denote the set of all distinct arrangements of n_i things taken r at a time. Then $|S(n_i)| = (n_i)_r$ for $i = 1, \cdots m$. If $n_j \leq r-1$ for some j then $S(n_j) = \emptyset$. Assume, without loss of generality, that the first k of the n_i's are greater than $r-1$. By definition $n = \sum_{i=1}^{m} n_i$. Let S denote the set of all distinct arrangements of n things taken r at a time. Each arrangement in $S(n_i)$ is also present in S. This show that $S(n_1) \cup S(n_2) \cup \cdots \cup S(n_k) \subseteq S$. Also since the $S(n_i)$'s are disjoint, we have

$$(n_1)_r + (n_2)_r + \cdots + (n_k)_r \leq (n_1 + n_2 + \cdots + n_k)_r = (n)_r$$

Equality occurs when $k = 0$ i.e., one of the n_i's is equal to n and the rest are zero. This gives the upper bound on $\sum_{i=1}^{m}(n_i)_r$.

Now we claim that $\sum_{i=1}^{m}(n_i)_r$ attains its minimum when all n_i's are equal i.e., $n_1 = n_2 = \cdots = n_m = \frac{n}{m}$. Suppose there exist n_i and n_j such that $n_i > \frac{n}{m}$ and $n_j < \frac{n}{m}$. Assume, without loss of generality, that $i = 1$ and $j = 2$. To prove the claim, we need but show that

$$(n_1 - 1)_r + (n_2 + 1)_r + \cdots + (n_k)_r < (n_1)_r + (n_2)_r + \cdots + (n_k)_r.$$

Let T_i denote the set containing n_i items. Clearly, $T_1 \cup T_2 \cup \cdots \cup T_m = X$. Let $x \in T_1$. The number of arrangements of items in T_1 taken r at a time that contain x is equal to $r(n_1-1)_{r-1}$. Suppose we remove x from T_1 and put it in T_2. Then the number of arrangements of items in T_2 taken r at a time that contain x is equal to $r(n_2)_{r-1}$. Thus we have

$$\begin{aligned}
&((n_1)_r + (n_2)_r + \cdots + (n_k)_r) - ((n_1-1)_r + (n_2+1)_r + \cdots + (n_k)_r) \\
&= |S(n_1) \cup S(n_2) \cup \cdots \cup S(n_m)| - |S(n_1-1) \cup S(n_2+1) \cup \cdots \cup S(n_m)| \\
&= |S(n_1) \cup S(n_2)| - |S(n_1-1) \cup S(n_2+1)| \\
&= |S(n_1-1)| + r(n_1-1)_{r-1} + |S(n_2)| - |S(n_1-1)| - |S(n_2)| - r(n_2)_{r-1} \\
&= r(n_1-1)_{r-1} - r(n_2)_{r-1} \\
&> 0
\end{aligned}$$

since $n_1 - 1 > n_2$. This gives the lower bound. □

The following proposition provides the minimum and maximum values of the r-balance of a function and the conditions under which they are attained. The proof follows directly from the definition of $\mu_r(h)$ and Proposition 3.2.

Proposition 3.3. *Let h be an (n,m)-hash function. Then*

$$\frac{1}{r-1}\log_m \frac{n^r}{(n)_r} \leq \mu_r(h) \leq \frac{1}{r-1}\log_m \frac{n^r}{m \cdot \left(\frac{n}{m}\right)_r} \tag{7}$$

The lower bound is attained when h is a constant function and the upper bound is attained when h is a regular function.

Let $\mu_r^{\min}(n,m)$ and $\mu_r^{\max}(n,m)$ denote the minimum and maximum values of the r-balance of an (n,m)-hash function. The quantity $\mu_r^{\min}(n,m)$ can be approximated as follows.

$$\begin{aligned}
\mu_r^{\min}(n,m) = \frac{1}{r-1}\log_m \frac{n^r}{(n)_r} &= \frac{1}{r-1}\log_m \frac{1}{\left(1-\frac{1}{n}\right)\cdots\left(1-\frac{r-1}{n}\right)} \\
&\approx \frac{1}{r-1}\log_m \frac{1}{e^{-1/n}\cdots e^{-(r-1)/n}} = \frac{1}{r-1}\log_m \frac{1}{e^{-\binom{r}{2}/n}} = \frac{r}{2n(\ln m)}.
\end{aligned}$$

This shows that, for large n, the $\mu_r^{\min}(n,m)$ is close to zero. Similarly one can approximate $\mu_r^{\max}(n,m)$ as follows.

$$\begin{aligned}
\mu_r^{\max}(n,m) = \frac{1}{r-1}\log_m \frac{n^r}{m \cdot \left(\frac{n}{m}\right)_r} &= \frac{1}{r-1}\log_m \frac{m^{r-1}}{\left(1-\frac{m}{n}\right)\cdots\left(1-\frac{(r-1)m}{n}\right)} \\
&\approx \frac{1}{r-1}\log_m \frac{m^{r-1}}{e^{-m/n}\cdots e^{-(r-1)m/n}} \\
&= \frac{1}{r-1}\log_m \left(m^{r-1} e^{\binom{r}{2}m/n}\right) = 1 + \frac{rm}{2n(\ln m)}.
\end{aligned}$$

This shows that for large n, $\mu_r^{\max}(n,m)$ is close to one.

3.3 Bounds on $C_h^{(r)}(q)$

For $I \in [q]_r$, $I = \{i_1, i_2, \cdots, i_r\}$, define a random variable Z_I as follows.

$$Z_I = \begin{cases} 1 & \text{if } x_{i_1}, x_{i_2}, \cdots, x_{i_r} \text{ form an } r\text{-collision} \\ 0 & \text{otherwise} \end{cases}$$

From Proposition 3.1 and the definition of r-balance we have

$$\mathbf{E}[Z_I] = \Pr[Z_I = 1] = \frac{\sum_{i=1}^{m} (n_i)_r}{n^r} = m^{-(r-1)\mu_r(h)} = p_r \tag{8}$$

Then $Z = \sum_{I \in [q]_r} Z_I$ denotes the number of r-collisions. The expected value of Z is then $\binom{q}{r} m^{-(r-1)\mu_r(h)}$. We are interested in an r-collision and would like to know the number of queries required to have the expected value of Z to be equal to 1. This is given by the value of q such that $(q)_r = r! \times m^{(r-1)\mu_r(h)}$. Using the inequality $(q - r)^r < (q)_r$, it can be easily shown that choosing $q = r + (r!)^{1/r} \times m^{(r-1)\mu_r(h)/r}$ ensures that $\mathbf{E}[Z] \geq 1$. This gives an indication of the "right" value of q required to obtain an r-collision.

We now consider that q trials are made and obtain bounds on $C_h^{(r)}(q)$. An upper bound on $C_h^{(r)}(q)$ is easy to obtain.

Theorem 3.4 (Upper Bound on $C_h^{(r)}(q)$)**.** *Let h be an (n, m)-hash function with $n \geq r$ and $m \geq 2$. Then for any integer $q \geq r$,*

$$C_h^{(r)}(q) \leq \binom{q}{r} p_r. \tag{9}$$

Proof. Let $\{i_1, \ldots, i_r\} \subseteq [q]$. The probability that $x_{i_1}, \ldots, x_{i_r}$ forms an r-collision is p_r. The result now follows from the union bound on probability. $\square$

To obtain a lower bound on $C_h^{(r)}(q)$, we need the following lemma.

Lemma 3.5. *Let h be an (n, m)-hash function and ℓ be an integer such that $\ell > r$. Then*

$$\left(\sum_{i=1}^{m} (n_i)_\ell \right)^r \leq \left(\sum_{i=1}^{m} (n_i)_r \right)^\ell. \tag{10}$$

As a consequence, $p_\ell \leq p_r^{\ell/r}$.

Proof. Without loss of generality assume that $n_1 \geq n_2 \geq \cdots \geq n_m$. Let $A_i = (n_i)_\ell$, $B_i = (n_i)_r$ and $C_i = (n_i - r) \cdots (n_i - \ell + 1)$, so that $A_i = B_i C_i$. We are required to show

$$(B_1 C_1 + \cdots + B_m C_m)^r \leq (B_1 + \cdots + B_m)^\ell. \tag{11}$$

Consider the multinomial expansion of the left hand side of this equation. A term of this expansion is of the form

$$\binom{r}{d_1\ d_2 \cdots d_m} (B_1 C_1)^{d_1} (B_2 C_2)^{d_2} \cdots (B_m C_m)^{d_m}$$

where $d_1 + \cdots + d_m = r$. We show that this term is less than or equal to

$$\binom{\ell}{d_1 + (\ell - r)\ d_2 \cdots d_m} B_1^{d_1 + (\ell - r)} B_2^{d_2} \cdots B_m^{d_m}$$

which (since $\ell > r$) is a term in the multinomial expansion of the right hand side of (11). This inequality is shown by separately proving the following two inequalitites.

1. $\binom{r}{d_1\ d_2 \cdots d_m} \leq \binom{\ell}{d_1+(\ell-r)\ d_2 \cdots d_m}$.

2. $(B_1C_1)^{d_1}(B_2C_2)^{d_2}\cdots(B_mC_m)^{d_m} \leq B_1^{d_1+(\ell-r)}B_2^{d_2}\cdots B_m^{d_m}$.

Point (1) holds if $\dfrac{r!}{d_1!} \leq \dfrac{\ell!}{(d_1+\ell-r)!}$, i.e., if

$$\frac{\ell(\ell-1)\cdots(r+1)}{(d_1+\ell-r)(d_1+\ell-r-1)\cdots(d_1+1)} \geq 1.$$

This inequality holds if for $1 \leq j \leq \ell-r$, $r+j \geq d_1+j$ which clearly holds since $d_1 \leq r$.

Now consider the second point, which holds if $C_1^{d_1}C_2^{d_2}\cdots C_m^{d_m} \leq B_1^{\ell-r}$. For $1 \leq j \leq \ell-r$, let $E_j = (n_1-r-j-1)^{d_1}\cdots(n_m-r-j-1)^{d_m}$. Then, it follows that

$$C_1^{d_1}C_2^{d_2}\cdots C_m^{d_m} = E_1E_2\cdots E_{\ell-r}.$$

Point (2) now follows if for each $1 \leq j \leq \ell-r$, $E_j \leq B_1$. This follows on noting that $B_1 = n_1(n_1-1)\cdots(n_1-r+1)$ and that by assumption $n_1 \geq n_i$ for $1 \leq i \leq m$. This completes the proof of (10).

By definition,

$$p_\ell = \frac{\sum_{i=1}^m (n_i)_\ell}{n^\ell} \leq \frac{\left(\sum_{i=1}^m (n_i)_r\right)^{\ell/r}}{n^\ell} = \left(\frac{\sum_{i=1}^m (n_i)_r}{n^r}\right)^{\ell/r} = p_r^{\ell/r}.$$

This completes the proof. ∎

Theorem 3.6 (Lower Bound on $C_h^{(r)}(q)$)**.** *Let h be an (n,m)-hash function with $n \geq r$ and $m \geq 2$. Then*

$$C_h^{(r)}(q) \geq \frac{1}{2}\left(2 - \sum_{k=0}^{r-1}\binom{r}{k}\binom{q-r}{r-k}p_r{}^{(r-k)/r}\right)\cdot\binom{q}{r}\cdot p_r. \tag{12}$$

Proof. Let $[q]_{r,2}$ denote the set of all 2-element subsets of $[q]_r$. By the principle of inclusion and exclusion, we have

$$\begin{aligned} C_h^{(r)}(q) &= \Pr\left[\bigvee_{I\in[q]_r} Z_I = 1\right] & (13)\\ &= \sum_{I\in[q]_r}\Pr[Z_I=1] - \sum_{\substack{I,J\in[q]_r\\ I\neq J}}\Pr[Z_I=1 \wedge Z_J=1] \\ &\quad + \cdots + (-1)^{\binom{q}{r}-1}\Pr\left[\bigwedge_{I\in[q]_r} Z_I=1\right] & (14)\end{aligned}$$

Considering the first two terms in the above equation will give us a lower bound on $C_h^{(r)}(q)$.

$$C_h^{(r)}(q) \geq \sum_{I\in[q]_r}\Pr[Z_I=1] - \sum_{\{I,J\}\in[q]_{r,2}}\Pr[Z_I=1 \wedge Z_J=1] \tag{15}$$

From Theorem 3.4, it follows that $\sum_{I\in[q]_r}\Pr[Z_I=1] \leq \binom{q}{r}p_r$.

$$\sum_{I \in [q]_r} \Pr[Z_I = 1] = \binom{q}{r} \Pr[Z_I = 1] = \binom{q}{r} \cdot p_r \tag{16}$$

In order to obtain the required lower bound, we need to maximize the second term of Equation (15). This is where our proof deviates from the one given in [BK04].

For $k = 0, 1, \cdots, r-1$, let N_k be the number of pairs $\{I, J\} \in [q]_{r,2}$ such that $|I \cap J| = k$. The k common elements can be chosen in $\binom{q}{k}$ ways. The remaining $r-k$ elements in I can be chosen in $\binom{q-k}{r-k}$ ways and for each such I, we can choose the remaining $r-k$ elements in J in $\binom{q-r}{r-k}$ ways. But this way we are counting every unordered pair twice (i.e., $\{I, J\}$ and $\{J, I\}$ are indistinguishable but both are counted). Therefore, we have

$$N_k = \frac{1}{2}\binom{q}{k}\binom{q-k}{r-k}\binom{q-r}{r-k} = \frac{1}{2}\binom{q}{r}\binom{r}{k}\binom{q-r}{r-k} \tag{17}$$

We can now break up the second term in Equation (15) as follows:

$$\sum_{\{I,J\} \in [q]_{r,2}} \Pr[Z_I = 1 \wedge Z_J = 1] = \sum_{k=0}^{r-1} N_k \cdot \Pr[Z_I = 1 \wedge Z_J = 1 \mid (|I \cap J| = k)] \tag{18}$$

Let x_I and x_J denote the set of points corresponding to the index sets I and J respectively. Since these points are sampled independently and uniformly at random from X, x_I and x_J may not be disjoint when I and J are disjoint. Whether or not I forms an r-collision is independent of the value Z_J takes. Hence when $k = 0$, the random variables Z_I and Z_J are independent and so we have

$$\Pr[Z_I = 1 \wedge Z_J = 1 \mid (|I \cap J| = 0)] = \Pr[Z_I = 1] \cdot \Pr[Z_J = 1] = {p_r}^2 \tag{19}$$

When $k \geq 1$, the events $Z_I = 1$ and $Z_J = 1$ indicate that the elements in I map to a common point and so do the elements in J. Since $I \cap J \neq \emptyset$, the common image of the elements of both I and J must be the same. Hence $\Pr[Z_I = 1 \wedge Z_J = 1 | (|I \cap J| = k)]$ is the probability that the $2r-k$ distinct elements in $I \cup J$ form a $2r-k$-collision. That is,

$$\Pr[Z_I = 1 \wedge Z_J = 1] = p_{2r-k} \tag{20}$$

Combining Equations (17), (18), (19) and (20), we obtain the following:

$$\Pr[Z_I = 1 \wedge Z_J = 1] = N_0 \cdot {p_r}^2 + \sum_{k=1}^{r-1} N_k \cdot p_{2r-k} \tag{21}$$

To obtain an upper bound on the above expression, we need an upper bound on p_{2r-k}. From Lemma 3.5, we have

$$p_{2r-k} \leq p_r^{(2r-k)/r} = p_r p_r^{(r-k)/r}. \tag{22}$$

Combining Equations (17), (21) and (22), we obtain

$$\sum_{\{I,J\} \in [q]_{r,2}} \Pr[Z_I = 1 \wedge Z_J = 1] \leq \frac{1}{2}\binom{q}{r} \cdot p_r \cdot \sum_{k=0}^{r-1} \binom{r}{k}\binom{q-r}{r-k} {p_r}^{(r-k)/r} \tag{23}$$

Combining Equations (15), (16) and (23), we obtain the lower bound stated in Equation (12) as follows.

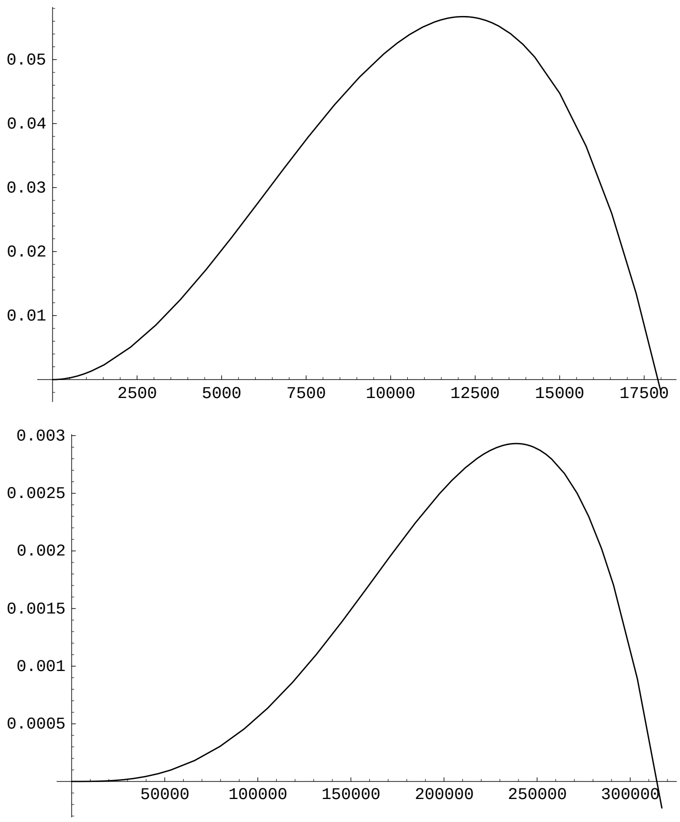

Figure 1: Behaviour of $L_h^{(r)}(q)$ for $r = 2$ and $r = 3$ with $m = 2^{32}$ and $\mu_r = 0.9$.

$$\begin{aligned} C_h^{(r)}(q) &\geq \binom{q}{r} \cdot p_r - \frac{1}{2}\binom{q}{r} \cdot p_r \cdot \sum_{k=0}^{r-1} \binom{r}{k}\binom{q-r}{r-k} {p_r}^{(r-k)/r} \\ &= \frac{1}{2}\binom{q}{r} \cdot p_r \cdot \left(2 - \sum_{k=0}^{r-1} \binom{r}{k}\binom{q-r}{r-k} {p_r}^{(r-k)/r}\right) \end{aligned}$$

$\square$

Towards a better lower bound. We now discuss the behaviour of this lower bound. Let

$$s(q) = 2 - \sum_{k=0}^{r-1} \binom{r}{k}\binom{q-r}{r-k} {p_r}^{(r-k)/r}.$$

and let the lower bound of Theorem 3.6 be denoted $L_h^{(r)}(q)$. We have

$$L_h^{(r)}(q) = \frac{1}{2} \cdot p_r \binom{q}{r} s(q).$$

$L_h^{(r)}(q)$ is a polynomial in q of degree $2r$. One can make the following observations about this polynomial.

Table 1: h is a hash function with $n = 2^{512}$, $m = 2^{160}$ and $\mu_2(h) = 0.8$.

r	$qmax_r(h)$	Lower bound on $C_h^{(r)}(q)$	Upper bound on $C_h^{(r)}(q)$
3	1.9327×10^{25}	2.9312×10^{-3}	1.0391×10^{-2}
4	2.4385×10^{28}	8.6908×10^{-5}	3.7393×10^{-4}
5	1.6855×10^{30}	1.6724×10^{-6}	8.4565×10^{-6}
6	2.7482×10^{31}	2.2583×10^{-8}	1.3117×10^{-7}
7	1.9718×10^{32}	2.2585×10^{-10}	1.4813×10^{-9}
8	8.4939×10^{32}	1.7403×10^{-12}	1.2719×10^{-11}
9	2.6088×10^{33}	1.065×10^{-14}	8.5817×10^{-14}
10	6.3321×10^{33}	5.3018×10^{-17}	4.6689×10^{-16}

- The binomial coefficient $\binom{q}{r} = \frac{1}{r!}q(q-1)\cdots(q-(r-1))$ and it vanishes at the points $0, 1, \cdots, r-1$ which means these are roots of $L_h^{(r)}(q)$. It is also monotone increasing and positive for $q \geq r$.
- $s(q)$ is decreasing in q and becomes negative after a certain point causing $L_h^{(r)}(q)$ to decrease.
- The polynomial $s(q)$ has exactly one sign change and by Descartes' rule of signs it will have at most one positive real root.

These observations show that $L_h^{(r)}(q)$ has exactly $r+1$ non-negative real roots including $0, 1, \cdots, r-1$. This is because $L_h^{(r)}(q)$ is positive at $q = r$ and after a certain point becomes negative which means it is zero at exactly one point after $r-1$. Let the $(r+1)^{st}$ real root be denoted as θ. In the interval ranging from $q = r$ to $q = \theta$, the curve representing $L_h^{(r)}(q)$ must have one turning point. Figure 1 gives some examples to show how $L_h^{(r)}(q)$ behaves. Let the value of q at which the curve turns be denoted $\mathsf{qmax}_r(h)$ and let $\mathsf{cmax}_r(h) = L_h^{(r)}(\mathsf{qmax}_r(h))$. For $q \leq \mathsf{qmax}_r(h)$ the lower bound will be $L_h^{(r)}(q)$. For $q > \mathsf{qmax}_r(h)$, $L_h^{(r)}(q)$ is decreasing but the probability of finding r-collisions cannot decrease as we increase the number of trials. Hence $L_h^{(r)}(\mathsf{qmax}_r(h))$ is a better lower bound. Based on this discussion and Theorems 3.4 and 3.6, we are able to state more appropriate bounds on $C_h^{(r)}(q)$.

Theorem 3.7. *Let h be an (n, m)-hash function. Then*

$$\max_{r \leq t \leq q} L_h^{(r)}(t) \leq C_h^{(r)}(q) \leq \binom{q}{r} \cdot p_r \tag{24}$$

Note. Theorem 3.7 is valid for all q (and for all $r \geq 2$). This is to be contrasted with the bound obtained in [BK04] for the case $r = 2$ (see Theorem 2.1).

How close are the bounds? Since $L_h^{(r)}(q)$ is difficult to analyse, we provide computational results to show how close the bounds are. Table 1 provides lower and upper bounds on $C_h^{(r)}(q)$ for different values of r and a fixed h. Both the bounds are evaluated at $qmax_r(h)$. The values indicate that the upper bound may be roughly ten times the lower bound for $q \geq qmax_r(h)$. For values of $q \geq qmax_r(h)$, the gap between the bounds increases.

Further simplifications. The lower bound stated in Theorem 3.7 can be further simplified as shown below.

Corollary 3.8. *Let h be an (n,m)-hash function. Assume $n \geq r \geq 2$. Let*

$$\alpha(q) = qm^{-\left(\frac{r-1}{r}\right)\mu_r(h)}. \tag{25}$$

Then

$$C_h^{(r)}(q) \geq \max_{r \leq t \leq q} \frac{1}{2}\ \left(3 - (\alpha(t)+1)^r\right) \cdot \binom{t}{r} \cdot m^{-(r-1)\mu_r(h)}. \tag{26}$$

Proof. We proceed as in the proof of Theorem 3.6 upto Equation (23). It is after this point that the proof will deviate. Using Equations (15), (16) and (23) we get

$$\begin{aligned}
C_h^{(r)}(q) &\geq \binom{q}{r} \cdot p_r - \frac{1}{2}\binom{q}{r} \cdot p_r \cdot \sum_{k=0}^{r-1} \binom{r}{k}\binom{q-r}{r-k} {p_r}^{(r-k)/r} \\
&= \frac{1}{2}\binom{q}{r} \cdot p_r \cdot \left(2 - \sum_{k=0}^{r-1} \binom{r}{k}\binom{q-r}{r-k} {p_r}^{(r-k)/r}\right) \\
&\geq \frac{1}{2}\binom{q}{r} \cdot p_r \cdot \left(2 - \sum_{k=0}^{r-1} \binom{r}{k} q^{r-k} {p_r}^{(r-k)/r}\right) \\
&= \frac{1}{2}\binom{q}{r} \cdot p_r \cdot \left(2 - \sum_{k=0}^{r-1} \binom{r}{k} (\alpha(q))^{r-k}\right) \\
&= \frac{1}{2}\binom{q}{r} \cdot p_r \cdot \left(2 - ((\alpha(q)+1)^r - 1)\right) \\
&= \frac{1}{2}\binom{q}{r} \cdot p_r \cdot \left(3 - (\alpha(q)+1)^r\right)
\end{aligned}$$

Using the same arguments that led to Theorem 3.7, we get the bound stated in Equation (26). □

This simplification actually weakens the bound since q^{r-k} is a weak upper bound on $\binom{q-r}{r-k}$ but can make it easier to work with the expressions. Nonetheless we prefer to work with Theorem 3.7.

3.4 Bounds on $Q_h^{(r)}(c)$

Now we obtain upper and lower bounds on $Q_h^{(r)}(c)$. These bounds can be directly obtained from the bounds on $C_h^{(r)}(q)$.

Theorem 3.9. *Let h be an (n,m)-hash function with $n \geq r$ and $m \geq 2$. Let c be a real number such that $0 \leq c < 1$. Then*

$$c^{1/r}\left(\frac{r}{e}\right) m^{\left(\frac{r-1}{r}\right)\mu_r(h)} \leq Q_h^{(r)}(c) \leq \min\{q : L_h^{(r)}(q) = c\}, \tag{27}$$

the upper bound being true when

$$c < \mathsf{cmax}_r(h). \tag{28}$$

Table 2: h is a hash function with $m = 2^{80}$ and $\mu_r(h) = 0.9$.

r	$\mathsf{cmax}_r(h)$
2	5.67003×10^{-2}
3	2.93125×10^{-3}
4	8.69089×10^{-5}
5	1.67242×10^{-6}
6	2.25836×10^{-8}
7	2.25855×10^{-10}
8	1.74035×10^{-12}

Proof. From Theorem 3.7 we have

$$C_h^{(r)}(q) \leq \underbrace{\binom{q}{r} m^{-(r-1)\mu_r(h)}}_{U_h^{(r)}(q)}$$

To get the lower bound of Equation (27) we need to solve for q in the equation $U_h^{(r)}(q) = c$.

$$\begin{aligned} c &= \binom{q}{r} m^{-(r-1)\mu_r(h)} \\ &\leq \left(\frac{qe}{r}\right)^r \frac{1}{m^{(r-1)\mu_r(h)}} \end{aligned}$$

$$q \geq c^{1/r}\left(\frac{r}{e}\right) m^{\left(\frac{r-1}{r}\right)\mu_r(h)}$$

This proves the lower bound of Equation (27).

Similarly solving for q in $L_h^{(r)}(q) = c$ will yield an upper bound on $Q_h^{(r)}(c)$. This is possible only when Equation (28) holds. We have discussed earlier the behaviour of the function $L_h^{(r)}(q)$ from which it follows that, for a given $c > 0$, there are at most two values of q for which $L_h^{(r)}(q) = c$. The smaller of the two q-values is a better upper bound on the number of trials. Thus we have

$$Q_h^{(r)}(c) \leq \min\{q : L_h^{(r)}(q) = c\}.$$

□

Theorem 3.9 states that, for a given hash function h, the number of trials required to obtain an r-collision with a given probability c is at least as much as the lower bound. Also for $c \leq \mathsf{cmax}_r(h)$, the maximum number of trials required to obtain success probability c is given by the the upper bound on $Q_h^{(r)}(c)$. For values of c greater than $\mathsf{cmax}_r(h)$ we are unable to say anything about the maximum number of trials required to attain success probability c. But, the lower bound still continues to hold, i.e., we are still able to say that at least those many queries will be required to attain success probability c.

It would be interesting to know the range of values of c for which the upper bound on $Q_h^{(r)}(c)$ holds for different values of r. Because of the form of $L_h^{(r)}(q)$, we are unable to solve for q in $L_h^{(r)}(q) = c$ or get a closed form expression for $\mathsf{cmax}_r(h)$. Table 2 shows how $\mathsf{cmax}_r(h)$ varies with r when m and $\mu_r(h)$ are

Table 3: $n = 2^{512}$, $m = 2^{160}$ and $c = 0.78$.

$\mu_4(h)$	Lower bound on $Q_h^{(4)}(c)$
0.22	1.22456×10^{8}
0.33	1.15233×10^{12}
0.44	1.08436×10^{16}
0.55	1.0204×10^{20}
0.66	9.60207×10^{23}
0.77	9.03568×10^{27}
0.88	8.5027×10^{31}
0.99	8.00115×10^{35}

fixed. One can observe that the value of $\mathsf{cmax}_r(h)$ is decreasing rapidly with increasing values of r which means that as r grows larger the upper bound of Theorem 3.9 is valid across smaller ranges of c.

Sensitivity of $Q_h^{(r)}(c)$ to r-balance. We now provide some computational results that indicate how the number of trials required by the generic multi-collision attack changes according to the r-balance of the function being attacked. Table 3 shows the lower bound on $Q_h^{(4)}(c)$ for a fixed c and for functions with different values of 4-balance.

The table indicates that for functions with higher 4-balance it is harder to find 4-collisions using the generic multi-collision attack when compared to functions with low 4-balance.

4 Random Functions

Consider a uniform random (n, m)-hash function. We consider the resistance of such a hash function to the generic multi-collision attack. Our aim is to show that the attack works better against uniform random functions compared to regular functions. This is shown by proving that the success probability of the attack is higher for a uniform random function than for a regular function. Informally, one may consider that having higher success probability means that it is easier to find r-collisions.

Note that the generic multi-collision attack as described earlier is not the best attack on a uniform random function. As mentioned in the introduction, for such a function one does not need to apply any sampling technique to choose the domain points. One simply has to pick q distinct domain points. We discuss this issue in more details below.

Let $C_{n,m}^{\$(r)}(q)$ be the probability that the generic multi-collision attack on a uniform random (n, m)-hash function succeeds in q trials. Here the probability is over the choice of the function and the points picked by the attack. Similarly, let $Q_{n,m}^{\$(r)}(c)$ denote the minimum number of trials required to obtain an r-collision with probability greater than or equal to c.

Let $p_r^{\$}$ denote the probability that r elements, chosen independently and uniformly at random from the domain X, form an r-collision. Let r elements $w_1, w_2, \cdots, w_r$ be picked independently and uniformly at random from the domain X. If A is the event that these are distinct and B is the event that $h(w_1) =$

$\cdots = h(w_r)$, then $p_r^\$ = \Pr[A] \cdot \Pr[B]$. Clearly,

$$\Pr[A] = \frac{(n)_r}{n^r} \text{ and } \Pr[B] = m \cdot \frac{1}{m^r}$$

since there are m choices for the common image. Thus we have,

$$p_r^\$ = \frac{(n)_r}{n^r} \cdot \frac{1}{m^{r-1}}.$$

Note. Suppose instead of choosing the points $x_1, \ldots, x_r$ using random sampling with replacement, we simply choose them to be *any* r distinct points. Then the probability that they form an r-collision is $1/m^{r-1}$. Clearly, this probability is greater than $p_r^\$$. By extension, it is not difficult to see that if we simply pick q *distinct* points (instead of sampling them with replacement), then the probability (say $\chi_{n,m}^{(r)}(q)$) of obtaining an r-collision is greater than $C_{n,m}^{\$(r)}(q)$. The main result of this section shows that in fact $C_{n,m}^{\$(r)}(q) > C_h^{(r)}(q)$ for any regular (n,m)-hash function h. Then, it follows that

$$\chi_{n,m}^{(r)}(q) > C_{n,m}^{\$(r)}(q) > C_h^{(r)}(q).$$

In other words, the success probability of the simpler attack is even higher and actually butresses our assertion that random functions offer lesser security compared to regular functions.

In view of the above discussion, in the rest of this section we will only consider the generic multi-collision attack on a uniform random (n,m)-hash function. The bounds on $C_{n,m}^{\$(r)}(q)$ and $Q_{n,m}^{\$(r)}(c)$ are obtained in a manner similar to that for a concrete hash function and we state some of the results without proofs.

Lemma 4.1. *Let ℓ be an integer such that $\ell > r$. Then*

$$p_\ell^\$ \leq (p_r^\$)^{\ell/r}$$

Theorem 4.2. *For a uniform random (n,m)-hash function with $n > r$ the following holds.*

$$\max_{r \leq t \leq q} L_{n,m}^{\$(r)}(t) \leq C_{n,m}^{\$(r)}(q) \leq \binom{q}{r} \cdot p_r^\$ \tag{29}$$

where the function $L_{n,m}^{\$(r)}(t)$ is defined as follows:

$$L_{n,m}^{\$(r)}(t) = \frac{1}{2}\left(2 - \sum_{k=0}^{r-1} \binom{r}{k}\binom{t-r}{r-k}(p_r^\$)^{(r-k)/r}\right) \cdot \binom{t}{r} \cdot p_r^\$ \tag{30}$$

For the purpose of comparison to regular functions we will use a simplified version of the lower bound on $C_{n,m}^{\$(r)}(q)$. This is obtained in a manner similar to the one given in the proof of Corollary 3.8.

Corollary 4.3. *For a uniform random (n,m)-hash function with $n > r$, let*

$$\alpha^\$(q) = qm^{-\left(\frac{r-1}{r}\right)}. \tag{31}$$

Then

$$C_{n,m}^{\$(r)}(q) \geq \max_{r \leq t \leq q} \frac{1}{2}\left(3 - (\alpha^\$(t) + 1)^r\right) \cdot \binom{t}{r} \cdot p_r^\$ \tag{32}$$

We now proceed towards obtaining bounds on $Q^{\$(r)}_{n,m}(c)$. The upper bound can be obtained the same way as in Theorem 3.9. Only a proof of the lower bound is provided here.

Theorem 4.4. *Consider a uniform random (n,m)-hash function with $n > r$ and let c be a real number such that $0 \le c < 1$. Then*

$$c^{1/r} r \cdot e^{\left(\frac{r-1}{2n}-1\right)} \cdot m^{\left(\frac{r-1}{r}\right)} \le Q^{\$(r)}_{n,m}(c) \le \min\{q : L^{\$(r)}_{n,m}(q) = c\}, \tag{33}$$

the upper bound being true when

$$c < \mathsf{cmax}^{\$}_r(n,m). \tag{34}$$

where $\mathsf{cmax}^{\$}_r(n,m)$ denotes the maximum positive value that the function $L^{\$(r)}_{n,m}(q)$ attains.

Proof. From Theorem 4.2 we have

$$C^{\$(r)}_{n,m}(q) \le \underbrace{\binom{q}{r} p^{\$}_r}_{U^{\$(r)}_{n,m}(q)}$$

To get the lower bound of Equation (33) we need to solve for q in the equation $U^{\$(r)}_{n,m}(q) = c$.

$$\begin{aligned} c &= \binom{q}{r} \cdot \left(1 - \frac{1}{n}\right)\left(1 - \frac{2}{n}\right) \cdots \left(1 - \frac{r-1}{n}\right) \cdot \frac{1}{m^{r-1}} \\ &\le \left(\frac{qe}{r}\right)^r e^{-1/n} e^{-2/n} \cdots e^{-(r-1)/n} \cdot \frac{1}{m^{r-1}} \\ &= \left(\frac{qe}{r}\right)^r e^{-r(r-1)/2n} \cdot \frac{1}{m^{r-1}} \end{aligned}$$

Solving for q in the above inequality will give

$$q \ge c^{1/r} r \cdot e^{\left(\frac{r-1}{2n}-1\right)} \cdot m^{\left(\frac{r-1}{r}\right)}$$

□

Comparison with regular functions. Let $C^{\mathrm{reg}(r)}_{n,m}(q)$ denote the probability of success of the generic multi-collision attack on a regular (n,m)-hash function. Let the maximum value of r-balance be denoted $\mu^{\max}_r$ and let the value of p_r corresponding to $\mu^{\max}_r$ be denoted as p^{reg}_r. Since all regular functions have the same value for p_r, we have $C^{\mathrm{reg}(r)}_{n,m}(q) = C^{(r)}_h(q)$ for some function h with maximum balance.

Lemma 4.5. *Let n, m and r be integers such that $r \ge 2$ and $n \ge rm$. Then*

$$\frac{n(n-1)\cdots(n-(r-1))}{n(n-m)\cdots(n-(r-1)m)} > 1 + \frac{m-1}{n-m} \cdot \frac{r(r-1)}{2}$$

Proof. The condition $n \geq rm$ ensures that the denominator $n(n-m)\cdots(n-(r-1)m)$ is non-zero.

$$\begin{aligned}
\frac{n(n-1)\cdots(n-(r-1))}{n(n-m)\cdots(n-(r-1)m)} &= \frac{n-1}{n-m}\cdot\frac{n-2}{n-2m}\cdots\frac{n-(r-1)}{n-(r-1)m} \\
&= \left(1+\frac{m-1}{n-m}\right)\cdot\left(1+\frac{2(m-1)}{n-2m}\right)\cdots\left(1+\frac{(r-1)(m-1)}{n-(r-1)m}\right) \\
&> 1+\frac{m-1}{n-m}+\frac{2(m-1)}{n-2m}+\cdots+\frac{(r-1)(m-1)}{n-(r-1)m} \\
&> 1+\frac{m-1}{n-m}+\frac{2(m-1)}{n-m}+\cdots+\frac{(r-1)(m-1)}{n-m} \\
&= 1+\frac{m-1}{n-m}\cdot\frac{r(r-1)}{2}
\end{aligned}$$

□

Recall the lower bound on $C^{\$(r)}_{n,m}(q)$ stated in Corollary 4.3. Let $\mathsf{qmax}^{\$}_r(n,m)$ denote the value of q at which this lower bound is maximum.

Theorem 4.6. *Let $r \geq 2$ and $n \geq rm$ and*

$$\beta = 1+\frac{m-1}{n-m}\cdot\frac{r(r-1)}{2}$$

Then

$$C^{\$(r)}_{n,m}(q) > C^{\mathrm{reg}(r)}_{n,m}(q) \tag{35}$$

for all q such that $q \leq \min\left(\mathsf{qmax}^{\$}_r(n,m), \left(\left(3-\frac{2}{\beta}\right)^{1/r}-1\right)m^{(r-1)/r}\right)$.

Proof. The lower bound on $C^{\$(r)}_{n,m}(q)$ stated in Corollary 4.3 increases till q reaches $\mathsf{qmax}^{\$}_r(n,m)$ after which it remains constant whereas, the upper bound on $C^{\mathrm{reg}(r)}_{n,m}(q)$ keeps increasing. So the comparison holds only for values of $q \leq \mathsf{qmax}^{\$}_r(n,m)$. From (31) recall that $\alpha^{\$}(q) = qm^{-\left(\frac{r-1}{r}\right)}$ and by the bound given on q, we have $1/\beta \leq (3-(\alpha^{\$}(q)+1)^r)/2$. This will be used in the computation below. Also Lemma 4.5 is used in the last but one step of the computation.

Assume $q \le \mathsf{qmax}_r^{\$}(n,m)$. From Corollary 4.3, we have,

$$\begin{aligned}
C_{n,m}^{\$(r)}(q) &\ge \max_{r \le t \le q} \frac{1}{2}(3-(\alpha^{\$}(t)+1)^r)\binom{t}{r}p_r^{\$} \\
&= \frac{1}{2}(3-(\alpha^{\$}(q)+1)^r)\binom{q}{r}p_r^{\$} \\
&\ge \frac{1}{\beta}\binom{q}{r}\frac{(n)_r}{n^r}\frac{1}{m^{r-1}} \\
&= \frac{1}{\beta}\binom{q}{r}\frac{(n)_r}{n^r}\frac{1}{m^{r-1}} \\
&= \frac{1}{\beta}\binom{q}{r}\frac{(n)_r}{n^r}\frac{1}{m^{r-1}}\frac{n(n-m)\cdots(n-(r-1)m)}{n(n-m)\cdots(n-(r-1)m)} \\
&= \frac{1}{\beta}\frac{(n)_r}{n(n-m)\cdots(n-(r-1)m)}\binom{q}{r}\frac{m\left(\frac{n}{m}\right)_r}{n^r} \\
&> \frac{1}{\beta}\left(1+\frac{m-1}{n-m}\cdot\frac{r(r-1)}{2}\right)\binom{q}{r}p_r^{\mathrm{reg}} \\
&\ge C_{n,m}^{\mathrm{reg}(r)}(q)
\end{aligned}$$

□

Theorem 4.6 shows that for a certain range of q, it is easier to find r-collisions for random functions than for regular functions. So, random functions provide lesser security compared to regular functions. The value of β is greater than 1 and consequently, the value of $(3-2/\beta)$ is also greater than 1 so that the upper bound on q required in Theorem 4.6 is not vacuous.

5 Expected Number of Trials to Obtain an r-Collision

Suppose the domain points are chosen one by one independently and uniformly at random and h is applied to them. The process is continued as long as necessary until an r-collision occurs. We would then like to know the expected number of trials $E_h^{(r)}$ to obtain an r-collision.

For the case of $r=2$, this was analysed by Bellare and Kohno. Given a hash function h, they denoted by E_h the expected number of trials required to obtain a collision (i.e., $E_h = E_h^{(2)}$) and obtained bounds on E_h. These bounds are obtained from two facts of a more general nature. They show that if $q \ge 2$ is the number of trials then $q(1-C_h(q-1)) \le E_h$ and $E_h \le q/C_h(q)$. The arguments used to obtain these bounds also go through for general r.

Proposition 5.1. *For any $q \ge r$,*

$$q(1-C_h^{(r)}(q-1)) \le E_h^{(r)} \le \frac{q}{C_h^{(r)}(q)}.$$

Proof. The ideas involved in the proof are from [BK04]. Let $D_h^{(r)}(q)$ be the probability that the first r-collision is found at trial number q. Then $\sum_{i \ge q} D_h^{(r)}(i)$ is the probability that the first r-collision is found after $(q-1)$ trials which is equal to the probability that the first $(q-1)$ trials do not provide an r-collision. So, $\sum_{i\geq q} D_h^{(r)}(i) = 1 - C_h^{(r)}(q-1)$. Then

$$E_h^{(r)} = \sum_{i\geq 1} iD_h^{(r)}(i) \geq q\sum_{i\geq q} D_h^{(r)}(i) = q(1 - C_h^{(r)}(q-1)).$$

Obtaining the upper bound is a little more involved. Consider the trials to be conducted in batches of q trials each, i.e., trials with $x_{q(i-1)+1}, \ldots, x_{qi}$ are conducted in batch number i. Let $X_i = 1$ if an r-collision is found in batch number i and 0 otherwise. Since the x_js are chosen independently and uniformly at random, the random variables $X_1, X_2, \ldots$ are mutually independent Bernoulli trials with $\Pr[X_i = 1] = C_h^{(r)}(q)$ for all $i \geq 1$. Let Y be a random variable whose value is i if $X_i = 1$ and $X_k = 0$ for $1 \leq k \leq i-1$. Then Y follows the geometric distribution. Denote $C_h^{(r)}(q)$ by ε and then the expected value of qY can be computed as

$$\begin{aligned}\mathbf{E}[qY] &= q\varepsilon + 2q(1-\varepsilon)\varepsilon + \cdots + iq(1-\varepsilon)^{i-1}\varepsilon + \cdots \\ &= q\varepsilon\left(\frac{1}{\varepsilon^2}\right) = \frac{q}{\varepsilon} = \frac{q}{C_h^{(r)}(q)}.\end{aligned}$$

The above process of batching ignores the possibility that an r-collision can occur between the trials of batch number i and the trials of the previous batches. So, batching can only increase the expected number of trials to find an r-collision and hence

$$E_h^{(r)} \leq \mathbf{E}[Y] \leq \frac{q}{C_h^{(r)}(q)}.$$

This completes the proof. □

To obtain more meaningful bounds, we need to evaluate the bounds in Proposition 5.1 for some values of q. A good value of q is $\mathsf{qmax}_r(h)$ which is the point where the function $L_h^{(r)}(q)$ attains its maximum, i.e., the value of q for which the lower bound on $C_h^{(r)}(q)$ attains the maximum value $\mathsf{cmax}_r(h)$. This gives the following bounds.

$$\mathsf{qmax}_r(h)(1 - C_h^{(r)}(\mathsf{qmax}_r(h) - 1)) \leq E_h^{(r)} \leq \frac{\mathsf{qmax}_r(h)}{\mathsf{cmax}_r(h)}.$$

As noted in Section 3.3, it is difficult to obtain a closed form expression for $\mathsf{qmax}_r(h)$, so it is still difficult to understand what the above bounds really mean. Further, these bounds are not in terms of the balance. To get them in terms of the balance, we have to evaluate the bounds for suitable values of q. In fact we evaluate the lower and upper bounds in Proposition 5.1 for different values of q.

Let $Q = m^{((r-1)/r)\mu_r(h)}$. Then from Theorem 3.4

$$C_h^{(r)}(Q-1) \leq \binom{Q-1}{r} p_r \leq \frac{Q^r p_r}{r!} = \frac{1}{r!}.$$

This shows

$$E_h^{(r)} \geq \left(1 - \frac{1}{r!}\right) m^{\frac{(r-1)}{r}\mu_r(h)}. \tag{36}$$

The upper bound involves a little more calculation. From Corollary 3.8 we have that, for $\alpha(q) = qm^{-(\frac{r-1}{r})\mu_r(h)}$,

$$C_h^{(r)}(q) \geq \frac{1}{2}\left(3 - (\alpha(q)+1)^r\right) \cdot \binom{q}{r} \cdot m^{-(r-1)\mu_r(h)}.$$

Put $\alpha(q) = \delta_r$. We specify the exact value of δ_r later.

For $q \geq r$, we have $(1 - (r-1)/q) \geq 1/r$ and so

$$\begin{aligned} \frac{q!}{r!(q-r)!} = \frac{q(q-1)\cdots(q-r+1)}{r!} &= \frac{q^r}{r!}\left(1-\frac{1}{q}\right)\cdots\left(1-\frac{r-1}{q}\right) \\ &\geq \frac{q^r}{r!}\left(1-\frac{r-1}{q}\right)^{r-1} \geq \frac{q^r}{r!r^{r-1}}. \end{aligned}$$

Putting $q = \delta_r m^{((r-1)/r)\mu_r(h)} = \delta_r Q$, we have

$$\begin{aligned} C_h^{(r)}(\delta_r Q) &\geq \frac{1}{2}(3-(\delta_r+1)^r)\frac{\delta_r^r Q^r m^{-(r-1)\mu_r(h)}}{r!r^{r-1}} \\ &= \frac{1}{2}\frac{(3-(\delta_r+1)^r)\delta_r^r}{r!r^{r-1}}. \end{aligned}$$

Proposition 5.1 now shows that

$$\begin{aligned} E_h^{(r)} \leq \frac{\delta_r Q}{C_h^{(r)}(\delta_r Q)} &\leq \frac{2r!r^{r-1}}{\delta_r^r(3-(\delta_r+1)^r)} \times \delta_r m^{\frac{(r-1)}{r}\mu_r(h)} \\ &= \frac{2r!r^{r-1}}{\delta_r^{r-1}(3-(\delta_r+1)^r)} \times m^{\frac{(r-1)}{r}\mu_r(h)}. \end{aligned}$$

The value of δ_r is chosen such that it maximizes $x^{r-1}(3-(x+1)^r)$. This in turn, minimizes the upper bound. Differentiating $x^{r-1}(3-(x+1)^r)$ with respect to x and setting to zero, we obtain $x^{r-1}((2r-1)x+r-1)-3(r-1) = 0$. (The solution $x = 0$ has been ruled out.) The polynomial $x^{r-1}((2r-1)x+r-1)-3(r-1)$ has exactly one sign change and by Descartes' rule of signs has exactly one positive real root. We let δ_r to be the value of this root. Combining the two bounds leads to the following result.

Proposition 5.2. *Let h be an (n,m) hash function and δ_r be the positive real root of the polynomial $x^{r-1}((2r-1)x+r-1)-3(r-1)$. Then*

$$\left(1-\frac{1}{r!}\right) m^{\frac{(r-1)}{r}\mu_r(h)} \leq E_h^{(r)} \leq \frac{2r!r^{r-1}}{\delta_r^{r-1}(3-(\delta_r+1)^r)} \times m^{\frac{(r-1)}{r}\mu_r(h)}.$$

For $r = 2$, δ_2 is the positive real root of $3x^2 + 4x - 2 = 0$ and so $\delta_2 = (\sqrt{5}-2)/3$. Using this, we obtain

$$\frac{1}{2} \cdot m^{\mu_2(h)/2} \leq E_h^{(2)} \leq 56 \cdot m^{\mu_2(h)/2}.$$

Recall that $m^{-\mu_2(h)} = m^{-\mu(h)} - 1/n$. This can be used to translated bounds obtained in terms of $\mu(h)$ into bounds in terms of $\mu_2(h)$. For the sake of comparison, we do this for the bounds on $E_h = E_h^{(2)}$ obtained in [BK04].

$$\frac{1}{2} \cdot \sqrt{\frac{n}{n+m^{\mu_2(h)}}} \times m^{\mu_2(h)/2} \leq E_h^{(2)} \leq 72 \cdot \sqrt{\frac{n}{n+m^{\mu_2(h)}}} \times m^{\mu_2(h)/2}.$$

Clearly, the bound that we obtain is better.

6 Conclusion

We have introduced the notion of r-balance of a concrete hash function h. This notion is used to quantify the resistance of h to generic multi-collision attack. Bounds are obtained on the success probability of finding r-collisions using q trials. These are then translated into bounds on the number of trials required for a desired success probability. A similar analysis for uniform random function shows that such functions offer less resistance compared to regular functions.

The work in this paper extended earlier work by Bellare and Kohno [BK04] for collisions, i.e., for $r = 2$ to any $r \geq 2$. To a certain extent, we complete the work started by them.

References

[BK04] M. Bellare and T. Kohno. Hash function balance and its impact on birthday attacks. In C. Cachin and J. Camanisch, editors, *Advances in Cryptology - EUROCRYPT '04*, volume 3027 of *Lecture Notes in Computer Science*. Springer-Verlag, 2004.

[BPVY00] E. Brickell, D. Pointcheval, S. Vaudenay, and M. Yung. Design validation for discrete logarithm based signature schemes. In *PKC'2000*, volume 1751 of *Lecture Notes in Computer Science*, pages 276–292. Springer-Verlag, 2000.

[GS94] M. Girault and J. Stern. On the length of cryptographic hash-values used in identification schemes. In *Advances in Cryptology - CRYPTO 1994*, volume 839 of *Lecture Notes in Computer Science*, pages 202–215. Springer-Verlag, 1994.

[HS06] J. J. Hoch and A. Shamir. Breaking the ICE - finding multicollisions in iterated concatenated and expanded (ICE) hash functions. In *Fast Software Encryption 2006*, volume 4047 of *Lecture Notes in Computer Science*, pages 179–194, Berlin, Germany, 2006. Springer-Verlag.

[JL09] A. Joux and S. Lucks. Improved generic algorithms for 3-collisions. Cryptology ePrint Archive, Report 2009/305, 2009. `http://eprint.iacr.org/`.

[Jou04] A. Joux. Multicollisions in iterated hash functions. application to cascaded constructions. In *Advances in Cryptology - CRYPTO 2004*, volume 3152 of *Lecture Notes in Computer Science*, pages 474–490, Berlin, Germany, 2004. Springer-Verlag.

[Lev81] B. Levin. A representation for multinomial cumulative distribution functions. *The Annals of Statistics*, 9(5):1123–1126, September 1981.

[McK66] E. H. McKinney. Generalized birthday problem. *The American Mathematical Monthly*, 73(4):385–387, April 1966.

[NS07] M. Nandi and D. R. Stinson. Multicollision attacks on some generalized sequential hash functions. *IEEE transactions on Information Theory*, 53(2):759–767, February 2007.

[Pre93] B. Preneel. *Analysis and Design of Cryptograhic Hash Functions.* PhD thesis, Katholieke Universiteit Leuven, Leuven, Belgium, 1993.

[RS96] R. Rivest and A. Shamir. PayWord and MicroMint - two simple micropayment schemes. *CryptoBytes*, 2(1):7–11, Spring 1996.