Double-block Hash-then-Sum: A Paradigm for Constructing BBB Secure PRF

Nilanjan Datta[1], Avijit Dutta[2], Mridul Nandi[2], Goutam Paul[2]

[1] Indian Institute of Technology, Kharagpur
[2] Indian Statistical Institute, Kolkata.
nilanjan_isi_jrf@yahoo.com,avirocks.dutta13@gmail.com,mridul.nandi@gmail.com,
goutam.paul@isical.ac.in

Abstract. SUM-ECBC (Yasuda, CT-RSA 2010) is the first beyond birthday bound (BBB) secure block cipher based deterministic MAC. After this work, some more BBB secure deterministic MACs have been proposed, namely PMAC_Plus (Yasuda, CRYPTO 2011), 3kf9 (Zhang et al., ASIACRYPT 2012) and LightMAC_Plus (Naito, ASIACRYPT 2017). In this paper, we have abstracted out the inherent design principle of all these BBB secure MACs and present a generic design paradigm to construct a BBB secure pseudo random function, namely **Double-block Hash-then-Sum** or in short (DbHtS). A DbHtS construction, as the name implies, computes a *double block hash* on the message and then *sum* the encrypted output of the two hash blocks. Our result renders that if the underlying hash function meets certain security requirements (namely cover-free and block-wise universal advantage is low), DbHtS construction provides $2n/3$-bit security. We demonstrate the applicability of our result by instantiating all the existing beyond birthday secure deterministic MACs (e.g., SUM-ECBC, PMAC_Plus, 3kf9, LightMAC_Plus) as well as a simple two-keyed variant for each of them and some algebraic hash based constructions.

Keywords: DbHtS · Beyond Birthday · Cover-free · Block-wise Universal · PRF · Sum of PRP.

1 Introduction

Pseudo Random Function or in short PRF plays an important role in symmetric key cryptography in providing solutions for authentication and encryption of any arbitrary length message. Mostly, PRFs are realized by iterating a block cipher or a fixed length compression function in a specific mode of operation. These PRFs are called *block cipher based PRF* or *compression function based PRF* respectively. Some of the commonly used block cipher based PRFs are CBC-MAC [BKR00], PMAC [BR02], OMAC [IK03], LightMAC [LPTY16] etc. and compression function based PRFs include NI-MAC [AB99], NMAC [BCK96] etc. These PRFs are secure only up to the birthday bound, i.e., the mode is secure only when the total number of blocks that the mode can process does not exceed $2^{n/2}$, where n is the block size of the underlying primitive (i.e., a block cipher for a block cipher based PRF, a compression function for a compression function based PRF etc.) of the construction. The bound $2^{n/2}$ is called the **birthday bound** in cryptography.

1.1 Limitations of Birthday Bound Secure PRFs

Birthday bound secure constructions are acceptable in practice, if one uses any of these constructions with a moderately large block size. For example, PMAC instantiated with AES-128 permits roughly about 2^{48} queries (using $5\ell q^2/2^n$ [NM08] bound), when the

longest message size is 2^{16} blocks and the success probability of breaking the scheme is restricted to 2^{-10}. However, the same construction becomes vulnerable to use if instantiated with some light weight (smaller block size) block ciphers, whose number has grown tremendously in recent years, e.g., PRESENT [BKL+07], GIFT [BPP+17], LED [GPPR12] etc. For example, PMAC, when instantiated with the PRESENT block cipher (a 64 bit block cipher), permits only about 2^{16} queries when the longest message size is 2^{16} blocks and the success probability of breaking the scheme is 2^{-10}. Therefore, it becomes risky to use the birthday bound secure constructions instantiated with light weight block ciphers. In practice 64-bit block ciphers are still widely used primarily due to legacy applications with backward compatibility e.g., financial sectors, web browsers etc uses triple DES instead of AES as using the latter one in corporate mainframe computers is more expensive. However, if the mode provides only birthday bound security, then 64-bit block cipher does not give adequate security. Having a beyond birthday secure mode solves the issue.

Many practical secure applications use standard AES. Using AES in a birthday secure mode provides 64-bit security which is adequate enough in current days technology. However, due to the technological advancement 64-bit security may not be adequate in future. In such situation, the better option would be to use a mode with beyond birthday security instead of replacing the cipher with larger block size. Note that, there are no standard block cipher of size higher than 128 bits.

1.2 Beyond Birthday Bound Constructions

In this line of research, Yasuda [Yas10] first proposed a BBB secure deterministic MAC, called SUM-ECBC, a rate-1/2 sequential mode of construction with four block cipher keys that offers roughly about $2n/3$-bit security. Followed by this work, Yasuda [Yas11] came up with another deterministic MAC, called PMAC_Plus that also offers roughly about $2n/3$-bit security. Unlike SUM-ECBC, PMAC_Plus is a rate-1 parallel mode of construction with three block cipher keys. Zhang et al. [ZWSW12] proposed another candidate of BBB secure deterministic MAC, called 3kf9, a rate-1 sequential mode of construction with three block cipher keys that offers $2n/3$-bit security. In all of these proposals security bound of the construction is some function of q and ℓ, where q is the total number of queries and ℓ is the maximum number of message blocks in any of the q queried messages. LightMAC_Plus, as proposed by Naito [Nai17], is the first deterministic MAC which is proven to have an ℓ independent beyond birthday bound and hence, it effectively offers a better security than that of all the earlier three proposals. In a very recent work, Datta et al. [DDN+17] proposed a single-keyed variant of the PMAC_Plus that offers a better security bound than that of PMAC_Plus. The MAC part of GCM-SIV2 [IM16] also achieves a stronger, beyond the birthday bound (roughly $2n/3$-bit) security. Besides block cipher based BBB secure PRFs, beyond birthday secure compression function based PRFs have also been studied by Yasuda [Yas08] and Dutta et al. [DNP16].

Interestingly, all these existing beyond birthday bound secure deterministic MACs (i.e., SUM-ECBC, PMAC_Plus, 3kf9, LightMAC_Plus) possess a similar structural design, which is a composition of two constituent elements: (i) a double block hash function that outputs a $2n$-bit hash value of the input message and (ii) a finalization phase that generates the final tag by xor-ing the encryption (via two independent block ciphers) of two n-bit hash values. However, all these MACs follow a different way to bound the security. This observation motivates us to come up with a generic design guideline to construct a beyond birthday bound secure PRF that brings all the existing BBB secure MACs under one common roof and enables us to give a unified security proof for all of them.

1.3 Our Contributions

The contributions of this paper are threefold:

1. We introduce a generic design which we call **Double-block Hash-then-Sum** (in short DbHtS) paradigm, a method of designing a beyond birthday bound secure PRF by xor-ing the encryption of the outputs of a double block hash function. Based on the usage of the keys, we call the DbHtS construction three-keyed (resp. two-keyed), if two block cipher keys are (resp. a single block cipher key is) used in the finalization phase along with the hash key. We would like to mention that we consider only the keyed hash functions unlike popular unkeyed hash functions (e.g., SHA-256, RIPEMD etc).

   We show that if the cover-free and the block-wise universal advantage (See Sect. 3.3 for the definition) of the underlying double block hash function is sufficiently low, then the two-keyed DbHtS is secure beyond the birthday bound. We also extend our generic security result from the two-keyed to the three-keyed DbHtS construction.

2. We show the applicability of our security result for the two-keyed DbHtS construction by instantiating the two-keyed variants of poly-hash based construction and existing beyond birthday secure deterministic MACs (i.e., SUM-ECBC, PMAC_Plus, 3kf9, LightMAC_Plus). Using our generic security result for the two-keyed DbHtS construction, we have shown that all the two-keyed variants (i.e., 2K-ECBC_Plus, 2K-PMAC_Plus, 2kf9 and 2K-LightMAC_Plus) achieve beyond birthday bound security. The bounds are given in Table 1.

3. Finally, we apply our generic security result for the three-keyed DbHtS construction to bound the PRF security of SUM-ECBC, PMAC_Plus, 3kf9 and LightMAC_Plus. Our approach not only provides a generic tool to achieve the BBB security of these constructions, but also helps us to obtain an improved bound for some of the constructions (e.g., SUM-ECBC and PMAC_Plus). Note that, a similar improvement in the security bound has also been observed in 1k-PMAC_Plus by Datta et al. [DDN+17]. Additionally, we have identified a flaw in the existing security proof of 3kf9 [ZWSW12] and to the best of our knowledge, this paper provides the first correct security bound of 3kf9. A comparison of the old security bounds of the existing BBB secure MACs with the new one is depicted in Table 1.

Very recently, Leurent et al. [LNS18] have shown attacks on all these constructions with $2^{3n/4}$ query complexity. This raises an interesting future problem to study the tightness of PRF security of these constructions.

Organization. We develop the notations and recall the basic security definitions in Sect. 2. In Sect. 3, we introduce the DbHtS paradigm and prove its PRF security. We instantiate DbHtS with algebraic double block hash function in Sect. 4. Sect. 5 deals with the security analysis of the two-keyed variants of the parallel constructions (i.e, PMAC_Plus and LightMAC_Plus) and provides an alternative security proof for PMAC_Plus and LightMAC_Plus. Sect. 6 deals with the security analysis of two-keyed variants of sequential constructions (i.e., SUM-ECBC and 3kf9) and provides an alternative security proof for SUM-ECBC and 3kf9. Finally, we conclude the paper by discussing some open problems and difficulties in proving the PRF security of the single-keyed DbHtS in Sect. 7.

2 Preliminaries

We will introduce necessary symbols and notations in Sect. 2.1 followed by the required security definitions in Sect. 2.2. We discuss the lazy sampling of permutations in Sect. 2.3.

Table 1: #Keys denote the number of block cipher keys used in the construction. Rate defines the average number of message blocks processed by a single execution of block cipher. q denotes the total number of queries and ℓ denotes the maximum number of message blocks in all q queries. Only the dominant terms of the security bounds are listed. ($\star$) symbolizes the new bound is improved over the existing one and (†) symbolizes the corresponding bound is incorrect. We discuss this issue at the end of Sect. 6.3.

Construction	(#Keys, rate)	Old bound	New Bound
Three-keyed DbHtS			
SUM-ECBC	$(4, 1/2)$	$q^3\ell^4/2^{2n}$	$q\ell^2/2^n + q^3/2^{2n}(\star)$
PMAC_Plus	$(3, 1)$	$q^3\ell^3/2^{2n} + q\ell/2^n$	$q^3\ell/2^{2n} + q^2\ell^2/2^{2n}(\star)$
3kf9	$(3, 1)$	$q^3\ell^3/2^{2n} + q\ell/2^n$ (†)	$q^3\ell^4/2^{2n}$
LightMAC_Plus	$(3, 1)$	$q^3/2^{2n}$	$q^3/2^{2n}$
Two-keyed DbHtS			
2K-ECBC_Plus	$(3, 1/2)$	-	$q\ell^2/2^n + q^3\ell^2/2^{2n}$
2K-PMAC_Plus	$(2, 1)$	-	$q^3\ell/2^{2n} + q^2\ell^2/2^{2n}$
2kf9	$(2, 1)$	-	$q^3\ell^4/2^{2n}$
2K-LightMAC_Plus	$(2, 1)$	-	$q^3/2^{2n} + q/2^n$

Sect. 2.4 briefly discusses about H-Coefficient Technique. Some basic results of linear algebra is given in Sect. 2.5, followed by the result on xor of two permutations in Sect. 2.6.

2.1 Notations

Given a finite set $\mathcal{S}$ and a random variable X, we write $X \xleftarrow{\$} \mathcal{S}$ to denote that X is sampled uniformly at random from $\mathcal{S}$.

We fix a positive integer n for the rest of this section. $\{0,1\}^n$ denotes the set of all binary strings of length n. A *block* is defined as an n-bit binary string. The functions $\mathsf{fix0}$ and $\mathsf{fix1}$ take an n-bit binary string x and return x with its least significant bit set to 0 and 1 respectively. We write $\mathbf{0}$ to denote the all zero binary string and $\mathbf{1}$ to denote the binary string whose first $n-1$ bits are all zeros and the least significant bit is one.

A tuple $\widetilde{x}$ over an index set $\mathcal{I}$ is denoted by $(x_i : i \in \mathcal{I})$. For notational simplicity, we sometimes write the tuple as $(x_i)_i$ when the index set is understood from the context. The i-th element of a tuple $\widetilde{x}$ is represented by x_i. Length of a tuple $\widetilde{x}$ refers to the number of elements in it and is denoted by $|\widetilde{x}|$. An element x_i of a tuple $\widetilde{x}$ is called a *fresh value* if for all $j \neq i$, $x_i \neq x_j$. Otherwise, we say x_i is a *colliding value* or alternatively *not fresh in* $\widetilde{x}$. A tuple is said to be *distinct* if each of its elements is fresh. Otherwise, we say it is *not a fresh tuple*. Concatenation of two tuples $\widetilde{x}$ and $\widetilde{y}$ is denoted by $(\widetilde{x}, \widetilde{y})$. A tuple is said to be a *block-tuple*, if each of its element is a member of $\{0,1\}^n$. For a set $\mathcal{X}$, $\mathcal{X}^{(q)}$ denotes the set of all distinct tuples over $\mathcal{X}$ of length q. If $\mathcal{X} = \{0,1\}^n$, then $(\{0,1\}^n)^{(q)}$ denotes the set of all block-wise distinct tuples of length q. For a positive integer q, we write $[q]$ to denote the set $\{1, 2, \ldots, q\}$. We denote the empty set as Φ.

We regard the set $\{0,1\}^n$ as a set of integers $\{0, 1, \ldots, 2^n - 1\}$ by converting an n-bit binary string $(a_{n-1}a_{n-2}\ldots a_1a_0) \in \{0,1\}^n$ to an integer $a_{n-1}2^{n-1} + a_{n-2}2^{n-2} + \ldots + a_1 2 + a_0$, where multiplication and addition are integer arithmetic. Let $GF(2^n)$ be the field with 2^n elements and we regard $\{0,1\}^n$ as $GF(2^n)$. We identify an n-bit string $(a_{n-1}a_{n-2}\ldots a_1a_0) \in \{0,1\}^n$ as a polynomial $a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \ldots + a_1x + a_0 \in GF(2)[x]$. To do operations on the elements of $GF(2^n)$, we fix an irreducible polynomial $f(x) \in GF(2)[x]$ and addition, denoted as $\oplus$ and multiplication, denoted as $\cdot$ are done modulo $f(x)$. With a slight abuse of notation, we write $\{0,1\}^n$ to denote the set of n-bit binary strings or the field $GF(2^n)$.

The set of all functions from $\mathcal{X}$ to $\mathcal{Y}$ is denoted as $\mathsf{Func}(\mathcal{X}, \mathcal{Y})$. Similarly, the set of all

permutations over $\mathcal{X}$ is represented by $\mathsf{Perm}(\mathcal{X})$. A function ϕ mapping an element from an arbitrary domain to $\{0,1\}^n$ is called a *block function*. Similarly, if ϕ maps to $(\{0,1\}^n)^2$, we call it a *double-block function*. We write a double block function as $\phi = (\phi_0, \phi_1)$, where ϕ_0 and ϕ_1 are block functions. We denote the set of all block functions with domain $\mathcal{X}$ as $\mathsf{Func}(\mathcal{X})$ [1] and the set of all block permutations as Perm. For integers $1 \leq b \leq a$, we write $(a)_b$ to denote $a(a-1)\dots(a-b+1)$, where $(a)_0 = 1$ by convention.

2.2 Security Definitions

PRF AND PRP. A *keyed function* with the key space $\mathcal{K}$, the domain $\mathcal{X}$ and the range $\mathcal{Y}$ is a function $F : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$ and we denote $F(K, X)$ by $F_K(X)$. Similarly, a *keyed permutation* with the key space $\mathcal{K}$ and the domain $\mathcal{X}$ is a mapping $E : \mathcal{K} \times \mathcal{X} \to \mathcal{X}$ such that for each key $K \in \mathcal{K}$, $X \mapsto E(K, X)$ is a permutation over $\mathcal{X}$ and we denote $E_K(X)$ for $E(K, X)$.

Let A be an oracle algorithm with oracle access to a function from $\mathcal{X}$ to $\mathcal{Y}$ that outputs a single bit. Without loss of generality, we assume that A can make at most q oracle queries with running time at most t. We call such an oracle algorithm a *distinguisher*. We define the prf-advantage of A against a keyed function F as

$$\mathbf{Adv}_F^{\mathrm{PRF}}(\mathsf{A}) := |\Pr[K \leftarrow_{\$} \mathcal{K} : \mathsf{A}^{F_K} = 1] - \Pr[\mathsf{RF} \leftarrow_{\$} \mathsf{Func}(\mathcal{X}, \mathcal{Y}) : \mathsf{A}^{\mathsf{RF}} = 1]|.$$

Similarly, we define the prp-advantage of the distinguisher A against a keyed permutation E as

$$\mathbf{Adv}_E^{\mathrm{PRP}}(\mathsf{A}) := |\Pr[K \leftarrow_{\$} \mathcal{K} : \mathsf{A}^{E_K} = 1] - \Pr[\Pi \leftarrow_{\$} \mathsf{Perm}(\mathcal{X}) : \mathsf{A}^{\Pi} = 1]|.$$

For a keyed function family F, $\mathbf{Adv}_F^{\mathrm{xxx}}(q,t)$ denotes $\max_{\mathsf{A}} \mathbf{Adv}_F^{\mathrm{xxx}}(\mathsf{A})$, where xxx is either prf or prp and maximum is taken over all distinguishers A running in time at most t and make at most q queries. If F is a keyed function (resp. permutation) family such that $\mathbf{Adv}_F^{\mathrm{xxx}}(q,t) \leq \delta$, then we say F is a $(\delta : q, t)$-PRF (resp. PRP). If A is a computationally unbounded distinguisher, then we disregard the time parameter from its advantage definition.

(ALMOST-XOR) UNIVERSAL ADVANTAGE OF HASH FUNCTION. Let $\mathcal{K}_h$ and $\mathcal{X}$ be two non-empty finite sets and $\epsilon > 0$. A keyed function $H : \mathcal{K}_h \times \mathcal{X} \to \{0,1\}^n$ is a ϵ-(almost-xor) universal hash function, if for any distinct $X, X' \in \mathcal{X}$ and for any $Y \in \{0,1\}^n$,

$$\Pr[K_h \leftarrow_{\$} \mathcal{K}_h : H_{K_h}(X) \oplus H_{K_h}(X') = Y] \leq \epsilon.$$

Moreover, H is said to be an ϵ-universal hash function, if for any distinct $X, X' \in \mathcal{X}$,

$$\Pr[K_h \leftarrow_{\$} \mathcal{K}_h : H_{K_h}(X) = H_{K_h}(X')] \leq \epsilon.$$

DOUBLE-BLOCK HASH FUNCTION. A keyed hash function H is said to be a Double-block Hash (DbH) function, if $H : \mathcal{K}_h \times \mathcal{X} \to (\{0,1\}^n)^2$. We denote the pair of block outputs as $(H_{K_h,0}(X), H_{K_h,1}(X))$, where $X \in \mathcal{X}$ and $H_{K_h,0}(X) \| H_{K_h,1}(X) = H_{K_h}(X)$.

2.3 Lazy Sampling of Random Permutation

Suppose, a distinguisher A is interacting with a random permutation $\Pi \leftarrow_{\$} \{0,1\}^n$. This interaction is simulated by a simulator that maintains a partial function (or sometimes we call it a list) Ψ which is initially set to an empty function (i.e., a function with empty domain). On the i-th query x_i, the simulator checks whether $x_i \in Dom(\Psi)$, where $Dom(\Psi)$ is the set of all elements of $\{0,1\}^n$ on which Ψ is defined. If so, the corresponding response y_i is set to $\Psi(x_i)$. Else, the response is sampled uniformly from $\{0,1\}^n \setminus Ran(\Psi)$, where $Ran(\Psi)$ is the set of all elements of $\{0,1\}^n$ which have at least one preimage under Ψ and x_i added to the set $Dom(\Psi)$.

[1] When $\mathcal{X} = \{0,1\}^n$ then we write Func to denote $\mathsf{Func}(\{0,1\}^n)$

2.4 H-Coefficient Technique

In this section, we briefly discuss the H-Coefficient Technique [Pat08c, CLL+14] which has been introduced by Patarin [Pat08c] and recently regained attention since Chen and Steinberger used it to analyze the iterated Even-Mansour cipher [CS14]. This technique gives a kind of "systematic" way to upper bound the statistical distance between the answers of two interactive systems and is typically used to prove the information theoretic pseudo randomness of constructions. In this setting, we consider a computationally unbounded and hence deterministic distinguisher A that interacts with either the real oracle, i.e., the construction of our interest, or the ideal oracle which is usually considered to be a uniform random function or permutation. The collection of all the queries and responses that A made and received to and from the oracle, is called the *transcript* of A, denoted as τ. Sometimes, we allow the oracle to release more internal information to A only after A completes all its queries and responses, but before it outputs its decision bit. In this case, the transcript of A includes the additional information about the oracle and clearly the maximum distinguishing advantage of A in this setting can not be less than that of without additional information. Observe that the transcript τ is a random variable and the randomness of the distribution of τ only comes from the randomness of the oracle with which A interacts.

Let X_{re} and X_{id} denote the probability distributions of the transcript τ induced by the real oracle and the ideal oracle respectively. The probability of realizing a transcript τ in the ideal oracle (i.e., $\Pr[X_{\mathrm{id}} = \tau]$) is called the *ideal interpolation probability*. Similarly, one can define the *real interpolation probability*. A transcript τ is said to be *attainable* with respect to A if the ideal interpolation probability is non-zero (i.e., $\Pr[X_{\mathrm{id}} = \tau] > 0$). We denote the set of all attainable transcripts by Θ. Following these notations, we state the main theorem of H-Coefficient Technique [Pat08c, CLL+14] as follows:

Theorem 1 (H-Coefficient Technique). *Let A be a fixed deterministic distinguisher that has access to either the real oracle $\mathcal{O}_{\mathrm{re}}$ or the ideal oracle $\mathcal{O}_{\mathrm{id}}$. Let $\Theta = \Theta_{\mathrm{g}} \sqcup \Theta_{\mathrm{b}}$ (disjoint union) be some partition of the set of all attainable transcripts of A. Suppose there exists $\epsilon_{\mathrm{ratio}} \geq 0$ such that for any $\tau \in \Theta_{\mathrm{g}}$,*

$$\frac{\Pr[X_{\mathrm{re}} = \tau]}{\Pr[X_{\mathrm{id}} = \tau]} \geq 1 - \epsilon_{\mathrm{ratio}},$$

and there exists $\epsilon_{\mathrm{bad}} \geq 0$ such that $\Pr[X_{\mathrm{id}} \in \Theta_{\mathrm{b}}] \leq \epsilon_{\mathrm{bad}}$. Then,

$$\mathbf{Adv}^{\mathcal{O}_{\mathrm{id}}}_{\mathcal{O}_{\mathrm{re}}}(\mathsf{A}) := |\Pr[\mathsf{A}^{\mathcal{O}_{\mathrm{re}}} = 1] - \Pr[\mathsf{A}^{\mathcal{O}_{\mathrm{id}}} = 1]| \leq \epsilon_{\mathrm{ratio}} + \epsilon_{\mathrm{bad}}. \tag{1}$$

When $\mathcal{O}_{\mathrm{id}}$ is a uniform random function and $\mathcal{O}_{\mathrm{re}}$ is some keyed construction defined over the same domain, then Eqn. (1) says that $\mathbf{Adv}^{\mathrm{prf}}_{\mathcal{O}_{\mathrm{re}}}(\mathsf{A}) \leq \epsilon_{\mathrm{ratio}} + \epsilon_{\mathrm{bad}}$.

2.5 Some Results on Linear Algebra

For a matrix L of dimension $s \times t$ defined over $GF(2^n)$, $L[i][j]$ denotes the element in its i-th row and j-th column. For a column vector c of dimension $s \times 1$, $L \| c$ denotes the augmented matrix of dimension $s \times (t+1)$. For any row vector $R := (R_1, \ldots, R_t)$ of dimension $1 \times t$, transpose of row vector R, denoted as R^{T}, denotes the column vector

$$R^{\mathsf{T}} := \begin{pmatrix} R_1 \\ R_2 \\ \vdots \\ R_t \end{pmatrix}$$

of dimension $t \times 1$.

One can represent any system of s linear equations with t unknowns $(Y_1, \ldots, Y_t)$ defined over $GF(2^n)$, denoted as $\mathcal{L}$, as a matrix L of dimension $s \times t$, where the i-th equation $\mathcal{L}_i := a_{i1} \cdot Y_1 \oplus \ldots \oplus a_{it} \cdot Y_t = c_i$, where $c_i \in GF(2^n)$, corresponds to the i-th row vector of L as $(a_{i1}, \ldots, a_{it})$. We say $\mathcal{L}$ is *consistent* if it has at least one solution, otherwise we call it *inconsistent*. For $\mathcal{L}$ to be consistent, one must have $rank(L) = rank(L\|c)$ [2], where $c = (c_1, \ldots, c_s)^\mathsf{T}$. $\mathcal{L}$ has a unique solution if $rank(L) = t$ and it has many solutions if $rank(L) < t$.

Let $L \cdot Y^\mathsf{T} = c$ represent a system of s linear equations with t unknowns $(Y_1, \ldots, Y_t)$, where $rank(L) = r$ and the elements of L are from $GF(2^n)$. Let $Y := (Y_1, \ldots, Y_t)$ be without replacement samples from a set $\mathcal{Y} \subseteq \{0,1\}^n$ and c is any arbitrary column vector of dimension $s \times 1$ with its elements from $GF(2^n)$. Thus, the probability of realizing a particular solution is at most $\frac{1}{(|\mathcal{Y}|-t+r)_r}$ as stated formally in the following lemma.

Lemma 1. *Let $Y := (Y_1, \ldots, Y_t)$ be without replacement samples from a set $\mathcal{Y} \subseteq \{0,1\}^n$ and L be a matrix of dimension $s \times t$ defined over $GF(2^n)$. Then, for any given column vector c of dimension $s \times 1$ over $GF(2^n)$, we have*

$$\Pr[(L)_{s\times t} \cdot Y^\mathsf{T} = c] \leq \frac{1}{(|\mathcal{Y}| - t + r)_r},$$

where r is the rank of the matrix L.

Proof. Since, the rank of L is r, the number of free variables in the system of equations is $(t - r)$. Now, each choice of free variables, which necessarily has to be distinct, uniquely determine the remaining variables such that the overall system of equations is satisfied. Therefore, the number of solutions is at most $(|\mathcal{Y}|)_{t-r}$ and the total number of ways we can choose t distinct variables $(Y_1, \ldots, Y_t)$ is $(|\mathcal{Y}|)_t$. Dividing the former one by later gives the result. □

2.6 Sum of Two Identical Permutations

In this section, we briefly revisit the security result of the sum of two identical random permutations. The sum of two permutations is one of the PRP to PRF transformations, suggested by Bellare et al. [BKR98] as:

$$\mathsf{SUM}_{E_{K_1}, E_{K_2}}(x) = E_{K_1}(x) \oplus E_{K_2}(x),$$

where E_{K_1} and E_{K_2} are two independent PRPs. We call this construction as the **sum construction**. This construction was later analyzed by Lucks [Luc00] who proved $2^{2n/3}$ security. Further improvements have been shown in [Pat08b, Pat10, Pat13]. The results are natively inherited by the construction that consists of the xor of three or more independent PRPs [CLP14, MP15].

Security of the single-keyed sum construction (i.e., the sum construction with $K_1 = K_2$), as simulated through the domain separation, suggested in [Luc00, BI99], has been shown to be provably secure by Bellare and Impagliazzo [BI99] up to $O(n) \cdot \frac{q^{3/2}}{2^{3n/2}}$. However, their security proof is too sketchy to verify and contains unverifiable gaps. In a series of papers [Pat08b, Pat10, Pat13], Patarin proved the optimal security of the construction using the standard H technique [Pat13] and the mirror theory technique [Pat10] but the proof is still unverifiable. Recently, Dai et al. [DHT17] showed $(1.5q + 3\sqrt{q})/2^n$ bound for the sum construction and its single-keyed variant using the chi-squared method.

In the following, we state and prove that the single-keyed sum construction is a secure PRF that offers $2n/3$-bit security. Formally, we have the following result:

[2] rank of a matrix L is defined as the maximum number of linearly independent columns of L

Lemma 2. *For any block tuple $(T_1, \ldots, T_q)$ of length q such that each T_i is non-zero, let*

$$\mathcal{Z} = \{(U_i, V_i)_i : U_i \oplus V_i = T_i \ \forall i \in [q],\ (U_i, V_i)_i \in (\{0,1\}^n)^{(2q)}\}.$$

Then, $|\mathcal{Z}| \geq \frac{(2^n)_{2q}}{2^{nq}}(1 - \frac{6q^3}{2^{2n}})$, with the assumption $q \leq 2^{n-2}$.

Proof. Datta et al. [DDN+17] showed in Theorem 2, that for any set $\mathcal{B} := \{B_1, \ldots, B_s\} \subseteq \{0,1\}^n$ and a q-length block tuple $(T_1, \ldots, T_q)$ such that each T_i is non-zero, the following holds:

$$|\underbrace{\{(H_i^0, H_i^1) : H_i^0 \oplus H_i^1 = T_i,\ (H_i^0, H_i^1)_i \in (\{0,1\}^n \setminus \mathcal{B})^{(2q)}\}}_{\mathcal{H}}| \geq \frac{(2^n - s)_{2q}}{2^{nq}}(1 - \mu_2), \quad (2)$$

where $\mu_2 \leq \frac{qs^2 + 2sq^2 + 4q^3/3}{(2^n - s - 2q)^2}$.
Now, note that the set $\mathcal{Z}$ is the same as $\mathcal{H}$ with $\mathcal{B}$ as an empty set and hence $s = 0$. Therefore, from Eqn. (2) and with the assumption $q \leq 2^{n-2}$, we obtain the result. □

Remark 1. It is natural to wonder that why we prove a weaker bound of the construction in the face of its existing optimal security bound. We note that the optimal security bound of the construction has been proved for PRF advantage. However, we need a counting results on the number of permutations to apply the H-coefficient technique. Currently, we do not know how to use this optimum PRF security result directly in our proof setting. In this regard, one can possibly use the Patarin's proof of sum construction using the mirror theory [Pat10] technique. However, the reliability of Patarin's proof [Pat10] is debatable. Thus, we independently prove the security of the sum construction up to $2^{2n/3}$ bound, which is good enough for our purpose. Moreover, as we will see later in the paper that we will use the above result in the security analysis of the two-keyed DbHtS construction. The dominant term of its security bound appears due to the cover-free advantage (defined later in Sect. 3.3) of its underlying DbH function, overkilling the optimal bound of the single-keyed sum construction.

3 DbHtS : A BBB Secure VIL PRF Paradigm

Hash-then-PRF or (HtP) is a well known paradigm for constructing a Variable Input Length (VIL) PRF by composing a universal hash function and a Fixed Input Length (FIL) PRF due to Shoup [Sho04]. Formally, HtP composition result says the following:

If H is an $\epsilon(\ell)$ universal hash function that outputs m bits and F is a $(\delta : q)$-PRF with domain $\{0,1\}^m$, then the composition construction $(F \circ H)$ is a $(\delta + \epsilon(\ell)q^2/2 : q, \ell)$-PRF.

To obtain the BBB PRF-security of a keyed construction following the HtP paradigm, the PRF advantage bound of F and the universal advantage bound of H need to be beyond birthday. It is feasible to construct a double block hash function (which outputs $m = 2n$ bits) with $\epsilon(\ell) = O(\ell^c 2^{-2n})$ (e.g., multi-linear hash [HK97], PolyHash [dB93, BJKS93, Tay93] etc). However, obtaining a beyond birthday bound secure PRF over $2n$ bits input would not be easy and efficient. It is needless to say that a beyond birthday bound secure F can be constructed from scratch or one can try some variants of the 5-rounds Luby-Rackoff [Pat98] or the Benes-Butterfly construction [Pat08a] that gives a beyond birthday bound secure PRF over $2n$ bits input. However, the former suggestion is non-trivial and the latter one would require at least 6 primitive calls for realizing $2n$ bits to n bits PRF. Moreover, its security proof is based on pseudorandom function. A possible way out is to instantiate each pseudorandom function with the sum of two independent block ciphers. But this idea comes at the cost of using total 12 block cipher keys. To realize it with less block cipher keys is non-trivial. .

This motivates us to design a paradigm for constructing a beyond birthday secure VIL PRF, where the underlying hash function H is required to achieve some stronger security assumption than the universal property, whereas we require a simple and efficient keyed function that is not required to be a PRF.

3.1 Double-block Hash-then-Sum (DbHtS) Paradigm

In this section, we describe the Double-block Hash-then-Sum (in short, DbHtS) paradigm to build a BBB secure VIL PRF. In this paradigm, a Double-block Hash (DbH) function is used with a very simple and efficient single-keyed or two-keyed sum function:

- Single-Keyed Sum Function: $\mathsf{Sum}_K(x, y) = E_K(x) \oplus E_K(y)$,
- Two-Keyed Sum Function: $\mathsf{Sum}_{K_1,K_2}(x, y) = E_{K_1}(x) \oplus E_{K_2}(y)$,

where E_K, E_{K_1}, E_{K_2} are n-bit block ciphers and K_1 and K_2 are independent. Given a DbH function and the sum function over two blocks, we apply the composition of the DbH function and the sum function to realize the DbHtS construction. Based on the types of sum function (i.e., single-keyed or two-keyed) used in the composition, we categorize DbHtS into following two categories:

- Three-Keyed DbHtS: $\mathsf{C}_3[H, E](M) := \mathsf{Sum}_{K_1,K_2}(H_{K_h,0}(M), H_{K_h,1}(M))$.
- Two-Keyed DbHtS: $\mathsf{C}_2[H, E](M) := \mathsf{Sum}_K(H_{K_h,0}(M), H_{K_h,1}(M))$.

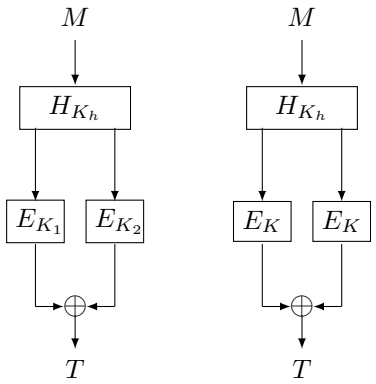

Figure 3.1: Two different types of Double block Hash then Sum constructions. Left : three-keyed construction $\mathsf{C}_3[H, E](M) := E_{K_1}(H_{K_h,0}(M)) \oplus E_{K_2}(H_{K_h,1}(M))$. Right : two-keyed construction $\mathsf{C}_2[H, E](M) := E_K(H_{K_h,0}(M)) \oplus E_K(H_{K_h,1}(M))$ where $K_h \in \mathcal{K}_h$. For simplicity of notations we sometimes simply refer them as C_3 and C_2 respectively.

We use the name *two-keyed (or three-keyed)* DbHtS construction, as we count the hash key as one key and the sum function requiring one key (or two independent keys respectively), independent of the hash key. However, a concrete instantiation of a DbH function may require multiple keys.

Most of the BBB secure deterministic MACs like SUM-ECBC, PMAC_Plus, 3kf9, Light-MAC_Plus are specific instantiations of the three-keyed DbHtS paradigm. However, we would like to work with the two-keyed DbHtS construction as it involves more challenging analysis than its three-keyed version. We would like to note that the three-keyed DbHtS does not outperform its two-keyed version in terms of providing improved security bound as evident from the last column of Table 1. Its only advantage lies in its simpler security proof, as the number of cases to analyze gets reduced.

Remark 2. As the sum function is not a PRF [3], we can not apply the HtP composition result directly to analyze the security of DbHtS. This says that we need a different type of composition result for the security analysis of DbHtS construction in which we require some higher security properties from its underlying DbH function instead of having only the universal property.

3.2 Proof Idea of Two-Keyed DbHtS Construction

In this section, we provide a brief idea of proving the security of the two-keyed DbHtS construction. We believe that this will motivate the reader to understand the crux of the main proof given in Sect. 3.4 and also help to understand a few definitions introduced in Section 3.3.

We use the H-Coefficient technique, which requires us to bound: (i) the probability of the bad transcripts in the ideal oracle and (ii) the ratio of the real to ideal interpolation probability of the good transcripts. The computation of the real interpolation probability is reduced to the probability of satisfying the following q many bi-variate equations:

$$\begin{cases} \Pi(H_{K_h,0}(M_1)) \oplus \Pi(H_{K_h,1}(M_1)) = T_1, \\ \Pi(H_{K_h,0}(M_2)) \oplus \Pi(H_{K_h,1}(M_2)) = T_2, \\ \qquad\qquad \vdots \\ \Pi(H_{K_h,0}(M_q)) \oplus \Pi(H_{K_h,1}(M_q)) = T_q, \end{cases}$$

Now, to obtain a meaningful lower bound of the real interpolation probability, we need at least one of the inputs of Π to be fresh for each equations and each T_i to be non-zero. In this regard, we call a transcript to be good if for each $i \in [q]$, either $H_{K_h,0}(M_i)$ or $H_{K_h,1}(M_i)$ or both are fresh in the following tuple:

$$\widetilde{H} := \big((H_{K_h,0}(M_1), H_{K_h,1}(M_1)), (H_{K_h,0}(M_2), H_{K_h,1}(M_2)), \ldots, (H_{K_h,0}(M_q), H_{K_h,1}(M_q))\big),$$

and every T_i is non-zero. In other words, we call a transcript to be bad if one of the following three conditions occur:

(i) $\exists i \in [q]$ such that $H_{K_h,0}(M_i) = H_{K_h,1}(M_i)$. We call it *the collision condition*.

(ii) $\exists i \neq j, i \neq k, b, b' \in \{0,1\}$ such that $H_{K_h,0}(M_i) = H_{K_h,b}(M_j), H_{K_h,1}(M_i) = H_{K_h,b'}(M_k)$. We call it *the covered condition*.

(iii) $\exists i : T_i = 0$.

If none of the above conditions happen, then for each $i \in [q]$, either $H_{K_h,0}(M_i)$ and $H_{K_h,1}(M_i)$ both are fresh in $\widetilde{H}$, or any one of the $H_{K_h,0}(M_i)$ or $H_{K_h,1}(M_i)$ are colliding in $\widetilde{H}$. If both the inputs are fresh, we can directly apply Lemma 2. Otherwise (w.l.o.g. assume that $H_{K_h,1}(M_i)$ is non-fresh), the permutation output of $H_{K_h,1}(M_i)$ is defined (or may need to sampled, if not defined already), which in turn uniquely determines the permutation output of $H_{K_h,0}(M_i)$ (as we have already fixed the response T_i). However, this uniquely determined output may collide with some already sampled values in range. We call this condition *the range collision condition* which actually creates a *permutation input-output compatibility issue*.

Therefore, bounding the probability of bad transcripts is nothing but to bound all of the above events. A detailed treatment of bounding the bad probability is given in Sect. 3.4. Finally, by computing the ratio of real to ideal interpolation probability concludes the proof of $\mathsf{C}_2[H, E]$.

[3] One can construct a PRF distinguisher A with PRF advantage very close to 1. A makes four queries $(x_1, y_1), (x_1, y_2), (x_2, y_1), (x_2, y_2)$ and check if the xor of their output is zero which holds with probability 1 for real oracle, and holds with probability 2^{-n} for ideal oracle.

Remark 3. We would like to mention that we have identified the bad events with a clear intention in mind to apply the sum construction result when these bad events do not happen. As a result, the bad events essentially boil down to investigating the collision, the covered and the range collision condition of the underlying DbH function. Whether all of these bad events directly lead to an attack in the construction, is not known.

3.3 Security Notions for DbH Functions

In this section, we define the necessary security notions of a DbH function which will be required in proving the main security result of this paper.

Let $\widetilde{g}$ and $\widetilde{h}$ be two tuples of length q. We say that the tuple $(\widetilde{g}, \widetilde{h})$ is covered at an index $i \in [q]$, if g_i and h_i are colliding values in $(\widetilde{g}, \widetilde{h})$, but they do not collide at the same value i.e., $g_i \neq h_i$ for all $i \in [q]$. As a matter of fact, $(\widetilde{g}, \widetilde{h})$ is covered at an index $i \in [q]$ if and only if $\exists j \neq i, k \neq i$ such that either of the following conditions hold:

$$(i)\ g_i = g_j,\ h_i = h_k \quad (ii)\ g_i = h_j,\ h_i = h_k \quad (iii)\ g_i = g_j,\ h_i = g_k \quad (iv)\ g_i = h_j,\ h_i = g_k$$

As there is no restriction on j and k, we can have $j = k$ and therefore plugging-in $j = k$ in (i) and (iv) gives rises the following two possibilities:

$$(v)\ \exists j \neq i\ :\ g_i = h_j,\ h_i = g_j \quad (vi)\ \exists j \neq i\ :\ g_i = g_j,\ h_i = h_j.$$

Note that, for (ii) and (iii), $j = k$ case is excluded by the *"no-collision at the same value condition"* at index i. Moreover, it is needless to mention that for $q = 1$, the tuple (g_1, h_1) is always cover-free.

We say that there is a *cross-collision* between $\widetilde{g}$ and $\widetilde{h}$, when any one of the conditions (ii)-(v) occur. The tuple $(\widetilde{g}, \widetilde{h})$ is called covered, if it is covered at some index $i \in [q]$. If the tuple $(\widetilde{g}, \widetilde{h})$ is not covered, we say that it is cover-free. So for a cover-free tuple $(\widetilde{g}, \widetilde{h})$ such that none of (g_i, h_i) collides at the same value, for every $i \in [q]$, either g_i is fresh in $(\widetilde{g}, \widetilde{h})$ or h_i is fresh in $(\widetilde{g}, \widetilde{h})$ or both. The tuple $(\widetilde{g}, \widetilde{h})$ is said to be *weak covered*, if $(\widetilde{g}, \widetilde{h})$ is covered but there is no *cross-collision* between $(\widetilde{g}, \widetilde{h})$. In other words, if $(\widetilde{g}, \widetilde{h})$ is weak covered, then only condition (i) or (vi) holds. Thus, a weak covered tuple is always a covered tuple but the other direction is not true.

Example 1. Let us consider two tuples $\widetilde{g} = (a, b, a, d, e, f)$ and $\widetilde{h} = (b, c, c, d, e, g)$ of length 6. Observe that $(\widetilde{g}, \widetilde{h})$ is covered at index $1, 2, 3$, but not covered at index 4 and 5 as $g_4 = h_4$ and $g_5 = h_5$. Moreover, it is not a weak covered tuple. On the other hand, the tuple $\widetilde{g} = (a, b, a, d, e, f)$ and $\widetilde{h} = (u, u, v, x, y, z)$ of length 6 is a weak covered tuple.

3.3.1 Security Definitions for DbH Function

Having defined the cover-free tuple, we now introduce the necessary security definitions for a DbH function. We begin with defining the cover-free advantage of a DbH function H.

For a q, a distinct tuple $(M_1, \ldots, M_q)$ and fixed $i, j, k \in [q]$ such that $i \neq j, i \neq k$, we define the following event:

$$\begin{aligned}\mathsf{CF}_{ijk} \quad := \quad & \bigvee_{b,b' \in \{0,1\}} \Big(H_{K_h,0}(M_i) = H_{K_h,b}(M_j), H_{K_h,1}(M_i) = H_{K_h,b'}(M_k)\Big) \\ & \bigvee \Big(H_{K_h,0}(M_i) = H_{K_h,0}(M_j), H_{K_h,1}(M_i) = H_{K_h,1}(M_j)\Big).\end{aligned}$$

We also define the event

$$\begin{aligned}\mathsf{WCF}_{ijk} \quad := \quad & \Big(H_{K_h,0}(M_i) = H_{K_h,0}(M_j), H_{K_h,1}(M_i) = H_{K_h,1}(M_k)\Big) \\ & \bigvee \Big(H_{K_h,0}(M_i) = H_{K_h,0}(M_j), H_{K_h,1}(M_i) = H_{K_h,1}(M_j)\Big).\end{aligned}$$

Definition 1. Let $\mathcal{K}_{\text{bad}}$ be a function from a set of tuple of q distinct messages to the power set of hash keys $\mathcal{P}(\mathcal{K}_h)$. We say that a DbH function $H : \mathcal{K}_h \times \mathcal{M} \to (\{0,1\}^n)^2$ is a $(\mathcal{K}_{\text{bad}}, \epsilon_{\text{cf}})$-cover-free DbH function, if for any q-tuple of distinct messages $(M_1, \ldots, M_q)$, each of length at most ℓ blocks such that

$$\forall i, j, k \text{ such that } i \neq j, i \neq k, \ \Pr[\mathsf{CF}_{ijk} \text{ holds}, K_h \in \mathcal{K}_h \setminus \mathcal{K}_{\text{bad}}(M_1, \ldots, M_q)] \leq \epsilon_{\text{cf}}.$$

We call ϵ_{cf} to be the *cover-free advantage for three messages* of the DbH function H.

Similarly, we define the weak-cover-free advantage of the DbH function H as follows:

Definition 2. Let $\mathcal{K}_{\text{bad}}$ be a function from a set of tuple of q distinct messages to the power set of hash keys $\mathcal{P}(\mathcal{K}_h)$. We say that a DbH function $H : \mathcal{K}_h \times \mathcal{M} \to (\{0,1\}^n)^2$ is a $(\mathcal{K}_{\text{bad}}, \epsilon_{\text{wcf}})$-weak-cover-free DbH function, if for any q-tuple of distinct messages $(M_1, \ldots, M_q)$, each of length at most ℓ blocks such that,

$$\forall i, j, k \text{ such that } i \neq j, i \neq k, \ \Pr[\mathsf{WCF}_{ijk} \text{ holds}, K_h \in \mathcal{K}_h \setminus \mathcal{K}_{\text{bad}}(M_1, \ldots, M_q)] \leq \epsilon_{\text{wcf}}.$$

We call ϵ_{wcf} to be the *weak-cover-free advantage for three messages* of the DbH function H.

NOTE. It is to be noted that in both the definitions $\mathcal{K}_{\text{bad}}$ is treated as a function that maps a tuple of q distinct messages to a subset of hash keys. As we work on a fixed tuple of q distinct messages, the set $\mathcal{K}_{\text{bad}}(M_1, \ldots, M_q)$ is a fixed set and hence for the sake of notational simplicity, we abuse the notation $\mathcal{K}_{\text{bad}}$ to indicate the image set of the function $\mathcal{K}_{\text{bad}}$.

Note that, for a cover-free tuple $(\widetilde{g}, \widetilde{h})$ and for a fixed index $i \in [q]$, either g_i is non-fresh and h_i is fresh in $(\widetilde{g}, \widetilde{h})$ or g_i is fresh and h_i is non-fresh in $(\widetilde{g}, \widetilde{h})$ or both are fresh in $(\widetilde{g}, \widetilde{h})$. Considering the first two cases, we now define the block-wise universal advantage of DbH function as follows:

For a q, a distinct tuple $(M_1, \ldots, M_q)$ and fixed $i, j \in [q]$ such that $i \neq j$, we define the following event:

$$\begin{aligned}\mathsf{UNIV}_{ij} \quad := \quad & \Big(H_{K_h,0}(M_i) = H_{K_h,0}(M_j)\Big) \bigvee \Big(H_{K_h,0}(M_i) = H_{K_h,1}(M_j)\Big) \\ & \bigvee \Big(H_{K_h,1}(M_i) = H_{K_h,0}(M_j)\Big) \bigvee \Big(H_{K_h,1}(M_i) = H_{K_h,1}(M_i)\Big).\end{aligned}$$

We also define the event

$$\mathsf{WUNIV}_{ij} \quad := \quad \Big(H_{K_h,0}(M_i) = H_{K_h,0}(M_j)\Big) \bigvee \Big(H_{K_h,1}(M_i) = H_{K_h,1}(M_j)\Big).$$

Definition 3. We say that a DbH function $H : \mathcal{K}_h \times \mathcal{M} \to (\{0,1\}^n)^2$ is $(\mathcal{K}_{\text{bad}}, \epsilon_{\text{univ}})$-block-wise universal DbH function, if for any q-tuple of distinct messages $(M_1, \ldots, M_q)$, each of length at most ℓ blocks such that

$$\forall i, j \text{ such that } i \neq j, \ \Pr[\mathsf{UNIV}_{ij} \text{ holds}, K_h \in \mathcal{K}_h \setminus \mathcal{K}_{\text{bad}}] \leq \epsilon_{\text{univ}}.$$

We call ϵ_{univ} to be the *block-wise universal advantage for two messages* of DbH function H.

Similarly, we define the weak-block-wise universal advantage of the DbH function H as follows;

Definition 4. We say that a DbH function $H : \mathcal{K}_h \times \mathcal{M} \to (\{0,1\}^n)^2$ is $(\mathcal{K}_{\text{bad}}, \epsilon_{\text{wuniv}})$-weak-block-wise universal DbH function, if for any q-tuple of distinct messages $(M_1, \ldots, M_q)$, each of length at most ℓ blocks such that,

$$\forall i, j, k \text{ such that } i \neq j, i \neq k, \ \Pr[\mathsf{WUNIV}_{ij} \text{ holds}, K_h \in \mathcal{K}_h \setminus \mathcal{K}_{\text{bad}}] \leq \epsilon_{\text{wuniv}}.$$

We call ϵ_{wuniv} to be the *weak-block-wise universal advantage for two messages* of the DbH function H.

The prior two events perfectly capture all the possibilities of having non-freshness condition in $(\widetilde{g}, \widetilde{h})$ tuple except the condition that g_i and h_i can collide at the same value. In the following, we define the event that captures the collision of g_i and h_i at the same value.

For a q, a distinct tuple $(M_1, \ldots, M_q)$ and a fixed $i \in [q]$, we define the following event:

$$\mathsf{COLL}_i := \Big(H_{K_h,0}(M_i) = H_{K_h,1}(M_i)\Big).$$

Definition 5. We say that a DbH function $H : \mathcal{K}_h \times \mathcal{M} \to (\{0,1\}^n)^2$ is a $(\mathcal{K}_{\text{bad}}, \epsilon_{\text{coll}})$-colliding DbH function, if for any q tuple of distinct messages $(M_1, \ldots, M_q)$, each of length at most ℓ blocks such that,

$$\forall i \in [q], \;\; \Pr[\mathsf{COLL}_i \text{ holds}, K_h \in \mathcal{K}_h \setminus \mathcal{K}_{\text{bad}}] \leq \epsilon_{\text{coll}}.$$

We call ϵ_{coll} to be the *maximum collision probability* of DbH function H.

Discussion. (1) We would like to point out that ϵ_{cf}, ϵ_{univ} and ϵ_{coll} are functions of ℓ only as these values depend only on the length of a triplet of messages, pair of messages and a single message respectively. To emphasize this fact, we often write $\epsilon_{\text{cf}}(3, \ell)$ and $\epsilon_{\text{univ}}(2, \ell)$ to denote ϵ_{cf} and ϵ_{univ} respectively. In the same line of reasoning $\epsilon_{\text{coll}}(1, \ell)$ should also denote ϵ_{coll}, but we prefer to denote it as ϵ_{coll}.

(2) The notion of weak-cover-free advantage and weak-block-wise universal advantage are the required properties for the DbH function of the three-keyed DbHtS construction. Because, in the three-keyed DbHtS construction, we apply the two-keyed sum function on the input of the underlying DbH function. As a result, we do not require to bother about considering any cross-collisions in the output of the hash function. Here we intend to use the notion weak in terms of the security definition. If a double-block hash function is cover-free or (block-wise universal) then it is also weak-cover-free or (weak-block-wise universal respectively), however the converse is not necessarily true.

(3) The notion for cover-free and collision have also been used in the context of the NI$^+$-MAC [DNP16] security proof. However, the notion of cover-free and collision used in their paper is substantially different from ours: (i) In [DNP16], the cover-free notion was used to refer to the collision event between the input of the final function call with the input of an intermediate function call and (ii) the collision event was used to denote the input collision in the final function call.

3.3.2 Security Definitions for Block-Separated DbH Function.

A DbH function $H_{K_h} = (H_{K_h,0}, H_{K_h,1})$ is said to be *block-separated* if the range of possible values of $H_{K_h,0}$ and $H_{K_h,1}$ are disjoint. It is easy to see that using fix0 and fix1 functions, one can easily transform any DbH function H_{K_h} to a block-separated DbH function H'_{K_h} as follows:

$$H'_{K_h} := (\mathsf{fix0}(H_{K_h,0}), \mathsf{fix1}(H_{K_h,1})).$$

Note that, for a block-separated DbH function H_{K_h}, $(\widetilde{H}_{K_h,0}, \widetilde{H}_{K_h,1})$ is covered at an index $i \in [q]$ implies that only condition (i) or (vi) holds i.e., one of the following conditions hold:

- $\exists i \neq j, i \neq k$ such that $H_{K_h,0}(M_i) = H_{K_h,0}(M_j), H_{K_h,1}(M_i) = H_{K_h,1}(M_k)$.
- $\exists i \neq j$ such that $H_{K_h,0}(M_i) = H_{K_h,0}(M_j), H_{K_h,1}(M_i) = H_{K_h,1}(M_j)$.

Accordingly, for a block-separated DbH function, the event CF_{ijk} will be

$$\begin{aligned}\mathsf{CF}_{ijk} &= \Big(H_{K_h,0}(M_i) = H_{K_h,0}(M_j), H_{K_h,1}(M_i) = H_{K_h,1}(M_k)\Big) \\ &\quad \bigvee \Big(H_{K_h,0}(M_i) = H_{K_h,0}(M_j), H_{K_h,1}(M_i) = H_{K_h,1}(M_j)\Big). \end{aligned} \tag{3}$$

Note that, the above event is exactly identical to WCF_{ijk} and therefore, the cover-free notion of a block-separated DbH function is equivalent to its weak-cover-free notion. Therefore, the cover-free advantage of a block-separated DbH function is equivalent to its weak-cover-free advantage. Similarly, for a block-separated DbH function H_{K_h}, the block-wise universal advantage implies one of the following conditions hold:

- $\exists i \neq j$ such that $H_{K_h,0}(M_i) = H_{K_h,0}(M_j)$.
- $\exists i \neq j$ such that $H_{K_h,1}(M_i) = H_{K_h,1}(M_j)$.

Accordingly, for a block-separated DbH function, the event UNIV_{ij} will be

$$\mathsf{UNIV}_{ij} = \Big(H_{K_h,0}(M_i) = H_{K_h,0}(M_j)\Big) \bigvee \Big(H_{K_h,1}(M_i) = H_{K_h,1}(M_j)\Big). \tag{4}$$

Similar as before, the above event UNIV_{ij} is exactly identical to $\mathsf{WUNIV}ij$ and therefore, the block-wise universal notion of a block-separated DbH function is equivalent to its weak-block-wise universal notion. Therefore, the block-wise universal advantage of a block-separated DbH function is equivalent to its weak-block-wise universal advantage. Moreover, it is easy to see that for a block-separated DbH function, COLL_i is an impossible event for any $i \in [q]$ and hence a block separated DbH H is always $(\mathcal{K}_{\rm bad}, 0)$-colliding DbH function.

3.4 Security of DbHtS

In this section, we state and prove the PRF security of DbHtS construction. In particular, we prove only the PRF security of two-keyed DbHtS construction $\mathsf{C}_2[H, E]$ based on a DbH function H and pseudo random permutation or block cipher E.

Theorem 2. *Let $\mathcal{K}, \mathcal{K}_h$ and $\mathcal{M}$ be three non-empty finite sets. Let $E : \mathcal{K}\times\{0,1\}^n \to \{0,1\}^n$ be a block cipher and $H : \mathcal{K}_h \times \mathcal{M} \to (\{0,1\}^n)^2$ be a DbH function. Let $\mathcal{K}_{\rm bad}$ be a function from a set of tuple of q distinct messages to the power set of hash keys $\mathcal{P}(\mathcal{K}_h)$ such that for any tuple of q distinct messages $(M_1, \ldots, M_q)$, one has $\Pr[K \leftarrow\!\!\$\, \mathcal{K}_h : K \in \mathcal{K}_{\mathsf{bad}}(M_1, \ldots, M_q)] \leq \epsilon_{\rm bh}$.*
(i) If H is $(\mathcal{K}_{\rm bad}, \epsilon_{\rm cf}(3,\ell))$ cover-free, $(\mathcal{K}_{\rm bad}, \epsilon_{\rm univ}(2,\ell))$ block-wise universal and $(\mathcal{K}_{\rm bad}, \epsilon_{\rm coll})$ colliding hash function, then

$$\mathbf{Adv}^{\rm prf}_{\mathsf{C}_2[H,E]}(q,\ell,t) \leq \mathbf{Adv}^{\rm prp}_{E}(2q,t') + \epsilon_{\rm bh} + q\cdot\epsilon_{\rm coll} + \frac{q^3}{6}\cdot\epsilon_{\rm cf}(3,\ell) + \frac{3q^3}{2^n}\cdot\epsilon_{\rm univ}(2,\ell) + \frac{6q^3}{2^{2n}} + \frac{q}{2^n},$$

where $t' = t + q(t_h + t_\gamma)$, t_h be the time complexity of hash computation for a single message, t_γ be the time complexity of making two primitive queries with xoring their reply and we have assumed that $q \leq 2^{n-2}$.
(ii) If H is $(\mathcal{K}_{\rm bad}, \epsilon_{\rm cf}(3,\ell))$ cover-free and $(\mathcal{K}_{\rm bad}, \epsilon_{\rm univ}(2,\ell))$ block-wise universal block-separated DbH function, then

$$\mathbf{Adv}^{\rm prf}_{\mathsf{C}_2[H,E]}(q,\ell,t) \leq \mathbf{Adv}^{\rm prp}_{E}(2q,t') + \epsilon_{\rm bh} + \frac{q^3}{6}\cdot\epsilon_{\rm cf}(3,\ell) + \frac{3q^3}{2^n}\cdot\epsilon_{\rm univ}(2,\ell) + \frac{6q^3}{2^{2n}} + \frac{q}{2^n},$$

where $t' = t + q(t_h + t_\gamma)$, t_h be the time complexity of hash computation for a single message, t_γ be the time complexity of making two primitive queries with xoring their reply and we have assumed that $q \leq 2^{n-2}$.

(iii) If H is $(\mathcal{K}_{\text{bad}}, \epsilon_{\text{wcf}}(3,\ell))$ weak-cover-free and $(\mathcal{K}_{\text{bad}}, \epsilon_{\text{wuniv}}(2,\ell))$ weak-block-wise universal hash function, then

$$\mathbf{Adv}^{\text{prf}}_{\mathsf{C}_3[H,E]}(q,\ell,t) \leq 2\mathbf{Adv}^{\text{prp}}_{E}(2q,t') + \epsilon_{\text{bh}} + \frac{q^3}{6}\cdot\epsilon_{\text{wcf}}(3,\ell) + \frac{3q^3}{2^n}\cdot\epsilon_{\text{wuniv}}(2,\ell) + \frac{2q^3}{2^{2n}},$$

where $t' = t + q(t_h + t_\gamma)$, t_h be the time complexity of hash computation for a single message, t_γ be the time complexity of making two primitive queries with xoring their reply and we have assumed that $q \leq 2^{n-2}$.

Proof of part (i). Using the standard argument of switching from computational setting to information theoretic setting, we analyze the security of the construction $\mathsf{C}_2^* := \mathsf{C}_2^*[H,\Pi]$ based on an n-bit random permutation Π and a double block hash function H. This conversion adds the term $\mathbf{Adv}^{\text{prp}}_{E}(2q,t')$ in the security bound. Therefore, we need to show that

$$\mathbf{Adv}^{\text{prf}}_{\mathsf{C}_2^*}(q,\ell) \leq \epsilon_{\text{bh}} + q\cdot\epsilon_{\text{coll}} + \frac{q^3}{6}\cdot\epsilon_{\text{cf}}(3,\ell) + \frac{3q^3}{2^n}\cdot\epsilon_{\text{univ}}(2,\ell) + \frac{6q^3}{2^{2n}} + \frac{q}{2^n}. \tag{5}$$

The remainder of the proof is organized as follows: We begin with describing the ideal oracle and the attack transcript of the adversary in Sect. 3.4.1. In Sect. 3.4.2, we define and bound the probability of bad transcripts in the ideal oracle. Analysis of the good transcripts is shown in Sect. 3.4.3. Finally, part (i) of Theorem 2 follows from Theorem 1 and Eqn. (5) above and Lemma 3 and Lemma 4 proven below.

3.4.1 Initial Setup

We fix a computationally unbounded and hence deterministic non-repeating query making distinguisher D that interacts with either (1) a real oracle $\mathsf{C}_2^*[H_{K_h},\Pi]$ for a random permutation Π and a random hashing key K_h or (2) an ideal oracle \$, making at most q queries adaptively to the oracle.

Description of the Ideal Oracle. The ideal oracle consists of two phases: (i) Online Phase : In this phase, for each queried message M_i, the oracle samples the response T_i uniformly at random from $\{0,1\}^n$ and returns it to the distinguisher D. When all the queries and responses are over, the oracle samples a dummy hash key K_h from the hash key space $\mathcal{K}_h$, uniformly and independently to all the previously sampled random variables. If the sampled hash key happens to fall in the set of bad hash keys $\mathcal{K}_h$ (note that the message tuple is fixed and thus we can talk about the set $\mathcal{K}_h$), then it aborts the game (see line 2 of Fig. 3.2), otherwise the oracle computes the hash value for all the q queried messages. During this hash computation, if for any message M_i, one block of the hash value collides with another block, then Coll is set to $\mathtt{1}$ and the game will be aborted (see line 5 of Fig. 3.2).
Otherwise the oracle checks if any index $i \in [q]$ has been covered or not. If covered, then Cover is set to $\mathtt{1}$ and the oracle aborts the game (see line 6 of Fig. 3.2); otherwise it continues.

If the game does not abort, that means there is a non-empty set of free indices $\mathcal{F}$ for which both blocks of the hash value are fresh in the tuple of $2q$ many hash blocks value. Then, the oracle samples the outputs for these fresh hash values in without replacement manner such that for any $i \in \mathcal{F}$, the sampled output $Z_{0,i}$ and $Z_{1,i}$ sums up to T_i, where T_i has already been sampled in the online phase of the game (see line 8 of Fig. 3.2).

Now the remaining cases are those where exactly one block of the hash value collides. For all $i \in [q] \setminus \mathcal{F}$, if the output of the colliding hash value, say $H_{K_h,0}(M_i)$, has not been sampled yet, then the oracle samples its output in without replacement manner, say $Z_{0,i}$, and sets the output of the remaining block, i.e., output of $H_{K_h,1}(M_i)$ as the sum of $Z_{0,i}$

ONLINE PHASE OF $\mathcal{O}_{\text{ideal}}$

$\forall i \in [q]:$ On i-th query M_i, **return** $T_i \leftarrow\!\!\$\ \{0,1\}^n$;

1 : **if** $\exists i:\ T_i = \mathbf{0}$ **then** $\boxed{\mathsf{ZeroT} \leftarrow 1}$, $\perp$;

OFFLINE PHASE OF $\mathcal{O}_{\text{ideal}}$, initialize $\mathcal{L} = \emptyset$

1 : $K_h \leftarrow\!\!\$\ \mathcal{K}_h$;

2 : **if** $K_h \in \mathcal{K}_{\text{bad}}$, **then** $\boxed{\mathsf{Bad\text{-}Hash} \leftarrow 1,}$ $\perp$;

3 : $\forall i \in [q]:\ \ (H_{K_h,0}(M_i), H_{K_h,1}(M_i)) \leftarrow H_{K_h}(M_i)$;

4 : $\widetilde{H}_0 := (H_{K_h,0}(M_1), \ldots, H_{K_h,0}(M_q))$, $\widetilde{H}_1 := (H_{K_h,1}(M_1), \ldots, H_{K_h,1}(M_q))$;

5 : **if** $\exists i \in [q] : H_{K_h,0}(M_i) = H_{K_h,1}(M_i)$ **then** $\boxed{\mathsf{Coll} \leftarrow 1,}$ $\perp$;

6 : **if** $(\widetilde{H}_0, \widetilde{H}_1)$ is not a cover-free tuple **then** $\boxed{\mathsf{Cover} \leftarrow 1,}$ $\perp$;

7 : $\mathcal{F} := \{i \in [q] : H_{K_h,0}(M_i)$ and $H_{K_h,1}(M_i)$ both are fresh in $(\widetilde{H}_0, \widetilde{H}_1)\}$; $f = |\mathcal{F}|$;

8 : $(Z_{0,i}, Z_{1,i})_{i\in\mathcal{F}} \leftarrow\!\!\$\ \mathcal{S} := \{(Q_i, R_i)_{i\in\mathcal{F}} \in (\{0,1\}^n)^{(2f)} :\ Q_i \oplus R_i = T_i\ \forall i \in \mathcal{F}\}$;

9 : $\forall i \in [q] \cap \mathcal{F} : \Psi(H_{K_h,0}(M_i)) \leftarrow Z_{0,i}, \Psi(H_{K_h,1}(M_i)) \leftarrow Z_{1,i}$;

10 : $\forall i \in [q] \setminus \mathcal{F}:$ let $H_{K_h,b}(M_i)$ be not fresh in $(\widetilde{H}_0, \widetilde{H}_1)$, $b \in \{0,1\}$;

11 : **if** $H_{K_h,b}(M_i) \notin Dom(\Psi)$ **then** $\Psi(H_{K_h,b}(M_i)) \leftarrow Z_{b,i} \leftarrow\!\!\$\ \{0,1\}^n \setminus Ran(\Psi), Z_{1-b,i} \leftarrow T_i \oplus Z_{b,i}$;

12 : **else** $Z_{b,i} \leftarrow \Psi(H_{K_h,b}(M_i))$ and $Z_{1-b,i} \leftarrow T_i \oplus Z_{b,i}$;

13 : **if** $Z_{1-b,i} \in Ran(\Psi)$ **then** $\boxed{\mathsf{RC} \leftarrow 1}, \perp$;

14 : $\Psi(H_{K_h,1-b}(M_i)) \leftarrow Z_{1-b,i}$;

15 : **return** $(K_h, \widetilde{Z}_0, \widetilde{Z}_1)$;

Figure 3.2: Ideal oracle \$: Boxed statements denote bad events. Whenever a bad event is set to 1, the ideal oracle immediately aborts (denoted as $\perp$) and returns the remaining values of the transcript in any arbitrary manner. So, if the game aborts for some bad event, then we can surely assume that its previous bad events have not happened. Line 10 indicates that there exists some j for which $H_{K_h,b}(M_i) = H_{K_h,b}(M_j)$ or $H_{K_h,b}(M_i) = H_{K_h,1-b}(M_j)$ and line 11 indicates that permutation output of $H_{K_h,b}(M_i)$ is not defined yet. $\widetilde{Z}_0$ denotes the tuple $(Z_{0,1}, Z_{0,2}, \ldots, Z_{0,q})$ and $\widetilde{Z}_1$ denotes the tuple $(Z_{1,1}, Z_{1,2}, \ldots, Z_{1,q})$.

and T_i (see line 11 of Fig. 3.2). Otherwise, the oracle sets the output of $H_{K_h,0}(M_i)$ to the already defined element and adjusts the output of the other block accordingly (see line 12 of Fig. 3.2). Note that in the latter case, the oracle does not sample the output.

In the above said adjustment, if the output of $H_{K_h,1}(M_i)$ happens to collide with any previously sampled output, then RC is set to 1 (see line 13 of Fig. 3.2) and aborts the game. Note that, this event cannot hold for the real oracle, as $H_{K_h,1}(M_i)$ is fresh in the tuple of $2q$ many hash block values. Finally, it returns all these sampled values along with the sampled hash key to the distinguisher D.

Description of Attack Transcript. Let $\tau = \Big((M_1, T_1), (M_2, T_2), \ldots, (M_q, T_q)\Big)$ be the list of queries and responses of D which constitutes the query transcript of the attack. For convenience, we slightly modify the experiment where we reveal some more information to the distinguisher D in addition to the queries and responses only after D made all its queries and responses but before it output its decision. Therefore, the transcript of D essentially consists of all the internal values which are obtained while computing C_2^* for all

q queries. All in all, the transcript of the attack is

$$\tau = \Big((M_1, T_1, Z_{0,1}, Z_{1,1}), (M_2, T_2, Z_{0,2}, Z_{1,2}), \ldots, (M_q, T_q, Z_{0,q}, Z_{1,q}), K_h\Big).$$

In case of D interacting with the real oracle, we release the hash key K_h and the values

$$\forall i \in [q], Z_{0,i} := \Pi(H_{K_h,0}(M_i)) \;\text{ and }\; Z_{1,i} := \Pi(H_{K_h,1}(M_i)),$$

to D where $H_{K_h}(M_i) = (H_{K_h,0}(M_i), H_{K_h,1}(M_i))$.
Note that a transcript τ in the real oracle must satisfy all of the following:

1. $Z_{0,i} \oplus Z_{1,i} = T_i$ for all $i \in [q]$ and
2. the $2q$-tuples of input and output blocks of Π, namely $\mathsf{I} := (\widetilde{H}_0, \widetilde{H}_1)$ and $\mathsf{O} := (\widetilde{Z}_0, \widetilde{Z}_1)$ are permutation compatible. [4] Note that, I is uniquely determined by the message tuples $(M_1, \ldots, M_q)$ and the hash key K_h.

Recall that X_{re} and X_{id} are the probability distributions for the transcript τ induced by the real and the ideal oracle respectively. τ is attainable if $\Pr[X_{\text{id}} = \tau] > 0$ and let Θ denotes the set of all attainable transcripts.

3.4.2 Definition and Probability of Bad Transcripts

An attainable transcript τ is said to be bad if either of the following bad flags

$$\mathsf{ZeroT}, \mathsf{Bad\text{-}Hash}, \mathsf{Coll}, \mathsf{Cover}, \mathsf{RC}$$

is set to $\mathtt{1}$ as defined in Fig. 3.2. We define the event

$$\mathsf{Bad} := \mathsf{ZeroT} \vee \mathsf{Bad\text{-}Hash} \vee \mathsf{Coll} \vee \mathsf{Cover} \vee \mathsf{RC}.$$

Let Θ_b denote the set of all bad transcripts and $\Theta_g = \Theta \setminus \Theta_b$ be the set of all good transcripts. Having identified the set of all bad transcripts, we bound the probability of realizing the bad transcript in the ideal oracle in the following lemma:

Lemma 3. *Let X_{id} and Θ_b be defined as above then,*

$$\epsilon_{\text{bad}} := \Pr[X_{\text{id}} \in \Theta_b] \leq \epsilon_{\text{bh}} + q \cdot \epsilon_{\text{coll}} + \frac{q^3}{6} \cdot \epsilon_{\text{cf}}(3, \ell) + \frac{3q^3}{2^n} \cdot \epsilon_{\text{univ}}(2, \ell) + \frac{q}{2^n}. \tag{6}$$

Proof. Bounding the probability of the bad transcripts in the ideal oracle is equivalent to bounding the probability of the event Bad in the ideal oracle. Using the union bound we have,

$$\begin{aligned} \Pr[\mathsf{Bad}] \;&\leq\; \Pr[\mathsf{ZeroT}] + \Pr[\mathsf{Bad\text{-}Hash}] + \Pr[\mathsf{Coll} \wedge \overline{\mathsf{Bad\text{-}Hash}}] \\ &+\; \Pr[\mathsf{Cover} \wedge \overline{\mathsf{Bad\text{-}Hash}}] + \Pr[\mathsf{RC} \wedge \overline{\mathsf{Bad\text{-}Hash}}]. \end{aligned} \tag{7}$$

In the following, we separately bound each of the above terms.

Bounding ZeroT. The bad flag ZeroT is set to $\mathtt{1}$, if out of q responses, there exists at least one response T_i such that $T_i = \mathbf{0}$, i.e.

$$\Pr[\mathsf{ZeroT}] = \Pr[\vee_{i=1}^{q} T_i = \mathbf{0}] \leq \sum_{i=1}^{q} \Pr[T_i = \mathbf{0}] = \frac{q}{2^n}. \tag{8}$$

[4] For two block tuples $\widetilde{x}$ and $\widetilde{y}$ having equal length over the same index set, we say $\widetilde{x}$ is permutation-compatible with $\widetilde{y}$, if there exists a permutation π such that $\forall i, \pi(x_i) = y_i$.

Last equality follows due to the uniform and independent sampling of the responses in the ideal oracle.

Bounding Bad-Hash. This probability is basically determined from the probability of sampling the hash key of the underlying DbH function. Therefore,

$$\Pr[\mathsf{Bad\text{-}Hash}] \leq \epsilon_{\mathrm{bh}}. \tag{9}$$

Bounding Coll $\wedge$ $\overline{\text{Bad-Hash}}$. From the definition of the ideal oracle game, Coll is set to 1 if there exists at least one $i \in [q]$ such that $H_{K_h,0}(M_i) = H_{K_h,1}(M_i)$ and $K_h \in \mathcal{K}_h \setminus \mathcal{K}_{\mathrm{bad}}$. Therefore,

$$\begin{aligned}\Pr[\mathsf{Coll} \wedge \overline{\mathsf{Bad\text{-}Hash}}] &\leq \sum_{i=1}^{q} \Pr[H_{K_h,0}(M_i) = H_{K_h,1}(M_i),\ K_h \in \mathcal{K}_h \setminus \mathcal{K}_{\mathrm{bad}}] \\ &= \sum_{i=1}^{q} \Pr[\mathsf{COLL}_i \text{ holds}, K_h \in \mathcal{K}_h \setminus \mathcal{K}_{\mathrm{bad}}] \overset{(1)}{\leq} q \cdot \epsilon_{\mathrm{coll}}, \end{aligned} \tag{10}$$

where (1) follows from Definition 5.

Bounding Cover $\wedge$ $\overline{\text{Bad-Hash}}$. From the definition of the ideal oracle game, Cover is set to 1 if the tuple $(\widetilde{H}_0, \widetilde{H}_1)$ is not cover-free where the sampled hash key K_h belongs to the set $\mathcal{K}_h \setminus \mathcal{K}_{\mathrm{bad}}$. Moreover, Cover event set to 1 implies Coll event has not occurred. Therefore,

$$\begin{aligned}\Pr[\mathsf{Cover} \wedge \overline{\mathsf{Bad\text{-}Hash}}] &= \Pr[\big((H_{K_h,0}(M_i))_i,\ (H_{K_h,1}(M_i))_i\big) \text{ is covered}, K_h \in \mathcal{K}_h \setminus \mathcal{K}_{\mathrm{bad}}] \\ &\leq \sum_{i\neq j, i\neq k} \Pr[\mathsf{CF}_{ijk} \text{ holds}, K_h \in \mathcal{K}_h \setminus \mathcal{K}_{\mathrm{bad}}] \overset{(1)}{\leq} \frac{q^3}{6} \cdot \epsilon_{\mathrm{cf}}(3, \ell), \end{aligned} \tag{11}$$

where (1) follows from Definition 1.

Bounding RC $\wedge$ $\overline{\text{Bad-Hash}}$. The event holds when for some $b \in \{0,1\}$ and $i \in [q]$, $H_{K_h,b}(M_i)$ is not fresh in $\Big((H_{K_h,0}(M_1), \ldots, H_{K_h,0}(M_q)), (H_{K_h,1}(M_1), \ldots, H_{K_h,1}(M_q))\Big)$ and $Z_{1-b,i} \in Ran(\Psi)$ (see line 12-13 of Fig. 3.2). Observe that the event considers undesired collision among range elements. This bad event will occur if for some i, j, k, b, b', u with $i < j$, $i \neq k$ and $b, b', u \in \{0,1\}$, we have: (1) $H_{K_h,b}(M_i) = H_{K_h,b'}(M_j)$ and (2) $Z_{b,i} \oplus T_i = Z_{u,k}$ where $Z_{b,i} \xleftarrow{\$} \{0,1\}^n \setminus Ran(\Psi)$. Now, we split this bad event into the following cases and compute the probabilities for these cases individually:

- **Case A.** $j \neq k$**.** Since the first condition is an event of the sampling of hash key K_h and the second one is the event of lazy sampling (independent of the distribution of the hash key K_h), the probability of the bad event for this case for a specific choice of i, j, k, b, b', u would be

$$\begin{aligned}\mathsf{P} &:= \Pr[H_{K_h,b}(M_i) = H_{K_h,b'}(M_j), K_h \in \mathcal{K}_h \setminus \mathcal{K}_{\mathrm{bad}}] \times \Pr[Z_{b,i} = T_i \oplus Z_{u,k}] \\ &= \Pr[\mathsf{UNIV}_{ij} \text{ holds}, K_h \in \mathcal{K}_h \setminus \mathcal{K}_{\mathrm{bad}}] \times \Pr[Z_{b,i} = T_i \oplus Z_{u,k}] \\ &\overset{(1)}{\leq} \epsilon_{\mathrm{univ}}(2, \ell) \times \frac{1}{2^n - 2q}, \end{aligned}$$

  where (1) follows from Definition 3. By summing over all possible choices of i, j, k, b, b', u, the probability of bad event for this case would be bounded above by $2q(q-1)(q-2) \cdot \epsilon_{\mathrm{univ}}(2, \ell)/2^n$, with the assumption that $q \leq 2^{n-2}$.

- **Case B.** $j = k$**.** Here we sample $Z_{b,i} \xleftarrow{\$} \{0,1\}^n \setminus Ran(\Psi)$ first and then set $Z_{b',j}$ to $Z_{b,i}$. Now, we analyse this case in different sub cases:

– **Case B.1.** $u = b'$**.** We first consider the case when $u = b'$. In this case, $H_{K_h,b}(M_i) = H_{K_h,b'}(M_j)$ and $Z_{b,i} \oplus T_i = Z_{b',j}$, But these two events implies $T_i = \mathbf{0}$ which is impossible and hence the probability, denoted by $\mathsf{P}_{\mathrm{B.1}}$, becomes zero.

– **Case B.2.** $u \neq b'$ **and** $b = b'$**.** This case eventually boils down to the joint event (i) $H_{K_h,b}(M_i) = H_{K_h,b}(M_j)$ and (ii) $Z_{b,i} \oplus T_i = Z_{1-b,j}$. Note that, if the event $T_i = T_j$ holds, then condition (ii) is implied by condition (i) and the constraint $T_i = T_j$. Therefore, bounding the joint probability of $H_{K_h,b}(M_i) = H_{K_h,b}(M_j)$ and $Z_{b,i} \oplus T_i = Z_{1-b,j}$ is equivalent to bounding the joint probability of $H_{K_h,b}(M_i) = H_{K_h,b}(M_j)$ and $T_i = T_j$. Now, to bound the later one, we have

$$\begin{aligned}
\mathsf{P}_{\mathrm{B.2}} &:= \Pr[T_i = T_j, H_{K_h,b}(M_i) = H_{K_h,b}(M_j), K_h \in \mathcal{K}_h \setminus \mathcal{K}_{\mathrm{bad}}] \\
&= \Pr[H_{K_h,b}(M_i) = H_{K_h,b}(M_j), K_h \in \mathcal{K}_h \setminus \mathcal{K}_{\mathrm{bad}} | T_i = T_j] \cdot \Pr[T_i = T_j] \\
&= \Pr[\mathsf{UNIV}_{ij} \text{ holds}, K_h \in \mathcal{K}_h \setminus \mathcal{K}_{\mathrm{bad}} | T_i = T_j] \cdot \Pr[T_i = T_j] \\
&\overset{(1)}{\leq} \epsilon_{\mathrm{univ}}(2,\ell) \times \frac{1}{2^n}, \qquad (12)
\end{aligned}$$

where (1) follows from Definition 3 and Eqn. (12) follows from the argument that after conditioning $T_1, \ldots, T_q$, all the q messages would be fixed and hence, the conditional probability of $\mathsf{UNIV}_{ij}, K_h \in \mathcal{K}_h \setminus \mathcal{K}_{\mathrm{bad}}$ is at most $\epsilon_{\mathrm{univ}}(2,\ell)$. Moreover, the event $T_j = T_i$ is bounded by $\frac{1}{2^n}$. On the other hand, if $T_i \neq T_j$ then $Z_{b,j} \oplus T_j \neq Z_{b,i} \oplus T_i$ and hence the probability becomes zero.

– **Case B.3.** $u \neq b'$ **and** $b \neq b'$**.** This case eventually boils down to the joint event (i) $H_{K_h,b}(M_i) = H_{K_h,1-b}(M_j)$ and (ii) $Z_{b,i} \oplus T_i = Z_{b,j}$. Note that, $i \neq j$ as that would leads to $T_i = \mathbf{0}$ and hence the probability becomes zero. Now, if the event $T_i = T_j$ holds, then as before condition (ii) is implied by condition (i) and the constraint $T_i = T_j$. Using the previous argument we have

$$\mathsf{P}_{\mathrm{B.3}} := \Pr[T_i = T_j, H_{K_h,b}(M_i) = H_{K_h,1-b}(M_j), K_h \in \mathcal{K} \setminus \mathcal{K}_{\mathrm{bad}}] \overset{(1)}{\leq} \epsilon_{\mathrm{univ}}(2,\ell) \times \frac{1}{2^n},$$

where (1) follows from Definition 3. Moreover, if $T_i \neq T_j$ then $Z_{1-b,j} \oplus T_j \neq Z_{b,i} \oplus T_i$ and hence the probability in that case becomes zero.

By summing over all (i, j, b, b', u) with $i < j$ and $b, b', u \in \{0, 1\}$, the probability for case B, denoted by P_{B}, is bounded above by taking the maximum of $\mathsf{P}_{\mathrm{B.1}}, \mathsf{P}_{\mathrm{B.2}}$ and $\mathsf{P}_{\mathrm{B.3}}$, which is upper bounded by $\frac{q(q-1)\cdot\epsilon_{\mathrm{univ}}(2,\ell)}{2^n}$.

Therefore, we have

$$\begin{aligned}
\Pr[\mathsf{RC} \wedge \overline{\mathsf{Bad\text{-}Hash}}] &\leq \mathsf{P}_{\mathrm{A}} + \mathsf{P}_{\mathrm{B}} \\
&\leq \frac{q(q-1) \cdot \epsilon_{\mathrm{univ}}(2,\ell)}{2^n} + \frac{2q(q-1)(q-2) \cdot \epsilon_{\mathrm{univ}}(2,\ell)}{2^n} \\
&\leq \frac{3q^3}{2^n} \cdot \epsilon_{\mathrm{univ}}(2,\ell). \qquad (13)
\end{aligned}$$

Finally, the result follows from Eqn. (7), Eqn. (8), Eqn. (9), Eqn. (10), Eqn. (11) and Eqn. (13). □

3.4.3 Analysis of Good Transcripts

In this section, we lower bound the ratio of the probability of realizing a good transcript τ in the real and the ideal oracle. For this, let us first understand what does a good transcript in the ideal oracle mean. Note that, for each $i \in \mathcal{F}$, both $H_{K_h,0}(M_i)$ and $H_{K_h,1}(M_i)$ are fresh in the concatenated tuple $(\widetilde{H}_0, \widetilde{H}_1)$, as shown in line 7 of Fig. 3.2. Moreover, as the transcript τ is good, Cover is not set to 1 and therefore, for every $i \notin \mathcal{F}$, exactly one of the $H_{K_h,0}(M_i)$ or $H_{K_h,1}(M_i)$ is fresh in $(\widetilde{H}_0, \widetilde{H}_1)$. Thus, we have exactly $(q+f)$ many fresh blocks ($2f$ many fresh blocks for all those indices belong to $\mathcal{F}$ and additionally we have $(2q-2f)/2$ many fresh blocks) and $q-f$ many non-fresh blocks, where $f = |\mathcal{F}|$. Now, we define a relation $\sim$ on $\mathcal{F}^c := [q] \setminus \mathcal{F}$ as $i \sim j$ if $H_{K_h,b}(M_i) = H_{K_h,b'}(M_j)$ for $b, b' \in \{0,1\}$. Clearly, $\sim$ is an equivalence relation over $\mathcal{F}^c$ and hence partitions $\mathcal{F}^c$ as $\mathcal{C}_1 \sqcup \cdots \sqcup \mathcal{C}_r$. Note that each $\mathcal{C}_i$ contains at least two elements. Therefore, line 11 is executed only for one element of these $\mathcal{C}_j$'s. Let $c_j := \min \mathcal{C}_j$ be the minimum valued element of $\mathcal{C}_j$. So, when $i = c_j$ for some $j \in [r]$, we execute line 11 once for sampling the output of $H_{K_h,b}(M_i)$, which in turn determines the outputs for all $H_{K_h,b}(M_p)$, where $p \in \mathcal{C}_j$ and $b \in \{0,1\}$. Due to the definition of $Z_{0,i}, Z_{1,i}$ in line 8, 9, 11 and 12, for all $i \in [q]$ we have $Z_{0,i} \oplus Z_{1,i} = T_i$. As the event ZeroT does not hold, we also have $Z_{0,i} \neq Z_{1,i}$. Moreover, as τ is good, RC is not set to 1 and thus no range collision occurs for two different inputs. Thus, we have the following result:

Claim 1. *For a good transcript τ, $2q$-tuples of input and output blocks of Π, namely $\mathsf{I} := (\widetilde{H}_0, \widetilde{H}_1)$ and $\mathsf{O} := (\widetilde{Z}_0, \widetilde{Z}_1)$ are permutation compatible.*

We would like to mention here that the result of Claim 1 will be used to compute the ratio of real to ideal interpolation probability for a good transcript τ as follows:

Lemma 4. *Let τ be a good transcript. Then,*

$$\frac{\Pr[X_{\mathrm{re}} = \tau]}{\Pr[X_{\mathrm{id}} = \tau]} \geq 1 - \frac{6q^3}{2^{2n}}.$$

Proof. As τ is a good transcript, the pair of tuple $(\widetilde{H}_0, \widetilde{H}_1)$ is cover-free. We have considered the set $\mathcal{F}$, the set of all free indices, as defined in line 7 of Fig. 3.2 and let $f = |\mathcal{F}|$. We have also defined a set $\mathcal{S}$ in line 8 of Fig. 3.2. Recall that, Ψ is the list of responses of the lazy sampling made in the ideal game. Now,

$$\begin{aligned}
\Pr[X_{\mathrm{id}} = \tau] &= \frac{1}{|\mathcal{K}_h|} \cdot \frac{1}{2^{nq}} \cdot \Pr[\Psi(H_{K_h,0}(M_i)) = Z_{0,i}, \Psi(H_{K_h,1}(M_i)) = Z_{1,i}\ \forall i \in [q]] \\
&\overset{(1)}{=} \frac{1}{|\mathcal{K}_h|} \cdot \frac{1}{2^{nq}} \cdot \Pr[\underbrace{\Psi(H_{K_h,0}(M_i)) = Z_{0,i}, \Psi(H_{K_h,1}(M_i)) = Z_{1,i}\ \forall i \in \mathcal{F}}_{B}] \\
&\quad \cdot \Pr[\Psi(H_{K_h,0}(M_i)) = Z_{0,i}, \Psi(H_{K_h,1}(M_i)) = Z_{1,i}\ \forall i \in [q] \setminus \mathcal{F}\ |B] \\
&\overset{(2)}{=} \frac{1}{|\mathcal{K}_h|} \cdot \frac{1}{2^{nq}} \cdot \frac{1}{|\mathcal{S}|} \cdot \frac{1}{(2^n - 2f)_r}, \qquad (14)
\end{aligned}$$

where r denotes the number of equivalence classes $\mathcal{C}_i$. First, we use the fact that the hash key K_h, the response tuple $\widetilde{T}$ and the lazy sampling of Ψ are jointly independent as each T_i is distributed independent to (i) all the previously sampled T and (ii) the distribution of K_h and lazy sampling of Ψ, made in the offline phase of the game. Moreover, the distribution of K_h is independent to the distribution of lazy sampling. For the last equality (2), we note that Ψ is defined in two stages: (i) in the first stage, it samples elements from $\mathcal{S}$ randomly for all the free indices $i \in \mathcal{F}$ (see line 8 of Fig. 3.2) and thus $\Pr[B] = |\mathcal{S}|^{-1}$ and then (ii) in the next stage, it defines the rest of Ψ values by the lazy sampling method as described in line 11-14. Note that, in the second stage of the sampling process, the oracle samples the permutation output for r many distinct values in such a manner that

these sampled output should not collide with the already sampled values in the first stage of the sampling process. Hence, we have

$$\Pr[\Psi(H_{K_h,0}(M_i)) = Z_{0,i}, \Psi(H_{K_h,1}(M_i)) = Z_{1,i}\ \forall i \in [q] \setminus \mathcal{F}\ |B] = \frac{1}{(2^n - 2f)_r}.$$

Computing Real Interpolation Probability. Now, we compute the real interpolation probability. From Claim 1, it is obvious that $(\widetilde{H}_0, \widetilde{H}_1)$ is permutation compatible with $(\widetilde{Z}_0, \widetilde{Z}_1)$. Note that, the number of permutation outputs that we need to sample is exactly $q + f + r$. This is because, we have all total $q + f$ many fresh hash blocks value and for each equivalent class, we need to additionally sample the output for a single hash block value. Hence,

$$\Pr[X_{\text{real}} = \tau] \quad = \quad \frac{1}{|\mathcal{K}_h|} \cdot \frac{1}{(2^n)_{q+f+r}} \tag{15}$$

Computing the Ratio. Now we compute the ratio of Eqn. (15) to Eqn. (14) as follows:

$$\begin{aligned}
\frac{\Pr[X_{\text{real}} = \tau]}{\Pr[X_{\text{id}} = \tau]} &= \frac{2^{nq} \cdot (2^n - 2f)_r \cdot |\mathcal{S}|}{(2^n)_{q+f+r}} \\
&\overset{(1)}{\geq} \frac{2^{nq} \cdot (2^n - 2f)_r \cdot (2^n)_{2f}}{(2^n)_{q+f+r} \cdot 2^{nf}} \cdot \left(1 - \frac{6f^3}{2^{2n}}\right) \\
&= \underbrace{\frac{2^{n(q-f)}}{(2^n - 2f + r)_{q-f}}}_{\geq 1} \cdot \left(1 - \frac{6f^3}{2^{2n}}\right) \overset{(2)}{\geq} \left(1 - \frac{6q^3}{2^{2n}}\right)
\end{aligned}$$

where (1) follows after substituting the lower bound of $|\mathcal{S}|$ from Lemma 2 and (2) follows as $f \leq q$. □

3.5 Proof of Theorem 2 part (ii)

Proof of the second part of Theorem 2, i.e., the proof of the PRF security of $\mathsf{C}_2[H, E]$ construction when H is a block-separated DbH function, easily follows from that of the first part of Theorem 2. All the bad events from the first part of the theorem will remain same, except that the bad flag Coll cannot be set to 1 as there cannot be any collision event for a block-separated DbH function and hence we have $\epsilon_{\text{coll}} = 0$. Moreover, the analysis for the ratio of the real to ideal interpolation probability for a good transcript τ remains identical to the proof of the first part of the theorem. For all $i \in \mathcal{F}$ (set of free indices), we regard $H_{K_h,0}(M_i) = U_i$ and $H_{K_h,1}(M_i) = V_i$. Since, H is a block-separated DbH function, $(U_1, U_2, \ldots, U_f, V_1, V_2, \ldots, V_f) \in (\{0,1\}^n)^{(2f)}$. Now, to sample the corresponding output tuple of $(U_1, U_2, \ldots, U_f, V_1, V_2, \ldots, V_f)$, we sample $(Z_{0,1}, Z_{0,2}, \ldots, Z_{0,f}, Z_{1,1}, Z_{1,2}, \ldots, Z_{1,f}) \in (\{0,1\}^n)^{(2f)}$ such that $Z_{0,i} \oplus Z_{1,i} = T_i, \forall i \in \mathcal{F}$, where $f = |\mathcal{F}|$. This equivalence allows us to apply Lemma 2 for bounding the ideal interpolation probability (as done in the proof of the first part of the theorem).

3.6 Proof of Theorem 2 part (iii)

There are subtle differences in the proof of the third part of the theorem from its first part which we list as follows:

(a) Unlike $\mathsf{C}_2[H, E]$, where we used the same permutation in the sum function, in this case we use two "*independent*" random permutations instead of two identical permutation. Use of independent permutations makes the significant differences in defining the bad

events. Firstly, (i) we no longer need to have the zero output restriction, i.e., $T_i = \mathbf{0}$ for $i \in [q]$ as the bad event. This is because, we do not care if $H_{K_h,0}(M_i)$ collides with $H_{K_h,1}(M_i)$ for any $i = 1, \ldots, q$, as the two permutations are independent. This condition also alleviates the necessity to consider the maximum collision probability of the DbH function. Hence, in this security bound, we do not have the $q/2^n$ term and the maximum collision probability term.

(b) For analysing the ratio of the real to ideal interpolation probability, we use the result of Lucks (see Theorem 5, [Luc00]) for lower bounding the number of solutions to the sum of two independent permutations problem.

Summarizing above, security result for three-keyed DbHtS follows.

3.7 Application of Theorem 2.

To prove the BBB security of a particular construction that follows DbHtS paradigm, one needs to show the followings:

(a) The cover-free advantage of its underlying DbH function for any triplet of distinct messages should be of the order of $O(\ell^c/2^{2n})$.

(b) The block-wise universal advantage of its underlying DbH function for any pair of distinct messages should be $O(\ell^c/2^n)$.

(c) The maximum collision probability of its underlying DbH function (wherever it is applicable) must be of the order of $O(\ell^c/2^n)$.

(d) Finally, the probability bound of the bad-hash-key must be of beyond birthday bound.

Here c is some small positive constant and ℓ is the maximum number of message blocks among all q queries.

Discussion. The importance of introducing the set of bad hash keys in the security statement lies in providing the improved security bound for different instantiations of the two-keyed and the three-keyed DbHtS construction. A more detailed explanation follows in Sect. 6.6.

Remark 4. Dodis et al. [DS11] have shown that if H is a cover-free DbH function and the sum function is instantiated with two independent n-bit keyed unforgeable functions, then $\mathsf{C}_3[H, F]$ is unforgeable. One can similarly show the PRF-security of the construction when the sum function is instantiated with two independent n-bit keyed functions. For the PRF security of $\mathsf{C}_3[H, F]$, if the output tuple of the underlying DbH function is cover-free, then the output of $\mathsf{C}_3[H, F]$ is perfectly random. Hence, the security of the construction boils down to the cover-free advantage of the underlying DbH.

4 Instantiation of DbHtS Using PolyHash

In this section, we instantiate the DbH function using the double-block PolyHash function, that results in a PolyHash based DbHtS construction. PolyHash [dB93, BJKS93, Tay93] is a very efficient algebraic hash function. To apply this on a message M, we first use apply an injective padding such as 10^* i.e., pad 1 followed by minimum number of zeros so that the total number of bits in the padded message becomes mutiple of n. Let the padded message be $M^* = M_1 \| M_2 \| \ldots M_l$, where l is the number of n-bit blocks in it. Then, we define the PolyHash as follows:

$$\mathsf{PH}_{K_h}(M) = M_l K_h \oplus M_{l-1} K_h^2 \oplus \ldots \oplus M_1 K_h^l.$$

Now, we define the following PolyHash-DbH function:

$$\mathsf{PH\text{-}DbH}_{K_h,K_h^\star}(M) := \Big(\mathsf{fix0}(\mathsf{PH}_{K_h}(M)), \mathsf{fix1}(\mathsf{PH}_{K_h^\star}(M))\Big),$$

where K_h and $K_h^\star$ are two independent hash keys. Note that, PH-DbH is a block-separated DbH function. By composing PH-DbH with the single-keyed sum function, we obtain the two-keyed PolyHash based DbHtS construction, which we denote as $\mathsf{C}_2[\mathsf{PH\text{-}DbH}, E]$. Similarly, by composing PH3-DbH with the double-keyed sum function, where

$$\mathsf{PH3\text{-}DbH}_{K_h,K_h^\star}(M) := \Big((\mathsf{PH}_{K_h}(M)), (\mathsf{PH}_{K_h^\star}(M))\Big),$$

we obtain the three-keyed PolyHash based DbHtS construction, which we denote as $\mathsf{C}_3[\mathsf{PH3\text{-}DbH}, E]$).

Bad Hash Key. For PolyHash based DbH function, we consider that the set of the bad hash keys is empty, i.e., $\mathcal{K}_{\text{bad}} = \Phi$ for both PH-DbH and PH3-DbH.

The following result shows that PH-DbH is a $(\Phi, 4\ell^2/2^{2n})$-cover-free and $(\Phi, 2\ell/2^n)$-block-wise universal block-separated DbH function. Moreover, PH3-DbH is a $(\Phi, \ell^2/2^{2n})$-weak-cover-free and $(\Phi, \ell/2^n)$-weak-block-wise universal DbH function.

Theorem 3. PH-DbH *is a* $(\Phi, 4\ell^2/2^{2n})$*-cover-free and* $(\Phi, 2\ell/2^n)$*-block-wise universal block-separated* DbH *function. Moreover,* PH3-DbH *is a* $(\Phi, \ell^2/2^{2n})$*-weak-cover-free and* $(\Phi, \ell/2^n)$*-weak-block-wise universal* DbH *function.*

We defer the proof of Theorem 3 to Sect. 4.3. Assuming that the theorem holds, we now prove the PRF security of $\mathsf{C}_2[\mathsf{PH\text{-}DbH}, E]$ and $\mathsf{C}_3[\mathsf{PH3\text{-}DbH}, E]$ in Sect. 4.1 and Sect. 4.2 respectively.

4.1 Implication for PolyHash Based Two-Keyed DbHtS

Recall that, PH-DbH is a block-separated DbH function. As the set of bad hash keys of PH-DbH is empty, we have $\epsilon_{\text{bh}} = 0$. Security result for $\mathsf{C}_2[\mathsf{PH\text{-}DbH}, E]$ is as follows:

Theorem 4. *Let* $\mathcal{K}_h, \mathcal{K}$ *and* $\mathcal{M}$ *be three non-empty finite sets. Let* $E : \mathcal{K}\times\{0,1\}^n \to \{0,1\}^n$ *be a block cipher and* $\mathsf{PH\text{-}DbH} : (\mathcal{K}_h \times \mathcal{K}_h) \times \mathcal{M} \to (\{0,1\}^n)^2$ *be a block separated* DbH *function. Assume that there is no set of bad hash keys. Then, any distinguisher with running time at most* t*, making* q *tuple of distinct messages each of at most* ℓ *blocks long, can distinguish* $\mathsf{C}_2[\mathsf{PH\text{-}DbH}, E]$ *from an* n*-bit uniform random function by*

$$\mathbf{Adv}^{\text{prf}}_{\mathsf{C}_2[\mathsf{PH\text{-}DbH},E]}(q,\ell,t) \le \mathbf{Adv}^{\text{prp}}_{E}(2q,t') + \frac{2q^3\ell^2}{3\cdot 2^{2n}} + \frac{6q^3\ell}{2^{2n}} + \frac{6q^3}{2^{2n}} + \frac{q}{2^n},$$

where $t' = t + q(t_h + t_\gamma)$*,* t_h *be the time complexity of* PH-DbH *computation for a single message and* t_γ *be the time complexity of making two primitive queries with xoring their reply.*

Proof of the theorem directly follows from $\epsilon_{\text{bh}} = 0$, Theorem 3 and part (ii) of Theorem 2.

4.2 Implication for PolyHash Based Three-Keyed DbHtS

Recall that, PH3-DbH is not a block-separated DbH function and for PH3-DbH, we have $\epsilon_{\text{bh}} = 0$ (as its set of bad hash keys is empty). The security result for $\mathsf{C}_3[\mathsf{PH3\text{-}DbH}, E]$ is as follows:

Theorem 5. *Let $\mathcal{K}_h, \mathcal{K}$ and $\mathcal{M}$ be three non-empty finite sets. Let $E : \mathcal{K}\times\{0,1\}^n \to \{0,1\}^n$ be a block cipher and $\mathsf{PH3\text{-}DbH} : (\mathcal{K}_h \times \mathcal{K}_h) \times \mathcal{M} \to (\{0,1\}^n)^2$ be a DbH function. Assume that there is no set of bad hash keys. Then, any distinguisher with running time at most t, making q tuple of distinct messages each of at most ℓ blocks long, can distinguish $\mathsf{C}_3[\mathsf{PH3\text{-}DbH}, E]$ from an n-bit uniform random function by*

$$\mathbf{Adv}^{\mathrm{prf}}_{\mathsf{C}_3[\mathsf{PH3\text{-}DbH},E]}(q,\ell,t) \le 2\mathbf{Adv}^{\mathrm{prp}}_{E}(q,t') + \frac{q^3\ell^2}{6\cdot 2^{2n}} + \frac{3q^3\ell}{2^{2n}} + \frac{2q^3}{2^{2n}}$$

where $t' = t + q(t_h + t_\gamma)$, t_h be the time complexity of $\mathsf{PH3\text{-}DbH}$ computation for a single message and t_γ be the time complexity of making two primitive queries with xoring their reply.

Proof of the theorem directly follows from $\epsilon_{\mathrm{bh}} = 0$, Theorem 3 and part (*iii*) of Theorem 2.

4.3 Proof of Theorem 3

In this section, we bound the cover-free and the block-wise universal advantage of $\mathsf{PH\text{-}DbH}$. We also bound the weak-cover-free and the weak-block-wise universal advantage of $\mathsf{PH3\text{-}DbH}$. Recall that, we have considered that there is an empty set of bad hash keys for both $\mathsf{PH\text{-}DbH}$ and $\mathsf{PH3\text{-}DbH}$. Therefore, for analyzing the cover-free and the block-wise universal advantage of $\mathsf{PH\text{-}DbH}$ and for analyzing the weak-cover-free and the weak-block-wise universal advantage of $\mathsf{PH3\text{-}DbH}$, we sample the hash key from the set of all hash keys and as a result we have

$$\Pr[\mathsf{Bad\text{-}Hash}] := \epsilon_{\mathrm{bh}} = 0.$$

Bounding Block-wise Universal advantage of $\mathsf{PH\text{-}DbH}$. It is a well known result that the (almost-xor) universal advantage of the PolyHash [dB93, BJKS93, Tay93] is about $\ell/2^n$, where ℓ is the maximum number of message blocks. One can trivially extend this result to show that the universal advantage of the one-bit chopped version of the PolyHash, i.e., $\mathsf{fixb}(\mathsf{PH}_{K_h}(M))$, is $2\ell/2^n$, where $\mathsf{b} \in \{0,1\}$.

For a fixed pair of messages M and M', where the maximum number of message blocks of M and M' is ℓ, and for any $\mathsf{b} \in \{0,1\}$, we denote the event $\mathsf{fixb}(\mathsf{PH}_{K_h}(M)) = \mathsf{fixb}(\mathsf{PH}_{K_h}(M'))$ by $\mathsf{P}_{K_h,\mathsf{b}}(M,M')$. Therefore, for a fixed $\mathsf{b} \in \{0,1\}$, we have

$$\Pr[\mathsf{P}_{K_h,\mathsf{b}}(M,M')] = \sum_{b\in\mathbf{0},\mathbf{1}} \Pr[\mathsf{PH}_{K_h}(M) = \mathsf{PH}_{K_h}(M') \oplus b] \le \frac{2\ell}{2^n}, \tag{16}$$

where the last inequality follows from the almost-xor universal advantage of the PolyHash. For brevity, let us denote $\Pr[\mathsf{UNIV}_{ij}$ holds, $(K_h, K_h^\star) \in \mathcal{K}_h \times \mathcal{K}_h]$ by $\mathsf{P}_{\mathrm{univ}}$. Therefore, we have

$$\begin{aligned}\mathsf{P}_{\mathrm{univ}} &\overset{(1)}{=} \max\left(\Pr[\mathsf{P}_{K_h,0}(M,M'), (K_h,K_h^\star) \in \mathcal{K}_h\times\mathcal{K}_h], \Pr[\mathsf{P}_{K_h^\star,1}(M,M'), (K_h,K_h^\star) \in \mathcal{K}_h\times\mathcal{K}_h]\right)\\ &\overset{(2)}{=} \max\left(\Pr[\mathsf{P}_{K_h,0}(M,M')], \Pr[\mathsf{P}_{K_h^\star,1}(M,M')]\right) \overset{(3)}{\le} \frac{2\ell}{2^n},\end{aligned}$$

where (2) is equivalent to (1) and (3) follows from Eqn. (16). Therefore, we have

$$\epsilon_{\mathrm{univ}}(2,\ell) = \frac{2\ell}{2^n}. \tag{17}$$

Bounding Cover-free advantage of $\mathsf{PH\text{-}DbH}$. To bound the cover-free advantage for any three distinct messages, we first fix three distinct messages M_i, M_j and M_k and for

brevity we denote $\Pr[\mathsf{CF}_{ijk} \text{ holds}, (K_h, K_h^\star) \in \mathcal{K}_h \times \mathcal{K}_h]$ by P_{cf}. Therefore, we have

$$\begin{aligned}
\mathsf{P}_{\mathrm{cf}} &\overset{(1)}{=} \Pr[\mathsf{P}_{K_h,0}(M_i, M_j), \mathsf{P}_{K_h^\star,1}(M_i, M_k), (K_h, K_h^\star) \in \mathcal{K}_h \times \mathcal{K}_h] \\
&\overset{(2)}{=} \Pr[\mathsf{P}_{K_h,0}(M_i, M_j), \mathsf{P}_{K_h^\star,1}(M_i, M_k)] \\
&\overset{(3)}{=} \Pr[\mathsf{P}_{K_h,0}(M_i, M_j)] \cdot \Pr[\mathsf{P}_{K_h^\star,1}(M_i, M_k)] \\
&\overset{(4)}{\leq} \left(\frac{2\ell}{2^n}\right)^2 = \frac{4\ell^2}{2^{2n}},
\end{aligned}$$

where (1) follows from the definition of PH-DbH, (2) is an equivalent form of (1), (3) follows from the independence of K_h and $K_h^\star$ and finally (4) follows from Eqn. (16). Therefore, we have

$$\epsilon_{\mathrm{cf}}(3, \ell) = \frac{4\ell^2}{2^{2n}}. \tag{18}$$

Therefore, the first part of Theorem 3 follows from Eqn. (17) and Eqn. (18).

Bounding Weak-Cover-free and Weak-Block-Wise Universal advantage of PH3-DbH. We know that PolyHash is an $\ell/2^n$-AXU hash function. Therefore, by doing a similar analysis of the weak-cover-free and weak-block-wise universal advantage of PH3-DbH as similarly done for PH-DbH, we have

$$\epsilon_{\mathrm{wuniv}}(2, \ell) = \frac{\ell}{2^n}, \quad \epsilon_{\mathrm{wcf}}(3, \ell) = \frac{\ell^2}{2^{2n}}. \tag{19}$$

Therefore, the second part of Theorem 3 follows from Eqn. (19).

Remark 5. We would like to point out that the security proof for the MAC part of GCM-SIV2 (Lemma 2 of [IM16]) follows a similar analysis as used in the proof of Theorem 3. The MAC part of GCM-SIV2 uses two independent keyed hash functions to generate the two hash e values and independent random permutations in the sum function. Therefore, it provides the desired security even with much weaker assumption on the underlying hash function. To be more precise, the almost universal property of the hash function is sufficient and there is no need to have the cover-free or the blockwise universal restriction of hash function.

5 Parallel Block Cipher Evaluation

In this section, we instantiate the DbH function using block ciphers that operate in a parallel mode, results in a parallel block cipher based DbHtS construction. We analyze the underlying hash function of the PMAC_Plus and the LightMAC_Plus construction, which we refer to as PMAC_Plus-Hash and LightMAC_Plus-Hash respectively. We make a little twist in their design to construct the two-keyed variants of PMAC_Plus and LightMAC_Plus, which we refer to as 2K-PMAC_Plus and 2K-LightMAC_Plus respectively and prove their PRF security using our generalized security result for the two-keyed DbHtS construction.

The double block hash function for 2K-PMAC_Plus and 2K-LightMAC_Plus, which we refer to as 2K-PMAC_Plus-Hash and 2K-LightMAC_Plus-Hash respectively, are structurally almost similar to the PMAC_Plus-Hash and the LightMAC_Plus-Hash, except the following:

(i) We use the fix0 and fix1 functions (to incorporate the block-separated feature).

(ii) We multiply Λ' by 2 before applying the fix1 function on it [5] (see Fig. 5.1).

The algorithms of the DbH function for the PMAC_Plus and the LightMAC_Plus and their respective two-keyed variants are depicted in Figure 5.1.

[5] If we do not multiply by 2, then there exists a trivial birthday bound attack.

PMAC_Plus-Hash(K, M)

$M' \leftarrow M\|10^*$; $M'_1\| \ldots \|M'_l \twoheadleftarrow M'$;
$\Delta_0 \leftarrow E_K(\mathbf{0})$; $\Delta_1 \leftarrow E_K(\mathbf{1})$;
for $j = 1$ **to** l;
  $X_j = M'_j \oplus 2^j\Delta_0 \oplus 2^{2j}\Delta_1$;
  $Y_j = E_K(X_j)$;
$\Sigma' = Y_1 \oplus Y_2 \oplus \ldots \oplus Y_l$;
$\Lambda' = 2^{l-1} \cdot Y_1 \oplus 2^{l-2} \cdot Y_2 \oplus \ldots \oplus Y_l$;
return (Σ', Λ');

LightMAC_Plus-Hash(K, M)

$M' \leftarrow M\|10^*$;
$M'_1\| \ldots \|M'_l \twoheadleftarrow M'$;
for $j = 1$ **to** l;
  $X_j = \langle j \rangle_s \|M'_j$;
  $Y_j = E_K(X_j)$;
$\Sigma' = Y_1 \oplus Y_2 \oplus \ldots \oplus Y_l$;
$\Lambda' = 2^{l-1} \cdot Y_1 \oplus 2^{l-2} \cdot Y_2 \oplus \ldots \oplus Y_l$;
return (Σ', Λ');

2K-PMAC_Plus-Hash(K, M)

$(\Sigma', \Lambda') \leftarrow$ PMAC_Plus-Hash(K, M);
$\Sigma =$ fix0(Σ'); $\Lambda =$ fix1$(2\Lambda')$;
return (Σ, Λ);

2K-LightMAC_Plus-Hash(K, M)

$(\Sigma, \Lambda) \leftarrow$ LightMAC_Plus-Hash(K, M);
$\Sigma =$ fix0(Σ'); $\Lambda =$ fix1$(2\Lambda')$;
return (Σ, Λ);

Figure 5.1: Left: PMAC_Plus-Hash and 2K-PMAC_Plus-Hash; Right: LightMAC_Plus-Hash and 2K-LightMAC_Plus-Hash with s-bit counter. $M'_1\| \ldots \|M'_l \twoheadleftarrow M'$ denotes parsing of message M' into (n bit blocks for PMAC_Plus-Hash; $n-s$ bit blocks for LightMAC_Plus-Hash). $\langle j \rangle_s$ denotes the s-bit binary representation of integer j.

5.1 Bounding Cover-free and Universal Advantages

In this section, we bound the cover-free and universal advantages for 2K-PMAC_Plus-Hash, 2K-LightMAC_Plus-Hash, PMAC_Plus-Hash and LightMAC_Plus-Hash. To do so, we first need to identify the set of bad hash keys for 2K-PMAC_Plus-Hash, PMAC_Plus-Hash, 2K-LightMAC_Plus-Hash and LightMAC_Plus-Hash. Note that, for all these hash functions, the underlying set of bad hash keys is nothing but the set of permutations Π (we consider only the information theoretic setting as switching to the computational setting from the information theoretic one is done by a standard hybrid argument). Now, to identify the set of bad hash keys, we develop a few notations, which will also be required for the analysis of the cover-free and the block-wise universal advantage of these double block hash functions when the hash key is sampled from outside of the set of bad hash keys.

Notations. For a q tuple of distinct messages $(M_1, \ldots, M_q)$, w.l.o.g we assume that the message size(# of bits) for all q messages is a multiple n for PMAC_Plus and multiple of $(n - \lceil \log_2 \ell \rceil)$ for LightMAC_Plus, where ℓ is the maximum message length (# of blocks). We consider two distinct indices $i, j \in [q]$ and define the set $\mathsf{NEQ}_{i,j} := \{\alpha \in [\min\{l_i, l_j\}] : M^i_\alpha \neq M^j_\alpha\} \cup \{\alpha : \min\{l_i, l_j\} + 1 \leq \alpha \leq \max\{l_i, l_j\}\}$. In other words, the set $\mathsf{NEQ}_{i,j}$ contains all the positions, where the message blocks of i-th and j-th message are not equal. $\min \mathsf{NEQ}_{i,j}$ and $\min_2 \mathsf{NEQ}_{i,j}$ denote the minimum and second minimum element of the set $\mathsf{NEQ}_{i,j}$.

Bad Hash Keys for 2K-PMAC_Plus-Hash and PMAC_Plus-Hash. Recall that a hash key for 2K-PMAC_Plus-Hash or PMAC_Plus-Hash is a random permutation. We say that a hash key for 2K-PMAC_Plus-Hash is **bad**, if any of the following events holds:

(a) ZeroOneX: $\exists i \in [q], \alpha \in [l_i]$ such that $X^i_\alpha = \mathbf{0}$ or $X^i_\alpha = \mathbf{1}$.

(b) ZeroY: $\exists i \in [q], \alpha \in [l_i]$ such that $Y^i_\alpha = \mathbf{0}$.

(c) 3CollX: $\exists i \neq j \in [q], i_1, i_2, i_3 \in \{i, j\}, \alpha \in [l_{i_1}], \beta \in [l_{i_2}], \gamma \in \min \mathsf{NEQ}_{i,j}$ where $\alpha \neq \beta \neq \gamma$, such that $X^{i_1}_\alpha = X^{i_2}_\beta = X^{i_3}_\gamma$.

The set of bad hash keys for 2K-PMAC_Plus-Hash and PMAC_Plus-Hash is same and we denote it as $\mathcal{K}_{\text{bad}}^{\text{pp}} \subseteq \mathsf{Perm}$. Therefore, to bound the probability that a hash key, sampled uniformly at random from the hash key space, falls in the set $\mathcal{K}_{\text{bad}}^{\text{pp}}$, is same as bounding the probability of the event $\mathsf{Bad\text{-}Hash} := \mathsf{ZeroOneX} \vee \mathsf{ZeroY} \vee \mathsf{3CollX}$. Therefore, we bound the probability of Bad-Hash as follows:

$$\begin{aligned}\epsilon_{\text{bh}} := \Pr[\mathsf{Bad\text{-}Hash}] &\leq \Pr[\mathsf{ZeroOneX}] + \Pr[\mathsf{ZeroY}] + \Pr[\mathsf{3CollX}] \\ &\leq \frac{2q\ell}{2^n - 1} + \frac{q\ell}{2^n - q\ell} + \frac{q\ell(q\ell - 1)}{2^n(2^n - 1)} \\ &\leq \frac{q^2\ell^2}{2^{2n}} + \frac{5q\ell}{2^n}. \end{aligned} \tag{20}$$

Bad Hash Keys for 2K-LightMAC_Plus-Hash and LightMAC_Plus-Hash. For 2K-LightMAC_Plus-Hash and LightMAC_Plus-Hash, we consider an empty set of bad hash key and as a result we have $\epsilon_{\text{bh}} = 0$.

In the following, we bound the cover-free and the block-wise universal advantage of 2K-PMAC_Plus-Hash and 2K-LightMAC_Plus-Hash.

Theorem 6. 2K-PMAC_Plus-Hash *is a* $(\mathcal{K}_{\text{bad}}^{\text{pp}}, (18\ell + 22)/2^{2n})$*-cover-free,* $(\mathcal{K}_{\text{bad}}^{\text{pp}}, (2\ell + 5)/2^n)$*-block-wise universal* DbH *function and* 2K-LightMAC_Plus-Hash *is a* $(\Phi, 16/2^{2n})$*-cover-free,* $(\Phi, 4/2^n)$*-block-wise universal* DbH *function. In both the cases, we have assumed that* $\ell < 2^{n-1}/3$.

Similarly, we bound the cover-free and the block-wise universal advantage of PMAC_Plus-Hash and LightMAC_Plus-Hash as follows:

Theorem 7. PMAC_Plus-Hash *is a* $(\mathcal{K}_{\text{bad}}^{\text{pp}}, (6(\ell + 1))/2^{2n})$*-weak-cover-free,* $(\mathcal{K}_{\text{bad}}^{\text{pp}}, (2\ell + 3)/2^n)$*-weak-block-wise universal* DbH *function and* LightMAC_Plus-Hash *is a* $(\Phi, 4/2^{2n})$*-weak-cover-free and* $(\Phi, 2/2^n)$*-weak-block-wise universal* DbH *function. Here also we have assumed that* $\ell < 2^{n-1}/3$.

Proofs of Theorem 6 and Theorem 7 are deferred to Sect. 5.4. Assuming that these theorems hold, we now prove the PRF security of 2K-PMAC_Plus and 2K-LightMAC_Plus in Sect. 5.2 and that of PMAC_Plus and LightMAC_Plus in Sect. 5.3.

5.2 PRF Security of 2K-PMAC_Plus and 2K-LightMAC_Plus

2K-PMAC_Plus and 2K-LightMAC_Plus are two parallel mode of block cipher based instantiations of the two-keyed DbHtS. Algorithmic description of these two constructions are depicted in Fig. 5.2.

2K-PMAC_Plus(K_1, K_2, M)	2K-LightMAC_Plus(K_1, K_2, M)
1 : $(\Sigma, \Lambda) \leftarrow$ 2K-PMAC_Plus-Hash(K_1, M);	1 : $(\Sigma, \Lambda) \leftarrow$ 2K-LightMAC_Plus-Hash(K_1, M);
2 : $T \leftarrow E_{K_2}(\Sigma) \oplus E_{K_2}(\Lambda)$;	2 : $T \leftarrow E_{K_2}(\Sigma) \oplus E_{K_2}(\Lambda)$;
return T;	**return** T;

Figure 5.2: Algorithm for 2K-PMAC_Plus 2K-LightMAC_Plus.

The following two results show the PRF security bound of 2K-PMAC_Plus and 2K-LightMACPlus.

Theorem 8 (PRF-Security of 2K-PMAC_Plus). *Let* $\mathcal{K}$ *and* $\mathcal{M}$ *be two non-empty finite sets. Let* $E : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ *be a block cipher. Then, any distinguisher with*

running time at most t, making q tuple of distinct messages each of at most ℓ blocks long, can distinguish 2K-PMAC_Plus$[E]$ *from a n-bit uniform random function by,*

$$\mathbf{Adv}^{\mathrm{prf}}_{\text{2K-PMAC\_Plus}[E]}(q,\ell,t) \leq 2\mathbf{Adv}^{\mathrm{prp}}_{E}(\ell q+2,t') + \frac{9q^3\ell}{2^{2n}} + \frac{q^2\ell^2}{2^{2n}} + \frac{5q\ell}{2^n} + \frac{25q^3}{2^{2n}} + \frac{q}{2^n},$$

where t' is about t plus a time complexity necessary to compute E for $\ell q+2q+2$ times and $\ell < 2^{n-1}/3$.

Proof. Proof of the theorem follows from Eqn. (20), Theorem 6 and part (ii) of Theorem 2. □

Theorem 9 (PRF-Security of 2K-LightMAC_Plus). *Let $\mathcal{K}$ and $\mathcal{M}$ be two non-empty finite sets. Let $E : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher. Then, any distinguisher with running time at most t, making q tuple of distinct messages each of at most ℓ blocks long, can distinguish* 2K-LightMAC_Plus$[E]$ *from a n-bit uniform random function by,*

$$\mathbf{Adv}^{\mathrm{prf}}_{\text{2K-LightMAC\_Plus}[E]}(q,\ell,t) \leq 2\mathbf{Adv}^{\mathrm{prp}}_{E}(\ell q,t') + \frac{21q^3}{2^{2n}} + \frac{q}{2^n},$$

where t' is about t plus a time complexity necessary to compute E for $\ell q+2q$ times and $\ell < 2^{n-1}/3$.

Proof. As there is no set of bad hash keys, $\epsilon_{\mathrm{bh}} = 0$ and the rest of the proof follows from Theorem 6 and part (ii) of Theorem 2. □

5.3 PRF Security of PMAC_Plus and LightMAC_Plus

PMAC_Plus and LightMAC_Plus are two instantiations of the three-keyed DbHtS. Although, these constructions are existing ones, as proposed by Yasuda [Yas11] and Naito [Nai17] respectively, for the sake of completeness of this paper, we state and prove the security of these two constructions in our setting. We recall these two constructions in Fig. 5.3.

PMAC_Plus(K_1, K_2, K_3, M)

$(\Sigma', \Lambda') \leftarrow$ PMAC_Plus-Hash(K_1, M);
$T \leftarrow E_{K_2}(\Sigma') \oplus E_{K_3}(\Lambda')$;
return T;

LightMAC_Plus(K_1, K_2, K_3, M)

$(\Sigma', \Lambda') \leftarrow$ LightMAC_Plus-Hash(K_1, M);
$T \leftarrow E_{K_2}(\Sigma') \oplus E_{K_3}(\Lambda')$;
return T;

Figure 5.3: Algorithm for PMAC_Plus and LightMAC_Plus.

The following two results show the PRF security bound of PMAC_Plus and LightMAC_Plus.

Theorem 10 (PRF-Security of PMAC_Plus). *Let $\mathcal{K}$ and $\mathcal{M}$ be two non-empty finite sets. Let $E : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher. Then, any distinguisher with running time at most t, making q tuple of distinct messages each of at most ℓ blocks long, can distinguish* PMAC_Plus$[E]$ *from a n-bit uniform random function by,*

$$\mathbf{Adv}^{\mathrm{prf}}_{\text{PMAC\_Plus}[E]}(q,\ell,t) \leq 3\mathbf{Adv}^{\mathrm{prp}}_{E}(\ell q+2,t') + \frac{7q^3\ell}{2^{2n}} + \frac{q^2\ell^2}{2^{2n}} + \frac{5q\ell}{2^n} + \frac{12q^3}{2^{2n}},$$

where t' is about t plus a time complexity necessary to compute E for $\ell q+2q+2$ times and $\ell < 2^{n-1}/3$.

Proof. Proof of the theorem follows from Eqn. (20), Theorem 7 and part (iii) of Theorem 2. □

Theorem 11 (PRF-Security of LightMAC_Plus). *Let $\mathcal{K}$ and $\mathcal{M}$ be two non-empty finite sets. Let $E : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher. Then, any distinguisher with running time at most t, making q tuple of distinct messages each of at most ℓ blocks long, can distinguish* LightMAC_Plus$[E]$ *from a n-bit uniform random function by,*

$$\mathbf{Adv}^{\mathrm{prf}}_{\mathsf{LightMAC\_Plus}[E]}(q,\ell,t) \leq 3\mathbf{Adv}^{\mathrm{prp}}_{E}(\ell q, t') + \frac{9q^3}{2^{2n}},$$

where t' is about t plus a time complexity necessary to compute E for $\ell q + 2q$ times and $\ell < 2^{n-1}/3$.

Proof. Proof of the theorem follows from Theorem 7, part (iii) of Theorem 2 and the fact that $\epsilon_{\mathrm{bh}} = 0$ (as there is no set of bad hash keys). □

Note. The original security bound of the PMAC_Plus, as proven by Yasuda [Yas11], is roughly $q^3\ell^3/2^{2n}$ (we only mention the dominating term of the security bound). But, according to Theorem 10, the dominating term of the security bound of the PMAC_Plus is $q^3\ell/2^{2n}$, a substantial improvement of the security bound over its existing one. A similar improvement in the security bound is also done in 1k-PMAC_Plus [DDN+17] over PMAC_Plus. However, we are not gaining any security improvement for LightMAC_Plus.

5.4 Proof of Theorem 6 and Theorem 7

In this section, we mainly prove Theorem 6. In particular, we bound the cover-free advantage and the block-wise universal advantage of 2K-PMAC_Plus-Hash and 2K-LightMAC_Plus-Hash. Using parts of these analysis, we prove Theorem 7.

5.4.1 Cover-free and Block-wise universal Advantage of 2K-PMAC_Plus-Hash

We bound the cover-free and the block-wise universal advantage of 2K-PMAC_Plus-Hash.

Bounding Cover-free-advantage. We fix three distinct messages M_i, M_j and M_k and define the event $\mathsf{CollX}_{ijk} := X^{i_1}_{\alpha} = X^{i_2}_{\beta}$ [6], where $i_1, i_2 \in \{i,j,k\}$ are distinct and $\alpha \in \{l_{i_1}, \min \mathsf{NEQ}_{i_1,i_2}, \min_2 \mathsf{NEQ}_{i_1,i_2}\}$, $\beta \in [l_{i_2}]$ are distinct.

For brevity, let us introduce the following notations:

- $\mathsf{E}_{b,b'} := (\Sigma_i \oplus \Sigma_j = b, \Lambda_i \oplus \Lambda_k = b')$.
- $\mathcal{K}_g := \mathsf{Perm} \setminus \mathcal{K}^{\mathsf{pp}}_{\mathrm{bad}}$.
- $\mathsf{Good} := \overline{\mathsf{3CollX}} \wedge \overline{\mathsf{ZeroOneX}} \wedge \overline{\mathsf{ZeroY}}$.

Now we denote the probability of the joint event that CF_{ijk} holds and $\Pi \in \mathcal{K}_g$ by P_{cf}. Acording to the definition of cover-free advantage (see Definition 1), we have

$$\begin{aligned}
\mathsf{P}_{\mathrm{cf}} &:= \sum_{b,b' \in \{\mathbf{0},\mathbf{1}\}} \Pr[\mathsf{E}_{b,b'}, \Pi \in \mathcal{K}_g] \\
&= \sum_{b,b' \in \{\mathbf{0},\mathbf{1}\}} \sum_{(\delta_0,\delta_1):\mathsf{Good}} \underbrace{\Pr[\mathsf{E}_{b,b'}, (\Delta_0,\Delta_1) = (\delta_0,\delta_1), \Pi \in \mathcal{K}_g]}_{\epsilon_{b,b'}} \\
&= \underbrace{\sum_{(\delta_0,\delta_1):\mathsf{Good}} \epsilon_{\mathbf{1},\mathbf{0}}}_{\mu} + \underbrace{\sum_{\substack{b,b' \in \{\mathbf{0},\mathbf{1}\}, \\ (b,b') \neq (\mathbf{1},\mathbf{0})}} \sum_{(\delta_0,\delta_1):\mathsf{Good}} \epsilon_{b,b'}}_{\nu} \qquad (21)
\end{aligned}$$

[6] Although the event CollX_{ijk} involves only two indices, we define it over three indices as the set of bad hash keys are themselves defined over three indices.

BOUNDING μ: We bound μ as follows:

$$\mu = \sum_{(\delta_0,\delta_1):\mathsf{Good}} \underbrace{\Pr[\mathsf{E}_{\mathbf{1},\mathbf{0}}, \Pi \in \mathcal{K}_g \mid (\Delta_0,\Delta_1) = (\delta_0,\delta_1)]}_{\psi} \cdot \Pr[(\Delta_0,\Delta_1) = (\delta_0,\delta_1)]$$

By conditioning (Δ_0, Δ_1) to a fixed (δ_0, δ_1), we fix all the X^i_α values. This gives a collision relation $\sim$ among the X^i_α values: $(i,\alpha) \sim (j,\beta)$ iff $X^i_\alpha = X^j_\beta$. Since, (δ_0,δ_1) is a good pair [7], $X^i_\alpha \notin \{\mathbf{0},\mathbf{1}\}$, and hence Y^i_α (the corresponding permutation output of X^i_α) values are wor sampled from $\{0,1\}^n \setminus \{\delta_0,\delta_1\}$. Now the event $(\Sigma_i \oplus \Sigma_j = \mathbf{1}, \Lambda_i \oplus \Lambda_k = \mathbf{0})$ gives a system of two linear equations in Y variables

$$\mathcal{E} = \begin{cases} \mathcal{L}_1 := Y^i_1 \oplus \ldots Y^i_{l_i} \oplus Y^j_1 \oplus \ldots \oplus Y^j_{l_j} = \mathbf{1}, \\ \mathcal{L}_2 := 2^{l_i} Y^i_1 \oplus \ldots 2Y^i_{l_i} \oplus 2^{l_j} Y^j_1 \oplus \ldots \oplus 2Y^j_{l_j} = \mathbf{0}. \end{cases}$$

By applying the collision relation $\sim$ over $\mathcal{L}_1$ and $\mathcal{L}_2$, we obtain a reduced system of equations, denoted as $\mathcal{E}^\sim$. It is easy to see that the rank of $\mathcal{E}^\sim$ is 2 and hence, by applying Lemma 1, we have $\psi \le \frac{1}{(2^n - 3\ell + 1)_2}$. Therefore,

$$\mu \le \sum_{(\delta_0,\delta_1):\mathsf{Good}} \frac{1}{(2^n - 3\ell + 1)_2} \cdot \Pr[(\Delta_0,\Delta_1) = (\delta_0,\delta_1)] \le \frac{4}{2^{2n}}, \quad \text{where } \ell \le (2^{n-1}+1)/3. \tag{22}$$

BOUNDING ν: Here we have

$$\begin{aligned} \nu &= \sum_{\substack{(\delta_0,\delta_1):\mathsf{Good} \\ \wedge \mathsf{CollX}_{ijk}}} \epsilon_{b,b'} + \sum_{\substack{(\delta_0,\delta_1):\mathsf{Good} \\ \wedge \overline{\mathsf{CollX}_{ijk}}}} \epsilon_{b,b'} \\ &= \sum_{\substack{(\delta_0,\delta_1):\mathsf{Good} \\ \wedge \mathsf{CollX}_{ijk}}} \Pr[\mathsf{E}_{b,b'}, \Pi \in \mathcal{K}_g \mid (\Delta_0,\Delta_1) = (\delta_0,\delta_1)] \cdot \Pr[(\Delta_0,\Delta_1) = (\delta_0,\delta_1)] \\ &+ \sum_{\substack{(\delta_0,\delta_1):\mathsf{Good} \\ \wedge \overline{\mathsf{CollX}_{ijk}}}} \Pr[\mathsf{E}_{b,b'}, \Pi \in \mathcal{K}_g \mid (\Delta_0,\Delta_1) = (\delta_0,\delta_1)] \cdot \Pr[(\Delta_0,\Delta_1) = (\delta_0,\delta_1)] \end{aligned} \tag{23}$$

Now, we follow the similar approach as in the previous case. Here, we argue that (i) if CollX_{ijk} occurs, then the rank of the reduced system of equations $\mathcal{E}^\sim$ is at least 1 and (ii) else (i.e. CollX_{ijk} does not occur) the rank of the reduced system of equations $\mathcal{E}^\sim$ is 2 (see Claim 5, [DDN+17]). Note that, for $b = \mathbf{0}, b' = \mathbf{0}$, we need the event $\overline{\mathsf{ZeroY}}$ as otherwise the rank of the reduced system of equations $\mathcal{E}^\sim$ (when $\overline{\mathsf{CollX}_{ijk}}$ occurs) would have been 1 (say there are two messages M_1 and $M_1 \| M_2$, then $Y_2 = 0$ makes the first equation trivial). Hence, we have

$$\Pr[\mathsf{E}_{b,b'}, \Pi \in \mathcal{K}_g \mid (\Delta_0,\Delta_1) = (\delta_0,\delta_1)] \quad = \quad \begin{cases} \frac{1}{(2^n - 3\ell - 1)_1} & \text{if } \mathsf{CollX}_{ijk} \text{ occurs} \\ \frac{1}{(2^n - 3\ell - 1)_2} & \text{else} \end{cases}$$

Both these bounds follow from Lemma 1. Therefore, from Eqn.(23)

$$\nu \le \sum_{(\delta_0,\delta_1):\mathsf{Good} \wedge \mathsf{CollX}_{ijk}} \frac{\Pr[(\Delta_0,\Delta_1) = (\delta_0,\delta_1)]}{(2^n - 3\ell - 1)_1} + \sum_{(\delta_0,\delta_1):\mathsf{Good} \wedge \overline{\mathsf{CollX}_{ijk}}} \frac{\Pr[(\Delta_0,\Delta_1) = (\delta_0,\delta_1)]}{(2^n - 3\ell - 1)_2}$$

Putting the inequalities (i) $\Pr[(\Delta_0,\Delta_1) = (\delta_0,\delta_1)] = \frac{1}{2^n(2^n-1)}$, (ii) $|(\delta_0,\delta_1) : \mathsf{Good} \wedge \mathsf{CollX}_{ijk}| \le 2^n \cdot 3\ell$ and (iii) $|(\delta_0,\delta_1) : \mathsf{Good} \wedge \overline{\mathsf{CollX}_{ijk}}| \le 2^n(2^n-1)$, we have

$$\nu \le \frac{2^n \cdot 3\ell}{2^n(2^n-1)} \cdot \frac{1}{(2^n - 3\ell - 1)} + \frac{2^n(2^n-1)}{2^n(2^n-1)} \cdot \frac{1}{(2^n - 3\ell - 1)_2} \le \frac{6(\ell+1)}{2^{2n}}, \tag{24}$$

[7] (δ_0,δ_1) is said to be a good pair if none of the three events 3CollX, ZeroY or ZeroOneX occur.

where $\ell \leq (2^{n-1}-1)/3$.

Combining Eqn.(21), Eqn.(22) and Eqn.(24), we obtain $\mathsf{P}_{\mathrm{cf}} \leq \frac{18\ell+22}{2^{2n}}$, where $\ell \leq (2^{n-1}-1)/3$, and hence we can set

$$\epsilon_{\mathrm{cf}}(3,\ell) = \frac{18\ell+22}{2^{2n}}, \text{ assuming } \ell \leq (2^{n-1}-1)/3. \tag{25}$$

Bounding Block-wise-universal advantage. We fix two distinct messages M_i and M_j and define the event $\mathsf{CollX}_{ij} := X_\alpha^{i_1} = X_\beta^{i_2}$, where $i_1, i_2 \in \{i,j\}$ are distinct and $\alpha \in \{l_{i_1}, \min \mathsf{NEQ}_{i_1,i_2}, \min_2 \mathsf{NEQ}_{i_1,i_2}\}$, $\beta \in [l_{i_2}]$ are distinct. With a similar argument as used in the case of bounding the cover-free advantage, one can see that the number of (Δ_0, Δ_1) for which CollX_{ij} holds is at most to $2^n \cdot 2\ell$.

For brevity, let us introduce the notation:

- $\mathsf{E}_b^1 := \Sigma_i \oplus \Sigma_j = b$.
- $\mathsf{E}_b^2 := \Lambda_i \oplus \Lambda_k = b$.

Now we denote the probability of the joint event that UNIV_{ij} holds and $\Pi \in \mathcal{K}_g$ by $\mathsf{P}_{\mathrm{univ}}$. According to the definition of block-wise universal advantage (see Definition 3), we have

$$\mathsf{P}_{\mathrm{univ}} := \sum_{b \in \{\mathbf{0},\mathbf{1}\}} \max\left(\Pr[\mathsf{E}_b^1, \Pi \in \mathcal{K}_g],\ \Pr[\mathsf{E}_b^2, \Pi \in \mathcal{K}_g]\right)$$

Using similar approach as used in case of the cover-free case, here we have,

$$\begin{aligned}
\sum_{b \in \{\mathbf{0},\mathbf{1}\}} \Pr[\mathsf{E}_b^1, \Pi \in \mathcal{K}_g] &= \Pr[\mathsf{E}_{\mathbf{1}}^1, \Pi \in \mathcal{K}_g] + \Pr[\mathsf{E}_{\mathbf{0}}^1, \Pi \in \mathcal{K}_g] \\
&\leq \sum_{(\delta_0,\delta_1):\mathsf{Good}} \frac{\Pr[(\Delta_0,\Delta_1) = (\delta_0,\delta_1)]}{(2^n - 2\ell - 1)_1} \\
&+ \sum_{(\delta_0,\delta_1):\mathsf{Good} \wedge \mathsf{CollX}_{ij}} \Pr[(\Delta_0,\Delta_1) = (\delta_0,\delta_1)] \\
&+ \sum_{(\delta_0,\delta_1):\mathsf{Good} \wedge \overline{\mathsf{CollX}_{ij}}} \frac{\Pr[(\Delta_0,\Delta_1) = (\delta_0,\delta_1)]}{(2^n - 2\ell - 1)_1}
\end{aligned}$$

Putting the inequalities (i) $\Pr[(\Delta_0,\Delta_1) = (\delta_0,\delta_1)] = \frac{1}{2^n(2^n-1)}$, (ii) $|(\delta_0,\delta_1) : \mathsf{Good}| \leq 2^n(2^n-1)$, (iii) $|(\delta_0,\delta_1) : \mathsf{Good} \wedge \mathsf{CollX}_{ij}| \leq 2^n \cdot 2\ell$ and (iv) $|(\delta_0,\delta_1) : \mathsf{Good} \wedge \overline{\mathsf{CollX}_{ij}}| \leq 2^n(2^n-1)$, we have

$$\begin{aligned}
\sum_{b \in \{\mathbf{0},\mathbf{1}\}} \Pr[\mathsf{E}_b^1, \Pi \in \mathcal{K}_g] &\leq \frac{1}{(2^n - 2\ell - 1)} + \frac{2^n \cdot 2\ell}{2^n(2^n-1)(2^n-2\ell-1)} + \frac{2^n(2^n-1)}{2^n(2^n-1)(2^n-2\ell-1)} \\
&\leq \frac{2\ell+5}{2^n},
\end{aligned}$$

assuming $\ell \leq 2^{n-2}$. Here use the fact that (i) if $b = 1$ or (ii) $b = 0$ and CollX_{ij} doesn't occurs, then the rank of the reduced system of equations E_b^1 given a fixed value of (Δ_0, Δ_1) is at least 1. With a similar argument, one can show that

$$\sum_{b \in \{\mathbf{0},\mathbf{1}\}} \Pr[\mathsf{E}_b^2, \Pi \in \mathcal{K}_g] \leq \frac{2\ell+5}{2^n},$$

and hence we can set

$$\epsilon_{\mathrm{univ}}(2,\ell) = \frac{2\ell+5}{2^n}, \quad \text{assuming} \quad \ell \leq 2^{n-2}. \tag{26}$$

The first part of Theorem 6 follows from Eqn. (25) and Eqn. (26).

5.4.2 Cover-free and Block-wise universal Advantage of 2K-LightMAC_Plus-Hash

In this section, we bound the cover-free and the block-wise universal advantage of 2K-LightMAC_Plus-Hash. Recall that, for 2K-LightMAC_Plus-Hash, we have considered an empty set of bad hash keys.

Bounding Cover-free-advantage. Since there is an empty set of bad hash keys, we sample the hash key, i.e., the random permutation Π, from the set of all permutations for bounding the cover-free advantage. First, we fix three distinct messages M_i, M_j and M_k. Similar to the previous analysis, for two fixed $b, b' \in \{\mathbf{0}, \mathbf{1}\}$, we write the two equations $\Sigma_i \oplus \Sigma_j = b$ and $\Lambda_i \oplus \Lambda_k = b'$ in terms of the Y-variables as follows:

$$\mathcal{E} = \begin{cases} (Y_1^i \oplus \ldots \oplus Y_{l_i}^i) \oplus (Y_1^j \oplus \ldots \oplus Y_{l_j}^j) = b \\ (2^{l_i} Y_1^i \oplus \ldots \oplus 2Y_{l_i}^i) \oplus (2^{l_k} Y_1^k \oplus \ldots \oplus 2Y_{l_k}^k) = b'. \end{cases}$$

Given the two equations are consistent, one can always find two random variables $Y_\alpha^{i_1}$ and $Y_\beta^{i_2}$, where $i_1, i_2 \in \{i, j, k\}$ and distinct $\alpha \in \mathsf{NEQ}_{ij}, \beta \in \mathsf{NEQ}_{ik}$ such that the rank of $\mathcal{E}$ is 2. Again we use the notation P_{cf} to denote $\Pr[\mathsf{CF}_{ijk}$ holds, $\Pi \in \mathsf{Perm}]$. Therefore, we have

$$\begin{aligned} \mathsf{P}_{\mathrm{cf}} &= \sum_{b,b' \in \{\mathbf{0},\mathbf{1}\}} \Pr[\Sigma_i \oplus \Sigma_j = b, \Lambda_i \oplus \Lambda_k = b', \Pi \in \mathsf{Perm}] \\ &\overset{(1)}{\leq} \sum_{b,b' \in \{\mathbf{0},\mathbf{1}\}} \frac{1}{(2^n - 3\ell + 2)_2} \leq \frac{16}{2^{2n}}, \end{aligned} \tag{27}$$

where (1) follows by applying Lemma 1 and we assume that $\ell \leq 2^{n-1}/3$. Hence, we can set

$$\epsilon_{\mathrm{cf}}(3, \ell) = \frac{16}{2^{2n}}. \tag{28}$$

Bounding Block-wise-universal-advantage. To bound the block-wise-universal advantage, we first fix two distinct messages M_i and M_j. By definition, we need to bound

$$\mathsf{P}_{\mathrm{univ}} = \max\Big(\sum_{b \in \{\mathbf{0},\mathbf{1}\}} \Pr[\Sigma_i \oplus \Sigma_j = b, \Pi \in \mathsf{Perm}], \sum_{b \in \{\mathbf{0},\mathbf{1}\}} \Pr[\Lambda_i \oplus \Lambda_j = b, \Pi \in \mathsf{Perm}] \Big), \tag{29}$$

where $\mathsf{P}_{\mathrm{univ}}$ is a shorthand notation for $\Pr[\mathsf{UNIV}_{ij}$ holds, $\Pi \in \mathsf{Perm}]$. Now, we bound these terms case by case as follows:

Bounding $\Pr[\Sigma_i \oplus \Sigma_j = \mathbf{1}, \Pi \in \mathsf{Perm}]$: $\Sigma_i \oplus \Sigma_j = \mathbf{1}$ implies the following non-trivial equation:

$$(Y_1^i \oplus \ldots Y_{l_i}^i) \oplus (Y_1^j \oplus \ldots \oplus Y_{l_j}^j) = \mathbf{1}.$$

From Lemma 1, the above equation holds with probability at most $\frac{1}{2^n - 2\ell + 1} \leq \frac{2}{2^n}$, when $\ell \leq 2^{n-2}$.

Bounding $\Pr[\Sigma_i \oplus \Sigma_j = \mathbf{0}, \Pi \in \mathsf{Perm}]$: This is proven in the following sub-cases:

- We first consider the case $l_i = l_j$. Observe that, $|\mathsf{NEQ}_{i,j}| \geq 1$, otherwise the probability would have been zero. Therefore, let us assume $|\mathsf{NEQ}_{i,j}| = s > 1$. Now, $\Sigma_i = \Sigma_j$ implies the following equation:

  $$(Y_{\alpha_1}^i \oplus \ldots \oplus Y_{\alpha_s}^i) \oplus (Y_{\alpha_1}^j \oplus \ldots \oplus Y_{\alpha_s}^j) = \mathbf{0},$$

  where $\alpha_1, \ldots, \alpha_s \in \mathsf{NEQ}_{i,j}$. Since, $|\mathsf{NEQ}_{i,j}| > 1$, we obtain at least one random variable $Y_\alpha^\star$, where $\alpha \in \mathsf{NEQ}_{i,j}$ for which the rank of the above equation is 1 and hence from Lemma 1, the above equations holds with probability at most $\frac{1}{2^n - 2\ell + 1} \leq \frac{2}{2^n}$, when $\ell \leq 2^{n-2}$.

- Now, we consider the case when $l_i \neq l_j$. W.l.o.g we assume that $l_i > l_j$. Then $l_j+1, \ldots, l_i \in \mathsf{NEQ}_{i,j}$. Let $\mathsf{NEQ}'_{i,j} := \mathsf{NEQ}_{i,j} \setminus \{l_j+1, \ldots, l_i\}$ and let us also consider that $|\mathsf{NEQ}'_{i,j}| = s$. Note that, s can also be zero. Therefore, $\Sigma_i = \Sigma_j$ implies

$$(Y^i_{l_j+1} \oplus \ldots \oplus Y^i_{l_i}) \oplus (Y^i_{\alpha_1} \oplus \ldots \oplus Y^i_{\alpha_s}) \oplus (Y^j_{\alpha_1} \oplus \ldots \oplus Y^j_{\alpha_s}) = \mathbf{0},$$

  where $\alpha_1, \ldots, \alpha_s \in \mathsf{NEQ}'_{i,j}$. The above equation is non-trivial and hence its rank is 1. Therefore, from Lemma 1, the above equation holds with probability at most $\frac{1}{2^n - 2\ell + 1} \leq \frac{2}{2^n}$, when $\ell \leq 2^{n-2}$.

Bounding $\Pr[\Lambda_i \oplus \Lambda_j = \mathbf{1}, \Pi \in \mathsf{Perm}]$: $\Lambda_i \oplus \Lambda_j = \mathbf{1}$ gives rise to the following non-trivial equation:

$$(2^{l_i} Y^i_1 \oplus \ldots 2Y^i_{l_i}) \oplus (2^{l_j} Y^j_1 \oplus \ldots \oplus 2Y^j_{l_j}) = \mathbf{1},$$

which holds with probability at most $\frac{1}{2^n - 2\ell + 1} \leq \frac{2}{2^n}$, when $\ell \leq 2^{n-2}$.

Bounding $\Pr[\Lambda_i \oplus \Lambda_j = \mathbf{0}, \Pi \in \mathsf{Perm}]$: Similar to the earlier analysis, we bound the probability of the event in different sub-cases as follows:

- Similar to the previous argument, if $l_i = l_j$ and $|\mathsf{NEQ}_{i,j}| = 1$, then the probability would have been zero. Hence, we assume that $|\mathsf{NEQ}_{i,j}| = s > 1$. Then, $\Lambda_i = \Lambda_j$ implies the following equation:

$$2^{l_i - \alpha_1 - 1}(Y^i_{\alpha_1} \oplus Y^j_{\alpha_1}) \oplus \ldots \oplus 2^{l_i - \alpha_s - 1}(Y^i_{\alpha_s} \oplus Y^j_{\alpha_s}) = \mathbf{0},$$

  where $\alpha_1, \ldots, \alpha_s \in \mathsf{NEQ}_{i,j}$. Since, the above equation is non-trivial, from Lemma 1, the probability that the above equation holds is at most $\frac{1}{2^n - 2\ell + 1} \leq \frac{2}{2^n}$, when $\ell \leq 2^{n-2}$.

- For the case of $l_i \neq l_j$ (w.l.o.g we assume $l_i > l_j$), then $l_j + 1, \ldots, l_i \in \mathsf{NEQ}_{i,j}$. Moreover, $\mathsf{NEQ}'_{i,j} := \mathsf{NEQ}_{i,j} \setminus \{l_j + 1, \ldots, l_i\}$ and let us also consider $|\mathsf{NEQ}'_{i,j}| = s$. Therefore, the event $\Lambda_i = \Lambda_j$ implies

$$\begin{aligned}&\left(2^{l_i - l_j} Y^i_{l_j+1} \oplus \ldots \oplus 2Y^i_{l_i}\right) \oplus \left(2^{l_i - \alpha_1 - 1} Y^i_{\alpha_1} \oplus \ldots \oplus 2^{l_i - \alpha_s - 1} Y^i_{\alpha_s}\right) \\ &\oplus \left(2^{l_j - \alpha_1 - 1} Y^j_{\alpha_1} \oplus \ldots \oplus 2^{l_j - \alpha_s - 1} Y^j_{\alpha_s}\right) = \mathbf{0},\end{aligned}$$

  where $\alpha_1, \ldots, \alpha_s \in \mathsf{NEQ}'_{i,j}$. Since, the rank of the above equation is 1, from Lemma 1, the probability that the above equation holds is at most $\frac{1}{2^n - 2\ell + 1} \leq \frac{2}{2^n}$.

Therefore, we have

$$\sum_{b \in \{\mathbf{0},\mathbf{1}\}} \Pr[\Sigma_i \oplus \Sigma_j = b, \Pi \in \mathsf{Perm}] \leq \frac{4}{2^n}, \quad \sum_{b \in \{\mathbf{0},\mathbf{1}\}} \Pr[\Lambda_i \oplus \Lambda_j = b, \Pi \in \mathsf{Perm}] \leq \frac{4}{2^n}. \quad (30)$$

Therefore, from Eqn. (29) and Eqn. (30) we have $\mathsf{P}_{\text{univ}} \leq \frac{4}{2^n}$, and hence we can set

$$\epsilon_{\text{univ}}(2, \ell) = \frac{4}{2^n}. \quad (31)$$

The second part of Theorem 6 follows from Eqn. (28) and Eqn. (31).

5.4.3 Weak-cover-free and Weak-block-wise-universal Advantage of PMAC_Plus-Hash

For PMAC_Plus-Hash, we have

$$\mathsf{P}_{\mathrm{wcf}} = \Pr[\Sigma_i = \Sigma_j, \Lambda_i = \Lambda_k, \Pi \in \mathsf{Perm} \setminus \mathcal{K}^{\mathtt{pp}}_{\mathrm{bad}}] \leq \frac{6(\ell+1)}{2^{2n}},$$
$$\mathsf{P}_{\mathrm{wuniv}} = \max\left(\Pr[\Sigma_i = \Sigma_j, \Pi \in \mathsf{Perm} \setminus \mathcal{K}^{\mathtt{pp}}_{\mathrm{bad}}], \Pr[\Lambda_i = \Lambda_j, \Pi \in \mathsf{Perm} \setminus \mathcal{K}^{\mathtt{pp}}_{\mathrm{bad}}]\right) \leq \frac{2\ell+3}{2^n}$$

The bound for $\mathsf{P}_{\mathrm{wcf}}$ follows directly from the cover-free analysis of PMAC_Plus-Hash with $b = b' = \mathbf{0}$ and the bound for $\mathsf{P}_{\mathrm{wuniv}}$ follows directly from the universal analysis with $b = \mathbf{0}$.

Hence, we have

$$\epsilon_{\mathrm{wcf}}(3,\ell) = \frac{6(\ell+1)}{2^{2n}}, \quad \epsilon_{\mathrm{wuniv}}(2,\ell) = \frac{2\ell+3}{2^n}. \tag{32}$$

The first part of Theorem 7 follows from Eqn. (32).

5.4.4 Weak-cover-free and Weak-block-wise-universal Advantage of LightMAC_Plus-Hash

For LightMAC_Plus-Hash, we have

$$\mathsf{P}_{\mathrm{wcf}} = \Pr[\Sigma_i = \Sigma_j, \Lambda_i = \Lambda_k, \Pi \in \mathsf{Perm}] \leq \frac{4}{2^{2n}},$$
$$\mathsf{P}_{\mathrm{wuniv}} = \max\left(\Pr[\Sigma_i = \Sigma_j, \Pi \in \mathsf{Perm}], \Pr[\Lambda_i = \Lambda_j, \Pi \in \mathsf{Perm}]\right) \leq \frac{2}{2^n}$$

The bounds are directly derived from the bound for $\mathsf{P}_{\mathrm{wcf}}$ and $\mathsf{P}_{\mathrm{wuniv}}$ used in 2K-LightMAC_Plus-Hash with the restriction that $b = b' = \mathbf{0}$ in the first case and $b = \mathbf{0}$ in the second.

Hence, we have

$$\epsilon_{\mathrm{wcf}}(3,\ell) = \frac{4}{2^{2n}}, \quad \epsilon_{\mathrm{wuniv}}(2,\ell) = \frac{2}{2^n}. \tag{33}$$

The second part of Theorem 7 follows from Eqn. (33).

6 Sequential Block Cipher Evaluation

In this section, we instantiate the DbH function using block ciphers that operate in sequential mode, results in a sequential block cipher based DbHtS construction. We analyze the underlying hash function of the SUM-ECBC and the 3kf9 construction, which we refer to as ECBC-Hash and f9-Hash respectively and we also make a little twist in their design to construct the two-keyed variant of the SUM-ECBC and the 3kf9. As a result, we propose a two-keyed variant of the SUM-ECBC and the 3kf9, which we refer to as 2K-ECBC_Plus and 2Kf9 respectively and prove their PRF security using our generalized security result for the two-keyed DbHtS construction.

We refer to the DbH function for 2K-ECBC_Plus and 2Kf9 as 2K-ECBC-Hash and f9-Hash (for 2Kf9, we use the same DbH function as used for 3kf9) respectively. 2K-ECBC-Hash is structurally very similar to the ECBC-Hash, except the following that 2K-ECBC-Hash uses fix0 and fix1 functions to incorporate the block-separated feature, which are absent in the ECBC-Hash.

The algorithms of the DbH function for 2K-ECBC_Plus and 2Kf9 is depicted in Figure 6.1.

ECBC-Hash$(K, K_\star, M)$

$M' \leftarrow M\|10^*; M'_1\| \ldots \|M'_l \twoheadleftarrow M';$
$(Y, Y') \leftarrow (\mathbf{0}, \mathbf{0});$
`for` $j = 1$ `to` $l;$
  $X = M'_j \oplus Y; X' = M'_j \oplus Y';$
  $(Y, Y') \leftarrow (E_K(X), E_{K_\star}(X'));$
$(\Sigma'\Lambda') \leftarrow (Y, Y');$
return $(\Sigma', \Lambda');$

f9-Hash(K, M)

$M' \leftarrow M\|10^*; M'_1\| \ldots \|M'_l \twoheadleftarrow M';$
$(Y, Y') \leftarrow (\mathbf{0}, \mathbf{0});$
`for` $j = 1$ `to` $l;$
  $X = M'_j \oplus Y; Y \leftarrow E_K(X);$
  $Y' \leftarrow Y \oplus Y';$
$(\Sigma', \Lambda') = (Y, Y');$
return $(\Sigma', \Lambda');$

2K-ECBC-Hash$(K, K_\star, M)$

$(\Sigma', \Lambda') \leftarrow$ ECBC-Hash$(K, K_\star, M);$
$\Sigma \leftarrow$ fix0$(\Sigma'); \Lambda \leftarrow$ fix1$(\Theta');$
return $(\Sigma, \Lambda);$

Figure 6.1: Left: ECBC-Hash and 2K-ECBC-Hash; Right: f9-Hash. $M'_1\| \ldots \|M'_l \twoheadleftarrow M'$ denotes the parsing of the message M' into l many n bit blocks.

6.1 Bounding Cover-free and Universal Advantages

In this section, we bound the cover-free and universal advantages for 2K-ECBC-Hash, 2K-LightMAC_Plus-Hash, PMAC_Plus-Hash and LightMAC_Plus-Hash. To do so, we first need to identify the set of bad hash keys for 2K-ECBC-Hash and f9-Hash. Note that, for both of the DbH functions, the underlying set of bad hash keys is nothing but the set of permutations Π or the set of pair of independent permutations (Π_1, Π_2) (for 2K-ECBC-Hash). We consider only the information theoretic setting as switching to the computational setting from the information theoretic one is done by a standard hybrid argument. To identify the set of bad hash keys, we revisit to an important notion called **structure graph** [BPR05, GPR14, DNP16, JN16] and some of its important results which will help us in bounding the cover-free and the block-wise universal advantage of 2K-ECBC-Hash and f9-Hash, when the hash key is sampled outside from the set of bad hash keys.

Revisiting the Structure Graph. Here we briefly revisit the structure graphs, introduced by Bellare et al. [BPR05] and recall some of their results which would be required in the security analysis of 2K-ECBC-Hash and f9-Hash. Let M be a message and without loss of generality, we assume that the size of M (in number of bits) is a multiple of n. Thus, we partition M as a sequence of l many n-bit blocks as $M = M[1]\|M[2]\| \ldots \|M[l]$. Now, we apply CBC-MAC [BKR00], based on an n-bit uniform random permutation Π, on M and let the intermediate chaining values of CBC-MAC(M) be as follows:

$$Y_0 = 0^n, \text{ and } Y_i = \Pi(Y_{i-1} \oplus M[i]) \text{ for } i = 1, \ldots, l,$$

where $M[i]$ is the i-th block of message M.

Informally, for any two fixed distinct messages M, M' and a uniformly chosen random permutation Π, we construct the structure graph $\mathcal{G}^{\Pi}(M, M')$ with the set of nodes $\{0,1\}^n$ as follows: We follow the CBC-MAC computations for M followed by those of M' by creating nodes which are labeled by the intermediate chaining variables Y_i. In this process, if we arrive at a vertex already labeled, while not following an existing edge, we call this event a collision.[8] The sequence of alternating vertices and edges corresponding to the computations for a message M is called a message walk of M. Like for two distinct

[8] We use the term collision and accident interchangeably.

messages, we can similarly construct a structure graph corresponding to a q tuple of distinct messages, where $q \geq 3$. It is needless to say that the structure graph constructed for a tuple of q distinct messages is a union of q message walks for each message $M_i, i \in [q]$.

Let $\mathcal{M} := (M_1, \ldots, M_q)$ denotes a tuple of q distinct messages and let $\mathcal{G}(\mathcal{M})$ denotes the set of all structure graphs corresponding to the tuple of messages $\mathcal{M}$ (by varying Π over Perm). For a fixed structure graph $V \in \mathcal{G}(\mathcal{M})$, let $\mathsf{Coll}(V)$ denote the set of all collisions in V. Now, we state the following two folklore results.

Proposition 1 (Lemma 2, [GPR14]). *For a fixed structure graph $V \in \mathcal{G}(\mathcal{M})$, $\Pr[G \leftarrow_{\$} \mathcal{G}(\mathcal{M}) : G = V] \leq 2^{-n|\mathsf{Coll}(V)|}$.*

Proposition 2 (Corollary 1, [JN16]). $\Pr[G \xleftarrow{\$} \mathcal{G}(\mathcal{M}) : |\mathsf{Coll}(G)| \geq a] \leq (\frac{\ell^2}{2^n})^a$, *where ℓ is the maximum number of message blocks in a single message among all messages in $\mathcal{M}$.*

Now, we define the following events: for a fixed tuple of q-distinct messages $\mathcal{M} := (M_1, \ldots, M_q)$, such that each message is at most ℓ blocks long, we sample a permutation Π uniformly at random from Perm and construct the structure graph $G^{\Pi}(\mathcal{M})$. Now we define three events as follows:

- Coll_1: $\exists i \in [q]$ such that the number of accidents in the i-th message walk in $G^{\Pi}(\mathcal{M})$ is at least 1.
- Coll_2: $\exists \{i, j\} \subseteq [q]$ such that the number of accidents between the i-th and the j-th message walk in $G^{\Pi}(\mathcal{M})$ is at least 2.
- Coll_3: $\exists \{i, j, k\} \subseteq [q]$ susch that the number of accidents between the i-th, the j-th and the k-th message walk in $G^{\Pi}(\mathcal{M})$ is at least 2.

It is easy to see that $\mathsf{Coll}_2 \Rightarrow \mathsf{Coll}_3$. We need the event Coll_2 in the analysis of 2K-ECBC-Hash and Coll_3 for the analysis of f9-Hash.

Using Proposition 2, we can easily bound the probabilities of each of these events as follows:

$$\Pr[\mathsf{Coll}_1] \leq \frac{q\ell^2}{2^n}, \quad \Pr[\mathsf{Coll}_2] \leq \frac{q^2\ell^4}{2^{2n}}, \quad \Pr[\mathsf{Coll}_3] \leq \frac{q^3\ell^4}{2^{2n}}. \tag{34}$$

Bad Hash Keys for 2K-ECBC-Hash. Recall that a hash key for 2K-ECBC-Hash is a pair of independent random permutations (Π_1, Π_2). Therefore, evaluation of 2K-ECBC-Hash on a fixed tuple of q distinct messages $\mathcal{M} := (M_1, \ldots, M_q)$ gives two structure graphs; one that is induced from the permutation Π_1, denoted as $G_1 := \mathcal{G}^{\Pi_1}(\mathcal{M})$, and the other is induced from the permutation Π_2, denoted as $G_2 := \mathcal{G}^{\Pi_2}(\mathcal{M})$. Now, we say a hash key (Π_1, Π_2) is **bad**, if either of the following holds:

(a) Coll_1 holds in either of G_1 or G_2.

(b) Coll_2 holds in either of G_1 or G_2.

We denote the set of all bad hash keys as $\mathcal{K}_{\text{bad}}^{\mathsf{ecbc}} \subseteq \mathsf{Perm} \times \mathsf{Perm}$. Therefore, from Eqn. (34) we bound the probability of Bad-Hash as follows:

$$\epsilon_{\text{bh}} := \Pr[\mathsf{Bad\text{-}Hash}] \quad \leq \quad 2\big(\frac{q^2\ell^4}{2^{2n}} + \frac{q\ell^2}{2^n}\big). \tag{35}$$

Bad Hash Keys for f9-Hash. Recall that a hash key for f9-Hash is a uniform random permutation Π. Therefore, evaluation of f9-Hash on a fixed tuple of q distinct messages $\mathcal{M} := (M_1, \ldots, M_q)$ gives a structure graph, that is induced from the permutation Π, denoted as $G := \mathcal{G}^{\Pi}(\mathcal{M})$. Now, we say a hash key Π is **bad** if either of the following holds:

(a) Coll_1 holds in G.

(b) Coll_3 holds in G.

We denote the set of all bad hash keys as $\mathcal{K}_{\text{bad}}^{\mathtt{f9}} \subseteq \mathsf{Perm}$. Therefore, from Eqn. (34), we bound the probability of $\mathsf{Bad\text{-}Hash} := \mathsf{Coll}_1 \vee \mathsf{Coll}_3$ as follows:

$$\epsilon_{\text{bh}} := \Pr[\mathsf{Bad\text{-}Hash}] \quad \leq \quad \frac{q\ell^2}{2^n} + \frac{q^3\ell^4}{2^{2n}}. \tag{36}$$

In the following, we bound the cover-free and block-wise universal advantage of 2K-ECBC-Hash and f9-Hash.

Theorem 12. 2K-ECBC-Hash *is a* $(\mathcal{K}_{\text{bad}}^{\mathtt{ecbc}}, 144\ell^2/2^{2n})$*-cover-free and* $(\mathcal{K}_{\text{bad}}^{\mathtt{ecbc}}, 12\ell^2/2^{2n}))$*-block-wise universal* DbH *function, assuming* $\ell \leq (2^{n-1}+1)/2$*. On the other hand,* f9-Hash *is a* $(\mathcal{K}_{\text{bad}}^{\mathtt{f9}}, 84\ell^2/2^{2n})$*-cover-free,* $(\mathcal{K}_{\text{bad}}^{\mathtt{f9}}, 3\ell^2/2^n)$*-block-wise universal and* $(\mathcal{K}_{\text{bad}}^{\mathtt{f9}}, 2/2^n)$*-colliding* DbH *function, assuming* $\ell \leq (2^{n-1}+2)/3$*.*

Similarly, we bound the weak-cover-free and the weak-block-wise universal advantage of ECBC-Hash and f9-Hash and collision of f9-hash as follows:

Theorem 13. ECBC-Hash *is a* $(\mathcal{K}_{\text{bad}}^{\mathtt{ecbc}}, 4/2^{2n})$*-weak-cover-free and* $(\mathcal{K}_{\text{bad}}^{\mathtt{ecbc}}, 2/2^n)$*-weak-block-wise universal* DbH *function, assuming* $\ell \leq (2^{n-1}+1)/2$*. On the other hand,* f9-Hash *is a* $(\mathcal{K}_{\text{bad}}^{\mathtt{f9}}, 18\ell^2/2^{2n})$*-weak-cover-free and* $(\mathcal{K}_{\text{bad}}^{\mathtt{f9}}, 3\ell^2/2^n)$*-weak-block-wise universal* DbH *function, assuming* $\ell \leq (2^{n-1}+2)/3$*.*

Proofs of Theorem 12 and Theorem 13 are deferred to Sect. 6.4. Assuming that these theorems hold, we now prove the PRF security of 2K-ECBC_Plus and 2Kf9 in Sect. 6.2 and that of ECBC_Plus and 3Kf9 in Sect. 6.3 respectively.

6.2 PRF Security of 2K-ECBC_Plus and 2Kf9

2K-ECBC_Plus and 2Kf9 are two sequential mode of block cipher based instantiations of two-keyed DbHtS. Algorithmic description of these two constructions are depicted in Fig. 6.2. The following two results show the PRF security bound of 2K-ECBC_Plus and 2Kf9.

$\mathsf{2K\text{-}ECBC\_Plus}(K_1, K_2, K_3, M)$	$\mathsf{2Kf9}(K_1, K_2, M)$
$(\Sigma, \Lambda) \leftarrow \mathsf{2K\text{-}ECBC\text{-}Hash}(K_1, K_2, M);$	$(\Sigma', \Lambda') \leftarrow \mathsf{f9\text{-}Hash}(K_1, M);$
$T \leftarrow E_{K_3}(\Sigma) \oplus E_{K_3}(\Lambda);$	$T \leftarrow E_{K_2}(\Sigma') \oplus E_{K_2}(\Lambda');$
return T;	**return** T;

Figure 6.2: Algorithm for 2K-ECBC_Plus 2Kf9.

Theorem 14 (PRF-Security of 2K-ECBC_Plus). *Let* $\mathcal{K}$ *and* $\mathcal{M}$ *be two non-empty finite sets. Let* $E : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ *be a block cipher. Then, any distinguisher with running time at most* t*, making* q *tuple of distinct messages each of at most* ℓ *blocks long, can distinguish* $\mathsf{2K\text{-}ECBC\_Plus}[E]$ *from an* n*-bit uniform random function by,*

$$\mathbf{Adv}^{\text{prf}}_{\mathsf{2K\text{-}ECBC\_Plus}[E]}(q,\ell,t) \leq 3\mathbf{Adv}^{\text{prp}}_{E}(\ell q, t') + \frac{2q^2\ell^4}{2^{2n}} + \frac{2q\ell^2}{2^n} + \frac{6q^3}{2^{2n}} + \frac{60q^3\ell^2}{2^{2n}} + \frac{q}{2^n},$$

where t' *is about* t *plus a time complexity necessary to compute* E *for* $\ell q + 2q$ *times and* $\ell \leq (2^{n-1}+1)/2$*.*

Proof of this theorem directly follows from part (*ii*) of Theorem 2, Theorem 12 and Eqn. (35).

Theorem 15 (PRF-Security of 2Kf9). *Let $\mathcal{K}$ and $\mathcal{M}$ be two non-empty finite sets. Let $E : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher. Then, any distinguisher with running time at most t, making q tuple of distinct messages each of at most ℓ blocks long, can distinguish $\mathsf{2Kf9}[E]$ from an n-bit uniform random function by,*

$$\mathbf{Adv}^{\mathrm{prf}}_{\mathsf{2Kf9}[E]}(q,\ell,t) \leq 2\mathbf{Adv}^{\mathrm{prp}}_{E}(\ell q, t') + \frac{q\ell^2}{2^n} + \frac{q^3\ell^4}{2^{2n}} + \frac{23q^3\ell^2}{2^{2n}} + \frac{6q^3}{2^{2n}} + \frac{3q}{2^n},$$

where t' is about t plus a time complexity necessary to compute E for $\ell q + 2q$ times $\ell \leq (2^{n-1}+2)/3$.

Proof of this theorem directly follows from part (*i*) of Theorem 2, Theorem 12 and Eqn. (36).

6.3 PRF Security of SUM-ECBC and 3kf9

SUM-ECBC and 3kf9 are two instantiations of the three-keyed DbHtS. Although, these constructions are the existing ones, as proposed by Yasuda [Yas10] and Zhang et al. [ZWSW12] respectively, for the sake of completeness of this paper, we state and prove the security of these two constructions in our setting. We recall these two constructions in Fig. 6.3. The following two results show the PRF security bound of SUM-ECBC and 3kf9.

$\mathsf{SUM\text{-}ECBC}(K_1,K_2,K_3,K_4,M)$	$\mathsf{3kf9}(K_1,K_2,K_3,M)$
$1:\quad (\Sigma',\Lambda') \leftarrow \mathsf{ECBC\text{-}Hash}(K_1,K_2,M);$	$1:\quad (\Sigma',\Lambda') \leftarrow \mathsf{f9\text{-}Hash}(K_1,M);$
$2:\quad T \leftarrow E_{K_3}(\Sigma') \oplus E_{K_4}(\Lambda');$	$2:\quad T \leftarrow E_{K_2}(\Sigma') \oplus E_{K_3}(\Lambda');$
return T;	**return** T;

Figure 6.3: Algorithm for SUM-ECBC and 3kf9.

Theorem 16 (PRF-Security of SUM-ECBC). *Let $\mathcal{K}$ and $\mathcal{M}$ be two non-empty finite sets. Let $E : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher. Then, any distinguisher with running time at most t, making q tuple of distinct messages each of at most ℓ blocks long, can distinguish $\mathsf{SUM\text{-}ECBC}[E]$ from an n-bit uniform random function by,*

$$\mathbf{Adv}^{\mathrm{prf}}_{\mathsf{SUM\text{-}ECBC}[E]}(q,\ell,t) \leq 4\mathbf{Adv}^{\mathrm{prp}}_{E}(\ell q, t') + \frac{2q^2\ell^4}{2^{2n}} + \frac{2q\ell^2}{2^n} + \frac{9q^3}{2^{2n}},$$

where t' is about t plus a time complexity necessary to compute E for $\ell q + 2q$ times and $\ell \leq (2^{n-1}+1)/2$.

Proof of this theorem directly follows from part (*iii*) of Theorem 2, Theorem 13 and Eqn. (35).

Theorem 17 (PRF-Security of 3kf9). *Let $\mathcal{K}$ and $\mathcal{M}$ be two non-empty finite sets. Let $E : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher. Then, any distinguisher with running time at most t, making q tuple of distinct messages each of at most ℓ blocks long, can distinguish $\mathsf{3kf9}[E]$ from an n-bit uniform random function by,*

$$\mathbf{Adv}^{\mathrm{prf}}_{\mathsf{3kf9}[E]}(q,\ell,t) \leq 3\mathbf{Adv}^{\mathrm{prp}}_{E}(\ell q, t') + \frac{q\ell^2}{2^n} + \frac{q^3\ell^4}{2^{2n}} + \frac{12q^3\ell^2}{2^{2n}} + \frac{2q^3}{2^{2n}},$$

where t' is about t plus a time complexity necessary to compute E for $\ell q + 2q$ times and $\ell \leq (2^{n-1}+2)/3$.

Proof of this theorem directly follows from part (*iii*) of Theorem 2, Theorem 13 and Eqn. (36).

Remark 6. The original security bound of the SUM-ECBC, as proven by Yasuda [Yas10], is roughly $q^3\ell^4/2^{2n}$ (we only mention the dominating term of the security bound). But, according to Theorem 16, the dominating term of the security bound of the SUM-ECBC is $q\ell^2/2^n + q^3/2^{2n}$, a substantially improved security bound than the existing one. On the other hand, for 3kf9, our proven security bound, i.e., roughly $q^3\ell^4/2^{2n}$, is infact worse than the existing one($q^3\ell^3/2^{2n}$) [ZWSW12]. However, we have identified that the existing bound of 3kf9 is flawed one and the root cause of the fallacy is discussed in details in the following subsection.

6.4 Proof of Theorem 12 and 13

In this section, we prove Theorem 12. In particular, we bound the cover-free and the block-wise universal advantage of 2K-ECBC-Hash and f9-Hash along with the maximum collision probability of f9-Hash. Before doing that, we state a technical result in the following, which will be useful for us to bound the cover-free advantage and the block-wise-universal advantage of 2K-ECBC-Hash and f9-Hash along with the maximum collision probability of f9-Hash when the hash key is sampled outside from the set of bad hash keys.

6.4.1 A Technical Result

Let $Y_1, \ldots, Y_t$ be t many variables which take values from some set $\mathcal{Y} \subseteq \{0,1\}^n$ and $\mathcal{L}$ be a set of system of linear equations $\{L_1, \ldots, L_s\}$ over $\{0,1\}^n$. For any $i \in [s]$, L_i represents a linear (or affine) equation of the form $a_{i,1}Y_1 \oplus \ldots \oplus a_{i,t}Y_t \oplus c_i = \mathbf{0}$, where $c_i, a_{i,j} \in \{0,1\}^n$ for all i, j. Let the rank of the system of equations $\mathcal{L}$ is r; maximum number of linearly independent equations present in $\mathcal{L}$.

Now, we know that if the system of equations $\mathcal{L}$ is consistent (i.e., at least one solution exists), then the probabilty that the system of equations holds is at most $|\mathcal{Y}|^{-r}$. Moreover, if $Y_1, \ldots, Y_t$ be t many wor variables which take values from $\mathcal{Y} \subseteq \{0,1\}^n$, the due to Lemma1, the above probability becomes at most $1/(|\mathcal{Y}| - t + r)_r$.

Now, we want to estimate the probability of a given system of linear equations $\mathcal{L}$ along with a given collision relation $\sim$. In other words, we want to estimate the number of solutions $(Y_1, \ldots, Y_t)$ that satisfy $\mathcal{L}$ and inducing the given collision relation $\sim$. Unlike before, in this case the rank of $\mathcal{L}$ does not help us to give a good estimation on the number of such solutions. As an example, we consider the following:

Example 1. Suppose $\sim$ is an equivalence relation over $\{1, \ldots, 6\}$ which partitions the index set as $\{\{1,4\},\{2\},\{3,5,6\}\}$, i.e., $1 \sim 4, 3 \sim 5 \sim 6$. Now, we want to compute the number of solutions $(Y_1, \ldots, Y_6)$ which satisfy the following system of linear equations $\mathcal{E}$ for some fixed constant c and the above defined equivalence relation $\sim$.

$$\mathcal{E} = \begin{cases} L_1 := Y_2 \oplus Y_3 \oplus Y_4 \oplus Y_5 \oplus c = \mathbf{0}, \\ L_2 := Y_1 \oplus Y_2 \oplus c = \mathbf{0}, \\ \mathcal{E}[\sim] := \sim^Y = \sim \end{cases}$$

Note that $\mathcal{E}[\sim]$ actually represents a system of equations of the form $Y_1 \oplus Y_4 = \mathbf{0}$, $Y_3 \oplus Y_5 = \mathbf{0}$, $Y_5 \oplus Y_6 = \mathbf{0}$ and some non-equations saying that Y_1, Y_2 and Y_3 are distinct. Even though L_1 and L_2 are linearly independent, we see that given these equalities of $\mathcal{E}[\sim]$, L_1 and L_2 are not linearly independent. Therefore, to obtain a solution, we choose (Y_1, Y_2, Y_3) in such a way so that Y_1, Y_2 and Y_3 are distinct to each other. Once we choose a triplet (Y_1, Y_2, Y_3) such that Y_1, Y_2 and Y_3 are distinct, the rest of the Y_i's are defined by the equalities of $\mathcal{E}[\sim]$. So, we write equations L_1 and L_2 in terms of Y_1, Y_2 and Y_3 (by eliminating the

other Y variables). After applying these substitutions, both L_1 and L_2 represents the same equation:

$$Y_1 \oplus Y_2 \oplus c = \mathbf{0}.$$

We call the above equation a **reduced equation**. Therefore, we see that the reduced form of L_1 and L_2 are not linearly independent even though those were before reduction.

Let $\sim$ be an equivalence relation over the set $[t]$ and thus partitions $[t]$ into the following disjoint classes: $C_1, \ldots, C_v$. From each class C_i, we choose an element $x_i = \min C_i$. To each $x \in [t]$, we associate a variable Y_x. Now, given any linear equation L over Y_x variables, we can replace every variable Y_x present in L by Y_{x_i} where $x \in C_i$ and then simplify the equation. The modified equation is called a **reduced equation**, denoted as $L^{\sim}$. Observe that the system of equations and non-equations $\mathcal{E}[\sim] \cup \{L\}$ is equivalent to the system of equations and non-equations $\mathcal{E}[\sim] \cup \{L^{\sim}\}$. Applying the above said reduction for more than one linear equations yields us a reduced system of linear equations $\mathcal{L}^{\sim} = \{L^{\sim} : L \in \mathcal{L}\}$. In other words, we apply the reduction to every equation individually and the above observation can be easily extended to multiple linear equations. More precisely, for a system of linear equations $\mathcal{L}$, $(\mathcal{E}[\sim] \cup \mathcal{L})$ is equivalent to $(\mathcal{E}[\sim] \cup \mathcal{L}^{\sim})$. One can also easily observe that the tuple $Y := (Y_1, \ldots, Y_t)$ satisfies $\mathcal{E}[\sim] \cup \mathcal{L}$ is equivalent to the tuple $Y^{\sim}$ satisfies $\mathcal{L}^{\sim}$, where $Y^{\sim}$ is the reduced tuple of Y after applying the relation $\sim$ on Y_i variables. Therefore, we have

Lemma 5. *Let $\mathcal{L}$ be a system of linear equations in variables $(Y_x)_{x\in[t]}$ and $\sim$ be an equivalence relation over $[t]$ with v many classes. If rank$(\mathcal{L}^{\sim}) = r$, then*

$$|\{Y^{\sim} : Y^{\sim} \text{ satisfies } \mathcal{L}^{\sim}\}| \leq (|\mathcal{Y}|)_{v-r}.$$

Proof of the lemma directly follows from the proof of Lemma 1 where the number of variables is now v instead of t.

Structure Graph and Collision Relation. For a fixed q tuple of distinct messages $\mathcal{M} := (M_1, \ldots M_q)$, a structure graph $G(\mathcal{M})$ gives a collision relation $\sim$ between the Y variables, where Y variables are the intermediate chaining values of CBC-MAC computation. In specific, whenever there is an accident in a single message walk or between more than one message walks, the corresponding Y variables are said to be related. This relation is called the **collision relation**, which one can easily see to be an equivalence relation. Let $\mathcal{G}(\mathcal{M})$ denotes the set of all possible structure graphs (by varying the underlying permutation Π).

Now, let us consider a system of linear equations $\mathcal{L}$ over $(Y_1, \ldots, Y_t)$ variables and with respect to a tuple of q distinct messages $\mathcal{M}$, we fix a structure graph $G(\mathcal{M})$, which is realized through these Y_i variables. The structure graph $G(\mathcal{M})$ yields a collision relation $\sim$ between the Y_i variables. Applying the collision relation $\sim$ to all the equations of $\mathcal{L}$ gives a reduced system of linear equations, denoted as $\mathcal{L}^{\sim}$. Moreover, each accident [9] in the structure graph $G(\mathcal{M})$ yields a linear equation of the form $Y_a \oplus Y_b = c$, and all such linear equations induced by the accidents in $G(\mathcal{M})$, are linearly independent. Let a be the total number of accidents in $G(\mathcal{M})$ and r be the rank of the system of equations $\mathcal{L}^{\sim} \cup \{Y_a \oplus Y_b = c\}$, where $\{Y_a \oplus Y_b = c\}$ is the set of all such linearly independent equations which are induced from the accidents in $G(\mathcal{M})$. We call this rank as the **joint rank**. Now, following Lemma 5, we have the following result.

Lemma 6. *Let us consider a structure graph $G(\mathcal{M}) \in \mathcal{G}(\mathcal{M})$ with respect to a fixed tuple of q distinct messages $\mathcal{M}$, realized through $(Y_1, \ldots, Y_t)$ variables. Let $\mathcal{L}$ be a system of linear equations in variables $Y := (Y_1, \ldots, Y_t)$ and $\sim$ be a collision relation over $[1, t]$ with v many classes, induced by $G(\mathcal{M})$. If the joint rank of $\mathcal{L}^{\sim} \cup \{Y_a \oplus Y_b = c\}$ is r, then*

$$\Pr[Y \text{ satisfies } \mathcal{L}, G = G(\mathcal{M})] \leq \frac{1}{(|\mathcal{Y}| - v + r)_r}.$$

[9] We use the term collision and accident interchangebaly.

Proof. To prove this result, the total number of solutions that satisfy $\mathcal{L}^{\sim}$ and all the linearly independent a many equations induced by the accidents in $G(\mathcal{M})$, are at most $(|\mathcal{Y}|)_{v-r}$ (follows from Lemma 5). Moreover, the total number of ways we can choose the variables are $(|\mathcal{Y}|)_v$ (keeping the distinctness of the variables). Dividing the former one by the latter yields the result. □

6.4.2 Cover-free and Block-wise universal Advantage of 2K-ECBC-Hash

We bound the cover-free and the block-wise universal advantage of 2K-ECBC-Hash when the hash key, i.e., the pair of independent random permutation (Π_1, Π_2), is sampled outside from $\mathcal{K}^{\mathsf{ecbc}}_{\mathrm{bad}}$.

Bounding Cover-free-advantage. To bound the cover-free advantage of 2K-ECBC-Hash, we first fix three distinct messages M_i, M_j and M_k. For brevity, we write $\Pr[\mathsf{CF}_{ijk} \text{ holds}, (\Pi_1, \Pi_2) \in (\mathsf{Perm} \times \mathsf{Perm}) \setminus \mathcal{K}^{\mathsf{ecbc}}_{\mathrm{bad}}]$ as P_{cf}. Now, we consider the two subsets of $\mathcal{G}(\mathcal{M})$: (i) $\mathcal{G}_0(\mathcal{M})$, which is the set of all structure graphs of $\mathcal{G}(\mathcal{M})$ such that there is no accident in between the i-th and the j-th message walks and (ii) $\mathcal{G}_1(\mathcal{M})$, which is the set of all structure graphs of $\mathcal{G}(\mathcal{M})$ such that there is exactly one accident in between the i-th and the j-th message walks. Now, by definition we have,

$$\begin{aligned}\mathsf{P}_{\mathrm{cf}} \;=\; \sum_{b,b' \in \{\mathbf{0},\mathbf{1}\}} \Big(& \Pr[\Sigma_i \oplus \Sigma_j = b, \Lambda_i \oplus \Lambda_k = b', G_1 \in \mathcal{G}_0(\mathcal{M}), G_2 \in \mathcal{G}_0(\mathcal{M})] \\ & + \Pr[\Sigma_i \oplus \Sigma_j = b, \Lambda_i \oplus \Lambda_k = b', G_1 \in \mathcal{G}_0(\mathcal{M}), G_2 \in \mathcal{G}_1(\mathcal{M})] \\ & + \Pr[\Sigma_i \oplus \Sigma_j = b, \Lambda_i \oplus \Lambda_k = b', G_1 \in \mathcal{G}_1(\mathcal{M}), G_2 \in \mathcal{G}_0(\mathcal{M})] \\ & + \Pr[\Sigma_i \oplus \Sigma_j = b, \Lambda_i \oplus \Lambda_k = b', G_1 \in \mathcal{G}_1(\mathcal{M}), G_2 \in \mathcal{G}_1(\mathcal{M})] \Big),\end{aligned}$$

where G_1 and G_2 are two independent structure graphs (when viewed as random variables defined over the sample space $\mathcal{G}(\mathcal{M})$). In other words, we may view that G_1 is induced by a random permutation Π_1 whereas G_2 is induced by an another random permutation Π_2, which is independent of Π_1. Moreover, the event $\Sigma_i \oplus \Sigma_j = b$ is independent over $\Lambda_i \oplus \Lambda_k = b'$ as the first event is induced by the randomness of Π_1 and the second event is induced by the randomness of Π_2, where Π_1 and Π_2 are two independent random permutations. Therefore, we write

$$\begin{aligned}\mathsf{P}_{\mathrm{cf}} \;=\; \sum_{b,b' \in \{\mathbf{0},\mathbf{1}\}} \Big(& \Pr[\Sigma_i \oplus \Sigma_j = b, G_1 \in \mathcal{G}_0(\mathcal{M})] \cdot \Pr[\Lambda_i \oplus \Lambda_k = b', G_2 \in \mathcal{G}_0(\mathcal{M})] \\ & + \Pr[\Sigma_i \oplus \Sigma_j = b, G_1 \in \mathcal{G}_0(\mathcal{M})] \cdot \Pr[\Lambda_i \oplus \Lambda_k = b', G_2 \in \mathcal{G}_1(\mathcal{M})] \\ & + \Pr[\Sigma_i \oplus \Sigma_j = b, G_1 \in \mathcal{G}_1(\mathcal{M})] \cdot \Pr[\Lambda_i \oplus \Lambda_k = b', G_2 \in \mathcal{G}_0(\mathcal{M})] \\ & + \Pr[\Sigma_i \oplus \Sigma_j = b, G_1 \in \mathcal{G}_1(\mathcal{M})] \cdot \Pr[\Lambda_i \oplus \Lambda_k = b', G_2 \in \mathcal{G}_1(\mathcal{M})] \Big) \quad (37)\end{aligned}$$

Analysis of Cases: Now, we analyze different cases. Basically, we will bound the following four probabilities:

$$\begin{aligned}&\text{(A) } \Pr[\Sigma_i \oplus \Sigma_j = \mathbf{0}, G_1 \in \mathcal{G}_0(\mathcal{M})], \quad \text{(B) } \Pr[\Sigma_i \oplus \Sigma_j = \mathbf{1}, G_1 \in \mathcal{G}_0(\mathcal{M})], \\ &\text{(C) } \Pr[\Sigma_i \oplus \Sigma_j = \mathbf{0}, G_1 \in \mathcal{G}_1(\mathcal{M})], \quad \text{(D) } \Pr[\Sigma_i \oplus \Sigma_j = \mathbf{1}, G_1 \in \mathcal{G}_1(\mathcal{M})].\end{aligned}$$

Bounding Case (A): It is easy to see that the event $\Sigma_i \oplus \Sigma_j = \mathbf{0}, G_1 \in \mathcal{G}_0(\mathcal{M})$ is an impossible event. Because we are considering those structure graphs in which there is no accident in between the i-th and the j-th message walks. But at the same time we are considering the event $\Sigma_i = \Sigma_j$, which itself is an accident between the i-th and the j-th message walks. Hence, the probability in this case is zero.

Bounding Case (B): $\Sigma_i \oplus \Sigma_j = \mathbf{1}$ is a non-trivial linear equation over Y variables. In specific, the equation is:

$$Y^i_{l_i} \oplus Y^j_{l_j} = \mathbf{1},$$

which holds with probability $1/(2^n - 2\ell + 1)$, when there is no accident in between the i-th and the j-th message walks. Moreover, the number of such structure graphs is only one, which is uniquely determined by the message tuple. Hence, the probability in this case is at most $1/(2^n - 2\ell + 1) \leq 2/2^n$ when $\ell \leq (2^{n-1} + 1)/2$.

Bounding Case (C): To compute this probability, we write

$$\Pr[\Sigma_i \oplus \Sigma_j = \mathbf{0}, G_1 \in \mathcal{G}_1(\mathcal{M})] = \sum_{V \in \mathcal{G}_1(\mathcal{M})} \Pr[\Sigma_i \oplus \Sigma_j = \mathbf{0}, G_1 = V]$$

The joint rank of the system of equations $\Sigma_i \oplus \Sigma_j = \mathbf{0}$ along with the equation induced from the accident, is at least 1. Therefore, from Lemma 6, we have

$$\Pr[\Sigma_i \oplus \Sigma_j = \mathbf{0}, G_1 = V] \leq \frac{1}{2^n - 2\ell + 1}.$$

Moreover, in this case the number of structure graphs with exactly one accident is 1. Therefore, the probability in this case is at at most $\frac{1}{2^n - 2\ell + 1} \leq \frac{2}{2^n}$, with the assumption that $\ell \leq (2^{n-1} + 1)/2$.

Bounding Case (D): To compute this probability, we write

$$\Pr[\Sigma_i \oplus \Sigma_j = \mathbf{1}, G_1 \in \mathcal{G}_1(\mathcal{M})] = \sum_{V \in \mathcal{G}_1(\mathcal{M})} \Pr[\Sigma_i \oplus \Sigma_j = \mathbf{1}, G_1 = V]$$

The joint rank of the system of equations $\Sigma_i \oplus \Sigma_j = \mathbf{1}$ along with the equation induced from the accident, is exactly 2 as the linear equation induced from the accident is linearly independent over the equation $\Sigma_i \oplus \Sigma_j = \mathbf{1}$. Therefore, from Lemma 6, we have

$$\Pr[\Sigma_i \oplus \Sigma_j = \mathbf{1}, G_1 = V] \leq \frac{1}{(2^n - 2\ell + 2)_2}.$$

Moreover, in this case the number of structure graphs with exactly one accident is $\binom{2\ell}{2} \leq 2\ell^2$. Therefore, the probability in this case is at most $\frac{2\ell^2}{(2^n - 2\ell + 2)_2} \leq \frac{2\ell^2}{(2^n - 2\ell + 1)^2} \leq \frac{8\ell^2}{2^{2n}}$ with the assumption $\ell \leq (2^{n-1} + 1)/2$.

All the above result equally holds when Σ_i and Σ_j are replaced by Λ_i and Λ_k respectively. Now, we split up Eqn. (37) and write as follows:

$$\begin{aligned}
\mathsf{P}_{\mathrm{cf}} \;=\; & \sum_{b,b' \in \{\mathbf{0},\mathbf{1}\}} \Pr[\Sigma_i \oplus \Sigma_j = b, G_1 \in \mathcal{G}_0(\mathcal{M})] \cdot \Pr[\Lambda_i \oplus \Lambda_k = b', G_2 \in \mathcal{G}_0(\mathcal{M})] \\
+ & \sum_{b,b' \in \{\mathbf{0},\mathbf{1}\}} \Pr[\Sigma_i \oplus \Sigma_j = b, G_1 \in \mathcal{G}_0(\mathcal{M})] \cdot \Pr[\Lambda_i \oplus \Lambda_k = b', G_2 \in \mathcal{G}_1(\mathcal{M})] \\
+ & \sum_{b,b' \in \{\mathbf{0},\mathbf{1}\}} \Pr[\Sigma_i \oplus \Sigma_j = b, G_1 \in \mathcal{G}_1(\mathcal{M})] \cdot \Pr[\Lambda_i \oplus \Lambda_k = b', G_2 \in \mathcal{G}_0(\mathcal{M})] \\
+ & \sum_{b,b' \in \{\mathbf{0},\mathbf{1}\}} \Pr[\Sigma_i \oplus \Sigma_j = b, G_1 \in \mathcal{G}_1(\mathcal{M})] \cdot \Pr[\Lambda_i \oplus \Lambda_k = b', G_2 \in \mathcal{G}_1(\mathcal{M})]. \quad (38)
\end{aligned}$$

By varying over all possible choices of b and b' and plugging-in the above derived bound in Eqn. (38), we have the following result:

$$\mathsf{P}_{\mathrm{cf}} \leq \frac{16}{2^{2n}} + \frac{64\ell^2}{2^{3n}} + \frac{64\ell^4}{2^{4n}} \leq \frac{144\ell^2}{2^{2n}}, \quad (39)$$

assuming $\ell \leq 2^{n-3}$. Hence we can set

$$\epsilon_{\rm cf}(3,\ell) = \frac{144\ell^2}{2^{2n}}. \tag{40}$$

Bounding Block-wise-universal advantage. To bound the block-wise universal advantage of 2K-ECBC-Hash, we first fix two distinct messages M_i and M_j. For brevity, we write $\Pr[\mathsf{UNIV}_{ij}$ holds, $(\Pi_1,\Pi_2) \in (\mathsf{Perm}\times\mathsf{Perm})\setminus \mathcal{K}^{\mathtt{ecbc}}_{\rm bad}]$ as $\mathsf{P}_{\rm univ}$. Now, as before we consider the two subsets of $\mathcal{G}(\mathcal{M})$: (i) $\mathcal{G}_0(\mathcal{M})$ and (ii) $\mathcal{G}_1(\mathcal{M})$. Now, by definition we can write,

$$\mathsf{P}_{\rm univ} = \max\Big(\sum_{b\in\{\mathbf{0},\mathbf{1}\}} \Pr[\Sigma_i\oplus\Sigma_j = b, G_1\in\mathcal{G}_{01}(\mathcal{M})], \sum_{b\in\{\mathbf{0},\mathbf{1}\}} \big(\Pr[\Lambda_i\oplus\Lambda_j = b, G_2\in\mathcal{G}_{01}(\mathcal{M})]\Big), \tag{41}$$

where $\mathcal{G}_{01}(\mathcal{M})$ denotes the set $\mathcal{G}_0(\mathcal{M})\cup\mathcal{G}_1(\mathcal{M})$. Now, by varying all possible choices of b and b' and plugging-in the above bound of Case (A)-Case (D) into Eqn. (41), we have $\mathsf{P}_{\rm univ} \leq \frac{4}{2^n} + \frac{16\ell^2}{2^{2n}}$ and hence we have

$$\epsilon_{\rm univ}(2,\ell) = \frac{4}{2^n} + \frac{8\ell^2}{2^{2n}} \leq \frac{12\ell^2}{2^n}. \tag{42}$$

The first part of Theorem 12 follows from Eqn. (40) and Eqn. (42).

6.4.3 Collision, Cover-free and Block-wise universal Advantage of f9-Hash

In this section, we bound the maximum collision probability, the cover-free advantage and the block-wise universal advantage of f9-Hash. Recall that, f9-Hash is not a block-separated DbH function and thus we require to bound its maximum collision probability (or equivalently the collision advantage) along with its cover-free and block-wise universal advantage.

Bounding Collision advantage. To bound this event, we first fix a message M_i and for brevity, we write $\Pr[\mathsf{COLL}_i$ holds, $\Pi\in\mathsf{Perm}\setminus\mathcal{K}^{\mathtt{f9}}_{\rm bad}]$ as $\mathsf{P}_{\rm coll}$. Let $\widetilde{\mathcal{G}}_0(\mathcal{M})$ be the set of all structure graphs of $\mathcal{G}(\mathcal{M})$ such that the number of accidents in the i-th message walk is zero, i.e., in a stucture graph of $\widetilde{\mathcal{G}}_0(\mathcal{M})$, there contains no accident within the i-th message walk. This says that, we need to bound the probability of the event when number of accidents in the message walk of M_i is zero. Now, by definition we have,

$$\mathsf{P}_{\rm coll} \;=\; \Pr[\Sigma'_i=\Lambda'_i, G\in\widetilde{\mathcal{G}}_0(\mathcal{M})] = \sum_{V\in\widetilde{\mathcal{G}}_0(\mathcal{M})} \Pr[\Sigma'_i=\Lambda'_i, G=V] \tag{43}$$

As there is no accident in the i-th message walk of V, it does not induce any linear equation. Therefore, the only linear equation we have due to $\Sigma'_i=\Lambda'_i$, which is non-trivial and hence the rank of the system of linear equations is one. In other words, the event $\Sigma'_i=\Lambda'_i$ implies the following non-trivial equation:

$$Y^i_1 \oplus \ldots Y^i_{l_i-1} = \mathbf{0},$$

which holds with probability at most $\frac{1}{2^n-\ell} \leq \frac{2}{2^n}$, with the assumption that $\ell\leq 2^{n-1}$. Moreover, the number of structure graphs with no accident in the i-th message walk is 1. Therefore, from Eqn. (43), we have $\mathsf{P}_{\rm coll}\leq\frac{2}{2^n}$ and hence we have,

$$\epsilon_{\rm coll} = \frac{2}{2^n}. \tag{44}$$

Bounding Cover-free-advantage. Fix three distinct messages M_i, M_j and M_k. As before, for brevity, we write $\Pr[\mathsf{CF}_{ijk}$ holds, $\Pi\in\mathsf{Perm}\setminus\mathcal{K}^{\mathtt{f9}}_{\rm bad}]$ as $\mathsf{P}_{\rm cf}$. Now, we consider the

two subsets of $\mathcal{G}(\mathcal{M})$: (i) $\mathcal{G}_2(\mathcal{M})$, which is the set of all structure graphs of $\mathcal{G}(\mathcal{M})$ such that the there is no accident in the i-th, the j-th and the k-th message walks and (ii) $\mathcal{G}_3(\mathcal{M})$, which is the set of all structure graphs of $\mathcal{G}(\mathcal{M})$ such that the there is exactly one accident in the i-th, the j-th and the k-th message walks. Let us denote $\mathcal{G}_2(\mathcal{M}) \cup \mathcal{G}_3(\mathcal{M})$ by $\mathcal{G}_{23}(\mathcal{M})$ and recall that $\mathcal{G}_{01}(\mathcal{M})$ denotes the set $\mathcal{G}_0(\mathcal{M}) \cup \mathcal{G}_1(\mathcal{M})$ where $\mathcal{G}_0(\mathcal{M})$ is the set of all structure graphs of $\mathcal{G}(\mathcal{M})$ such that there is no accident in the i-th and the j-th message walks and $\mathcal{G}_1(\mathcal{M})$ is the set of all structure graphs of $\mathcal{G}(\mathcal{M})$ such that there is exactly one accident in the i-th and the j-th message walks.Now, by definition we have,

$$\begin{aligned}\mathsf{P}_{\mathrm{cf}} &= \Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_k, G \in \mathcal{G}_{23}(\mathcal{M})] + \Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Lambda'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \\ &+ \Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Sigma'_k, G \in \mathcal{G}_{23}(\mathcal{M})] + \Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Sigma'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \\ &+ \Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})] \end{aligned} \tag{45}$$

Note that, unlike all the earlier constructions, f9-Hash is not block separated and hence to analyze its cover-free advantage, we need to consider all the possible ways that the cover free event can occur, as described in Sect. 3.3. Now, to bound P_{cf}, we state the following claim, the proof of which is given in Appendix A.

Claim 2. *Let M_i, M_j and M_k be any three distinct messages such that the maximum number of message blocks among all these three messages is ℓ. Then, we have,*

$$(a)\ \Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \le \frac{18\ell^2}{2^{2n}}; (b)\ \Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Lambda'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \le \frac{20\ell^2}{2^{2n}};$$
$$(c)\ \Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Sigma'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \le \frac{18\ell^2}{2^{2n}}; (d)\ \Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Sigma'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \le \frac{20\ell^2}{2^{2n}}.$$

Moreover, we also have $\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})] \le \frac{8\ell^2}{2^{2n}}$, where we assume $\ell \le (2^{n-1}+2)/3$.

Following Eqn. (45) and Claim 2 we have, $\mathsf{P}_{\mathrm{cf}} \le \frac{84\ell^2}{2^{2n}}$ and hence

$$\epsilon_{\mathrm{cf}}(3, \ell) = \frac{84\ell^2}{2^{2n}}. \tag{46}$$

Bounding Block-wise-universal advantage. Fix two distinct messages M_i and M_j. According to the definition of block-wise universal advantage for a pair of distinct messages we have the following:

$$\begin{aligned}\mathsf{P}_{\mathrm{univ}} = \max\Big(&\Pr[\Sigma'_i = \Sigma'_j, G \in \mathcal{G}_{01}(\mathcal{M})], \quad \Pr[\Lambda'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})], \\ &\Pr[\Sigma'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})]\Big),\end{aligned} \tag{47}$$

where $\mathsf{P}_{\mathrm{univ}}$ is the shorthand notation for $\Pr[\mathsf{UNIV}_{ij}$ holds, $\Pi \in \mathsf{Perm} \setminus \mathcal{K}^{\mathtt{f9}}_{\mathrm{bad}}]$ and $\mathcal{G}_{01}(\mathcal{M})$ denotes $\mathcal{G}_0(\mathcal{M}) \cup \mathcal{G}_1(\mathcal{M})$. Now, to bound $\mathsf{P}_{\mathrm{univ}}$, we state the following claim, proof of which is given in Appendix B.

Claim 3. *Let M_i, M_j be any two distinct messages such that the maximum number of message blocks among these two messages is ℓ. Then, we have,*

$$(a)\ \Pr[\Sigma'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})] \le \frac{3\ell^2}{2^n};\ (b)\ \Pr[\Sigma'_i = \Sigma'_j, G \in \mathcal{G}_{01}(\mathcal{M})] \le \frac{\ell^2}{2^n};$$
$$(c)\ \Pr[\Lambda'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})] \le \frac{3\ell^2}{2^n},$$

where we assume $\ell \le (2^{n-1}+1)/2$.

From Eqn. (47) and Claim 3, we have $\mathsf{P}_{\mathrm{univ}} \le \frac{3\ell^2}{2^n}$ and hence we have

$$\epsilon_{\mathrm{univ}}(2, \ell) = \frac{3\ell^2}{2^n}. \tag{48}$$

Remark 7. Unlike 2K-PMAC_Plus-Hash, 2K-LightMAC_Plus-Hash and 2K-ECBC-Hash, for the analysis of f9-Hash, we have avoided the use of fix0 and fix1 functions to make its DbH function block-separated. Hence, we dealt with all the cross collision events (among Σ and Λ) while analysing its cover-free and block-wise universal advantage along with the maximum collision probability. This is an example to show that we could have proved the beyond birthday bound security of all the ealier two-keyed variants without using fix0 and fix1 functions, but then the analysis would have been more involved and tedious.

6.4.4 Weak-cover-free and Weak-block-wise-universal Advantage of ECBC-Hash

To bound the weak cover-free advantage of ECBC-Hash, we only need the case $\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_k$, probability of which is bounded by $\frac{4}{2^{2n}}$. Similarly, to bound the block-wise universal advantage, we only need the case that $\Sigma'_i = \Sigma'_j$ or $\Lambda'_i = \Lambda'_j$, probability of each of them is bounded by $\frac{2}{2^n}$.

Hence, we have

$$\epsilon_{\rm cf}(3,\ell) = \frac{4}{2^{2n}}, \quad \epsilon_{\rm univ}(2,\ell) = \frac{2}{2^n}. \tag{49}$$

6.4.5 Weak cover-free and Weak-block-wise-universal Advantage of f9-Hash

Since the DbH function for 3kf9 and 2Kf9 is same, we have $\epsilon_{\rm bh} = \frac{q^3\ell^4}{2^{2n}} + \frac{q\ell^2}{2^n}$. Similar to ECBC-Hash, bounding the cover-free advantage requires us to analyze only the case $\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_k$, probability of which is bounded by $\frac{18\ell^2}{2^{2n}}$ (see (a) of Claim 2). Similarly, to bound the block-wise universal advantage, we only need the case that $\Sigma'_i = \Sigma'_j$ or $\Lambda'_i = \Lambda'_j$, the maximum probability of these two is atmost $\frac{3\ell^2}{2^n}$ (see (b) and (c) of Claim 3). Hence, we have

$$\epsilon_{\rm cf}(3,\ell) = \frac{18\ell^2}{2^{2n}}, \quad \epsilon_{\rm univ}(2,\ell) = \frac{3\ell^2}{2^n}.$$

6.5 Incorrectness of the Existing Security Bound of 3kf9

We have found that the existing bound of 3kf9 (i.e., $O(q^3\ell^3/2^{2n} + q\ell/2^n)$) proven in [ZWSW12] is incorrect. The main flaw of the security proof lies in bounding the cover-free advantage (Case D, [ZWSW12]) of the underlying DbH function (See Lemma 1 of [ZWSW12]) while making a flawed assumption about the probability of $\Sigma_i = \Sigma_j$ is at most $1/2^n$. But, this assumption is not true. $\Sigma_i = \Sigma_j$ is essentially the collision event of the CBC-MAC and the authors have assumed that the probability of this collision is at most $1/2^n$, missing many accidents from considerations. The correct bound of the collision probability of the CBC-MAC is $d(\ell)/2^n$ as shown in [BPR05], where $d(\ell)$ is the maximum number of divisors of l for any $l \leq \ell$.

Observe that, the security bound of 3kf9 proven in this paper (i.e., $O(q^3\ell^4/2^{2n})$) is beyond birthday in terms of q only (not in terms of both q and ℓ [10]). We believe that it would be very difficult, if not impossible, to show the beyond birthday security of 3kf9 and its reduced keyed variant, in terms of both q and ℓ. In our analysis, the term $q^3\ell^4/2^{2n}$ arises as we allow at most one accident for any choice of three messages. Hence, it makes the security bound to be beyond birthday in terms of q, but not in terms of ℓ. Generically, if one goes up to allowing a many accidents in any triplet of messages, then one needs to bound the probability for the number of accidents greater than equal to $a+1$ in any triplet of messages, which gives the bound $O(q^3\ell^{2(a+1)}/2^{(a+1)n})$; not beyond birthday

[10] As an example, if the security advantage happens to be 2^{-10}, then with block length $n = 128$ and $q = 2^{50}$, it limits the maximum value of the message length to 2^{24} blocks.

secure in terms of both q and ℓ. Henceforth, to avoid the bound which is beyond birthday only in terms of q, one needs to allow n many accidents for three distinct messages and then analyze the probability of its cover-free advantage. This seems really difficult as the number of possibilities of having n many accidents in three messages is huge (e.g., one may try to enumerate the number of cases for allowing only 3 accidents in three distinct messages).

6.6 Importance of the Set of Bad Hash Keys

We have seen that for some constructions, we have analyzed their cover-free and the block-wise universal advantage when the hash key was sampled from outside of the set of all bad hash keys. The importance of drawing the hash key from a good key space while analyzing the cover-free and block-wise universal advantage lies in obtaining an improved security bound for those constructions. For example, in the analysis of the cover-free advantage of the 2K-PMAC_Plus-Hash, if we had sampled the hash key from the set of all hash keys, we would have obtained a bound $O(q^3\ell^2/2^{2n})$. This is because, to bound its cover-free advantage for a triplet of distinct messages, we would have to consider the 3CollX event among the chosen three messages, which would happen with probability $\ell^2/2^{2n}$. This would get multiplied with q^3, makes the resultant bound of the order of $q^3\ell^2/2^{2n}$, a blow up of an extra ℓ factor in the security bound.

A much serious degradation of bound takes place for 2K-ECBC_Plus. If we had sample the hash key from the entire hash key space, then we would have obtained the bound $O(q^3\ell^4/2^{2n})$. This is because, to bound its cover-free advantage for a triplet of distinct messages, we would have to consider the Coll_2 event among the chosen three messages, which would happen with probability $\ell^4/2^{2n}$. This would get multiplied with q^3, makes the resultant bound of the order of $q^3\ell^4/2^{2n}$, a blow up of an extra q factor in the security bound.

7 Conclusion and Future Work

With a rapid growth of computing power, birthday attacks gradually become a practical threat to cryptographic algorithms. Therefore, designing modes that guarantees security beyond the birthday bound is active and promising. In this paper, we give a generic treatment of constructing the two-keyed and the three-keyed beyond birthday bound secure PRFs with an actual concrete instantiations, backed up with a proper security proof. This work immediately opens up two different directions of possible future works:

Open Problem I: A trivial question that comes to the mind is, *whether it is possible to extend this work to analyze the security of the single-keyed DbHtS, where the hash key would be same as the block cipher key used in the sum function?* It is well known that any generic composition result demands independent keys for each module, and whether the security holds even with the same key is non-trivial and requires a different approach. In the same line of reasoning, the security analysis of the single-keyed DbHtS would require a different approach and the proof may become quite complex and involved. Technically speaking, the analysis of the single-keyed DbHtS would require one to bound the collision between hash values and the intermediate block inputs (during the internal hash computation) along with the usual hash collisions. This enforces many more bad events. Analyzing these bad events and obtaining a generic result is non-trivial and is left as an open problem. In this regard, we would like to mention that Datta et al. [DDN+17] have shown the BBB security of single-keyed PMAC_Plus. We believe that using a similar approach, one can also prove BBB security of the single-keyed version of LightMAC_Plus. However, we think that proving the beyond birthday bound security of single-keyed version of 3kf9 is challenging and one needs to employ extreme care in analyzing the security of this construction.

Open Problem II: In a very recent work of Leurent et al. [LNS18], SUM-ECBC, PMAC_Plus, 3kf9, LightMAC_Plus and their reduced keyed-variant have been attacked with the query complexity $2^{3n/4}$. We believe that all these constructions can also be proven secured upto $2^{3n/4}$, and hence establishing the tightness of the bound. But to prove that, one needs to analyze (i) the rank of three linear equations (instead of two), which we believe is cumbersome and non-trivial to do and (ii) uplift the security of the sum of permutation result to $2^{3n/4}$.

Acknowledgements. We would like to thank Damian Vizár for his invaluable comments and suggestions in preparing the final draft. We would also like to thank all the anonymous reviewers of FSE 2019 for helping us improve the work. Nilanjan Datta performed part of his work during his PhD at Indian Statistical Institute, Kolkata. Avijit Dutta and Mridul Nandi are supported by R.C.Bose Centre for Cryptology and Security.

References

[AB99] Jee Hea An and Mihir Bellare. Constructing vil-macsfrom fil-macs: Message authentication under weakened assumptions. In *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, pages 252–269, 1999.

[BCK96] Mihir Bellare, Ran Canetti, and Hugo Krawczyk. Keying hash functions for message authentication. In *Advances in Cryptology - CRYPTO '96, 16th Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 1996, Proceedings*, pages 1–15, 1996.

[BI99] Mihir Bellare and Russell Impagliazzo. A tool for obtaining tighter security analyses of pseudorandom function based constructions, with applications to PRP to PRF conversion. *IACR Cryptology ePrint Archive*, 1999:24, 1999.

[BJKS93] Jürgen Bierbrauer, Thomas Johansson, Gregory Kabatianskii, and Ben J. M. Smeets. On families of hash functions via geometric codes and concatenation. In *Advances in Cryptology - CRYPTO '93, 13th Annual International Cryptology Conference, Santa Barbara, California, USA, August 22-26, 1993, Proceedings*, pages 331–342, 1993.

[BKL+07] Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, Christof Paar, Axel Poschmann, Matthew J. B. Robshaw, Yannick Seurin, and C. Vikkelsoe. PRESENT: an ultra-lightweight block cipher. In *Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop, Vienna, Austria, September 10-13, 2007, Proceedings*, pages 450–466, 2007.

[BKR98] Mihir Bellare, Ted Krovetz, and Phillip Rogaway. Luby-rackoff backwards: Increasing security by making block ciphers non-invertible. In *Advances in Cryptology - EUROCRYPT '98, International Conference on the Theory and Application of Cryptographic Techniques, Espoo, Finland, May 31 - June 4, 1998, Proceeding*, pages 266–280, 1998.

[BKR00] Mihir Bellare, Joe Kilian, and Phillip Rogaway. The security of the cipher block chaining message authentication code. *J. Comput. Syst. Sci.*, 61(3):362–399, 2000.

[BPP+17] Subhadeep Banik, Sumit Kumar Pandey, Thomas Peyrin, Yu Sasaki, Siang Meng Sim, and Yosuke Todo. GIFT: A small present - towards reaching

the limit of lightweight encryption. In *Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings*, pages 321–345, 2017.

[BPR05] Mihir Bellare, Krzysztof Pietrzak, and Phillip Rogaway. Improved security analyses for CBC macs. In *CRYPTO 2005*, pages 527–545, 2005.

[BR02] John Black and Phillip Rogaway. A block-cipher mode of operation for parallelizable message authentication. In *EUROCRYPT 2002*, pages 384–397, 2002.

[CLL+14] Shan Chen, Rodolphe Lampe, Jooyoung Lee, Yannick Seurin, and John P. Steinberger. Minimizing the two-round even-mansour cipher. In *Advances in Cryptology - CRYPTO 2014 - 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part I*, pages 39–56, 2014.

[CLP14] Benoit Cogliati, Rodolphe Lampe, and Jacques Patarin. The indistinguishability of the XOR of k permutations. In *Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers*, pages 285–302, 2014.

[CS14] Shan Chen and John P. Steinberger. Tight security bounds for key-alternating ciphers. In *Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings*, pages 327–350, 2014.

[dB93] Bert den Boer. A simple and key-economical unconditional authentication scheme. *Journal of Computer Security*, 2:65–72, 1993.

[DDN+17] Nilanjan Datta, Avijit Dutta, Mridul Nandi, Goutam Paul, and Liting Zhang. Single key variant of pmac_plus. *IACR Trans. Symmetric Cryptol.*, 2017(4):268–305, 2017.

[DHT17] Wei Dai, Viet Tung Hoang, and Stefano Tessaro. Information-theoretic indistinguishability via the chi-squared method. In *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III*, pages 497–523, 2017.

[DNP16] Avijit Dutta, Mridul Nandi, and Goutam Paul. One-key compression function based MAC with security beyond birthday bound. In *Information Security and Privacy - 21st Australasian Conference, ACISP 2016, Melbourne, VIC, Australia, July 4-6, 2016, Proceedings, Part I*, pages 343–358, 2016.

[DS11] Yevgeniy Dodis and John P. Steinberger. Domain extension for macs beyond the birthday barrier. In *EUROCRYPT 2011*, volume 6632 of *LNCS*, pages 323–342. Springer, 2011.

[GPPR12] Jian Guo, Thomas Peyrin, Axel Poschmann, and Matthew J. B. Robshaw. The LED block cipher. *IACR Cryptology ePrint Archive*, 2012:600, 2012.

[GPR14] Peter Gazi, Krzysztof Pietrzak, and Michal Rybár. The exact prf-security of NMAC and HMAC. In *Advances in Cryptology - CRYPTO 2014 - 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part I*, pages 113–130, 2014.

[HK97] Shai Halevi and Hugo Krawczyk. MMH: software message authentication in the gbit/second rates. In *Fast Software Encryption, 4th International Workshop, FSE '97, Haifa, Israel, January 20-22, 1997, Proceedings*, pages 172–189, 1997.

[IK03] Tetsu Iwata and Kaoru Kurosawa. OMAC: one-key CBC MAC. In *Fast Software Encryption, 2003*, pages 129–153, 2003.

[IM16] Tetsu Iwata and Kazuhiko Minematsu. Stronger security variants of GCM-SIV. *IACR Trans. Symmetric Cryptol.*, 2016(1):134–157, 2016.

[JN16] Ashwin Jha and Mridul Nandi. Revisiting structure graphs: Applications to CBC-MAC and EMAC. *J. Mathematical Cryptology*, 10(3-4):157–180, 2016.

[LNS18] Gaetan Leurent, Mridul Nandi, and Ferdinand Sibleyras. Generic attacks against beyond-birthday-bound macs. volume 2018, page 541, 2018.

[LPTY16] Atul Luykx, Bart Preneel, Elmar Tischhauser, and Kan Yasuda. A MAC mode for lightweight block ciphers. *IACR Cryptology ePrint Archive*, 2016:190, 2016.

[Luc00] Stefan Lucks. The sum of prps is a secure PRF. In *EUROCRYPT 2000*, pages 470–484, 2000.

[MP15] Bart Mennink and Bart Preneel. On the XOR of multiple random permutations. In *Applied Cryptography and Network Security - 13th International Conference, ACNS 2015, New York, NY, USA, June 2-5, 2015, Revised Selected Papers*, pages 619–634, 2015.

[Nai17] Yusuke Naito. Blockcipher-based macs: Beyond the birthday bound without message length. Cryptology ePrint Archive, Report 2017/852, 2017.

[NM08] Mridul Nandi and Avradip Mandal. Improved security analysis of PMAC. *J. Mathematical Cryptology*, 2(2):149–162, 2008.

[Pat98] Jacques Patarin. About feistel schemes with six (or more) rounds. In *Fast Software Encryption*, pages 103–121, 1998.

[Pat08a] Jacques Patarin. A proof of security in $o(2^n)$ for the benes scheme. In *AFRICACRYPT*, pages 209–220, 2008.

[Pat08b] Jacques Patarin. A proof of security in o(2n) for the xor of two random permutations. In *Information Theoretic Security, Third International Conference, ICITS 2008, Calgary, Canada, August 10-13, 2008, Proceedings*, pages 232–248, 2008.

[Pat08c] Jacques Patarin. The "Coefficients H" Technique. In *Selected Areas in Cryptography, SAC*, pages 328–345, 2008.

[Pat10] Jacques Patarin. Introduction to mirror theory: Analysis of systems of linear equalities and linear non equalities for cryptography. *IACR Cryptology ePrint Archive*, 2010:287, 2010.

[Pat13] Jacques Patarin. Security in $o(2^n)$ for the xor of two random permutations \\ - proof with the standard H technique -. *IACR Cryptology ePrint Archive*, 2013:368, 2013.

[Sho04] Victor Shoup. Sequences of games: a tool for taming complexity in security proofs. *IACR Cryptology ePrint Archive*, 2004:332, 2004.

[Tay93] Richard Taylor. An integrity check value algorithm for stream ciphers. In *Advances in Cryptology - CRYPTO '93, 13th Annual International Cryptology Conference, Santa Barbara, California, USA, August 22-26, 1993, Proceedings*, pages 40–48, 1993.

[Yas08] Kan Yasuda. A one-pass mode of operation for deterministic message authentication- security beyond the birthday barrier. In *Fast Software Encryption, 15th International Workshop, FSE 2008, Lausanne, Switzerland, February 10-13, 2008, Revised Selected Papers*, pages 316–333, 2008.

[Yas10] Kan Yasuda. The sum of CBC macs is a secure PRF. In *CT-RSA 2010*, pages 366–381, 2010.

[Yas11] Kan Yasuda. A new variant of PMAC: beyond the birthday bound. In *CRYPTO 2011*, pages 596–609, 2011.

[ZWSW12] Liting Zhang, Wenling Wu, Han Sui, and Peng Wang. 3kf9: Enhancing 3gpp-mac beyond the birthday bound. In *ASIACRYPT 2012*, pages 296–312, 2012.

Appendix

A Proof of Claim 2

In this section, we prove claim 2, where we analyse the probability of the events according to the structure graph notion. First we recall the statement of the claim:

Claim 2. *Let M_i, M_j and M_k be any three distinct messages such that the maximum number of message blocks among all these three messages is ℓ. Then, we have,*

$$(a)\ \Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \leq \frac{18\ell^2}{2^{2n}}; (b)\ \Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Lambda'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \leq \frac{20\ell^2}{2^{2n}};$$
$$(c)\ \Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Sigma'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \leq \frac{18\ell^2}{2^{2n}}; (d)\ \Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Sigma'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \leq \frac{20\ell^2}{2^{2n}}.$$

Moreover, we also have $\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})] \leq \frac{8\ell^2}{2^{2n}}$*, where we assume* $\ell \leq (2^{n-1}+2)/3$.

We bound the events as stated in claim 2 based on the randomness of the underlying permutation Π. We would like to first set up the following notational convention, which will be used in our subsequent analysis:

Notational Convention: Number of message blocks of i-th message M_i is denoted by l_i and the α-th message block of i-th message is denoted by $M_i[\alpha]$. Moreover, the block cipher output of α-th block of i-th message is denoted by Y^i_α. ℓ denotes the maximum number of message blocks among all q queries.

A.1 Bound of $\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_k, G \in \mathcal{G}_{23}(\mathcal{M})]$

We have fixed three distinct messages M_i, M_j and M_k each of the length at most ℓ blocks. Let $\mathcal{G}(M_i, M_j, M_k)$ denotes the set of all structure graphs corresponding to the fixed triple of messages M_i, M_j and M_k. Now, we write

$$\begin{aligned}\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \quad = \quad & \Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_k \wedge |\mathsf{Coll}(G)| = 0] \\ & + \Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_k \wedge |\mathsf{Coll}(G)| = 1]. \quad (50)\end{aligned}$$

Now, we analyse the probability of $\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_k$, when number of accident in the structure graph is 0 and 1 as follows:

Number of Accident =0. When the number of accidents in the structure graph is 0, then the probability of $\Sigma'_i = \Sigma'_j$ is 0 as the event itself implies either (a) at least one collision between a pair of messages or (b) a collision in either of the message walk of M_i or M_j. But since the number of accident is zero, $\Sigma'_i = \Sigma'_j$ is an impossible event and hence the probability of the joint event $\Sigma'_i = \Sigma'_j$ and $\Lambda'_i = \Lambda'_k$ is also 0. Therefore,

$$\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_k \wedge |\mathsf{Coll}(G)| = 0] = 0. \tag{51}$$

Number of Accident = 1. Let α be the length of the common suffix of M_i and M_j and β be the length of the common prefix of M_i and M_k. Then we have,

$$\Sigma'_i = \Sigma'_j \Rightarrow Y^i_{l_i-\alpha-1} \oplus Y^j_{l_j-\alpha-1} = M_i[l_i - \alpha] \oplus M_j[l_j - \alpha]. \tag{52}$$

Moreover, $\Lambda'_i = \Lambda'_k$ implies the following equation:

$$Y^i_{\beta+1} \oplus \ldots \oplus Y^i_{l_i} \oplus Y^k_{\beta+1} \oplus \ldots \oplus Y^k_{l_k} = 0. \tag{53}$$

Note that, the rank of Eqn. (52) and Eqn. (53) along with the equation induced from the accident is atleast 2. Therefore, from Lemma 6 we have

$$\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_k \wedge |\mathsf{Coll}(G)| = 1] \leq \frac{9\ell^2}{2(2^n - 3\ell + 2)_2} \leq \frac{18\ell^2}{2^{2n}}, \tag{54}$$

where we assume $\ell \leq (2^{n-1} + 2)/3$ and the number of structure graphs with exactly one accident among a triplet of messages is at most $\binom{3\ell}{2} \leq 9\ell^2/2$. Plug-in the bound of Eqn. (51) and Eqn. (54) into Eqn. (50), we have

$$\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \leq 0 + \frac{18\ell^2}{2^{2n}} \leq \frac{18\ell^2}{2^{2n}},$$

with the assumption $\ell \leq (2^{n-1} + 2)/3$.

A.2 Bound of $\Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Lambda'_k, G \in \mathcal{G}_{23}(\mathcal{M})]$

We bound the event in a similar way as we did in bounding $\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_k, G \in \mathcal{G}_{23}(\mathcal{M})]$. Let $\mathcal{G}(M_i, M_j, M_k)$ denotes the set of all structure graphs corresponding to the fixed triple of messages M_i, M_j and M_k. Now, we write

$$\begin{aligned}\Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Lambda'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \quad = \quad & \Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Lambda'_k \wedge |\mathsf{Coll}(G)| = 0] \\ & + \Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Lambda'_k \wedge |\mathsf{Coll}(G)| = 1]. \end{aligned} \tag{55}$$

Now, we analyse the probability of $\Sigma'_i = \Lambda'_j, \Lambda'_i = \Lambda'_k$, when number of accident in the structure graph is 0 and 1 as follows:

Number of Accident = 0. When number of accident is 0, then $\Sigma'_i = \Lambda'_j$ and $\Lambda'_i = \Lambda'_k$ implies the following two system of equations:

$$\begin{cases} Y^i_{l_i} \oplus Y^j_1 \oplus \ldots \oplus Y^j_{l_j} = 0 \\ Y^i_{\alpha+1} \oplus \ldots \oplus Y^i_{l_i} \oplus Y^k_{\alpha+1} \oplus \ldots \oplus Y^k_{l_k} = 0, \end{cases}$$

where α be the length of the common prefix of M_i and M_k. Now, if $l_i \neq \alpha + 1$, then the rank of the above system of equations is 2 for two random variables $Y^i_{l_i}$ and $Y^i_{\alpha+1}$. If

$\alpha+1 = l_i$, then also the rank of the above system of equations is 2 for two random variables $Y^j_{l_j}$ and $Y^k_{l_k}$. Thefore, in each of the cases, the rank is 2 and hence from Lemma 6, the probability that the above system of equations hold is $\frac{1}{(2^n-3\ell+2)_2}$. Moreover, the number of structure graphs with no accident is exactly 1. Therefore,

$$\Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Lambda'_k \wedge |\mathsf{Coll}(G)| = 0] \leq \frac{1}{(2^n - 3\ell + 2)_2} \leq \frac{4}{2^{2n}}, \tag{56}$$

with the assumption $\ell \leq (2^{n-1}+2)/3$.

Number of Accident = 1. Let α be the length of the common prefix of M_i and M_k. Then we have,

$$\Sigma'_i = \Lambda'_j \Rightarrow Y^i_{l_i} \oplus Y^j_1 \oplus \ldots \oplus Y^j_{l_j} = 0. \tag{57}$$

Moreover, $\Lambda'_i = \Lambda'_k$ implies the following equation:

$$Y^i_{\alpha+1} \oplus \ldots \oplus Y^i_{l_i} \oplus Y^k_{\alpha+1} \oplus \ldots \oplus Y^k_{l_k} = 0. \tag{58}$$

Note that, if the accident occurs in between the message walk of M_i and M_j then Eqn. (57) is non-trivial. Similarly, if the accident occurs in between the message walk of M_i and M_k then Eqn. (58) is non-trivial. Otherwise accident occurs in the message walk of M_j and M_k and in that case Eqn. (57) is non-trivial. Therefore, in either of the three cases the rank of system of equations Eqn. (57) and Eqn. (58) along with the equation induced from the accident is at least 2. Hence, from Lemma 6, the probability that the above system of equations hold is at most $\frac{1}{(2^n-3\ell+2)_2}$. Moreover, the number of structure graphs with exactly one accident in a triplet of messages is at most $\binom{3\ell}{2} \leq 9\ell^2/2$. Therefore,

$$\Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Lambda'_k \wedge |\mathsf{Coll}(G)| = 1] \leq \frac{9\ell^2}{2(2^n - 3\ell + 2)_2} \leq \frac{18\ell^2}{2^{2n}}, \tag{59}$$

with the assumption $\ell \leq (2^{n-1}+2)/3$. Plug-in the bound of Eqn. (56) and Eqn. (59) into Eqn. (55), we have

$$\Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Lambda'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \leq \frac{4}{2^{2n}} + \frac{18\ell^2}{2^{2n}} \leq \frac{2(9\ell^2+2)}{2^{2n}} \leq \frac{20\ell^2}{2^{2n}},$$

with the assumption $\ell \leq (2^{n-1}+2)/3$.

A.3 Bound of $\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Sigma'_k, G \in \mathcal{G}_{23}(\mathcal{M})]$

As before, we consider $\mathcal{G}(M_i, M_j, M_k)$ denotes the set of all structure graphs corresponding to the fixed triple of messages M_i, M_j and M_k. Now, we write

$$\begin{aligned}\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Sigma'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \quad &= \quad \Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Sigma'_k \wedge |\mathsf{Coll}(G)| = 0] \\ &\quad + \Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Sigma'_k \wedge |\mathsf{Coll}(G)| = 1]. \end{aligned} \tag{60}$$

Now, we analyse the probability of $\Sigma'_i = \Sigma'_j, \Lambda'_i = \Sigma'_k$, when number of accident in the structure graph is 0 and 1 as follows:

Number of Accident = 0. When number of accident is 0, then we have seen in Sect. A.1 that probability of $\Sigma'_i = \Sigma'_j$ is 0 unless $M_i = M_j$ but this is not possible as M_i and M_j are distinct. Therefore, when the number of accident is 0, then the probability of the joint event $\Sigma'_i = \Sigma'_j$ and $\Lambda'_i = \Sigma'_k$ is also 0. Therefore,

$$\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Sigma'_k \wedge |\mathsf{Coll}(G)| = 0] = 0. \tag{61}$$

Number of Accident = 1. Let α be the length of the common suffix of M_i and M_j. Then we have,

$$\Sigma'_i = \Sigma'_j \Rightarrow Y^i_{l_i-\alpha-1} \oplus Y^j_{l_j-\alpha-1} = M_i[l_i - \alpha] \oplus M_j[l_j - \alpha]. \tag{62}$$

Moreover, $\Lambda'_i = \Sigma'_k$ implies the following equation:

$$Y^i_1 \oplus \ldots \oplus Y^i_{l_i} \oplus Y^k_{l_k} = 0. \tag{63}$$

Note that, the rank of Eqn. (62) and Eqn. (63) along with the equation induced from the accident is atleast 2. Therefore, from Lemma 6 we have,

$$\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Sigma'_k \wedge |\mathsf{Coll}(G)| = 1] \leq \frac{9\ell^2}{2(2^n - 3\ell + 2)_2} \leq \frac{18\ell^2}{2^{2n}}, \tag{64}$$

where we assume $\ell \leq (2^{n-1} + 2)/3$ and the number of structure graphs with exactly one accident in a triplet of messages is at most $9\ell^2/2$. Plug-in the bound of Eqn. (61) and Eqn. (64) into Eqn. (60), we have

$$\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Sigma'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \leq 0 + \frac{18\ell^2}{2^{2n}} \leq \frac{18\ell^2}{2^{2n}},$$

with the assumption $\ell \leq (2^{n-1} + 2)/3$.

A.4 Bound of $\Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Sigma'_k, G \in \mathcal{G}_{23}(\mathcal{M})]$

As before, we consider $\mathcal{G}(M_i, M_j, M_k)$ denotes the set of all structure graphs corresponding to the fixed triple of messages M_i, M_j and M_k. Now, we write

$$\begin{aligned}\Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Sigma'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \quad &= \quad \Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Sigma'_k \wedge |\mathsf{Coll}(G)| = 0] \\ &\quad + \Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Sigma'_k \wedge |\mathsf{Coll}(G)| = 1]. \end{aligned} \tag{65}$$

Now, we analyse the probability of $\Sigma'_i = \Lambda'_j, \Lambda'_i = \Sigma'_k$, when number of accident in the structure graph is 0 and 1 as follows:

Number of Accident = 0. When number of accident is 0, then $\Sigma'_i = \Lambda'_j$ and $\Lambda'_i = \Sigma'_k$ implies the following two system of equations:

$$\begin{cases} Y^i_{l_i} \oplus Y^j_1 \oplus \ldots \oplus Y^j_{l_j} = 0 \\ Y^k_{l_k} \oplus Y^i_{l_i} \oplus \ldots \oplus Y^i_{l_i} = 0. \end{cases}$$

Note that, the rank of the above system of equations is 2 for random variables $Y^i_{l_i}$ and $Y^k_{l_k}$. Thefore, due to Lemma 6, the probability that the above system of equations hold is $\frac{1}{(2^n-3\ell+2)_2}$. Moreover, the number of structure graphs with no accident is exactly 1. As a result, we have,

$$\Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Sigma'_k \wedge |\mathsf{Coll}(G)| = 0] \leq \frac{1}{(2^n - 3\ell + 2)_2} \leq \frac{4}{2^{2n}}, \tag{66}$$

with the assumption $\ell \leq (2^{n-1} + 2)/3$.

The argument for bounding the event when number of accident is one is similar to that of in Sect. A.2 while bounding $\Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Lambda'_k \wedge |\mathsf{Coll}(G)| = 1]$. If the accident occurs in the message walk of M_i and M_j or in between of M_j and M_k then $Y^i_{l_i} \oplus Y^j_1 \oplus \ldots \oplus Y^j_{l_j} = 0$

is a non-trivial equation. Similarly, if the accident is between message walk of M_i and M_k then $Y^k_{l_k} \oplus Y^i_{l_i} \oplus \ldots \oplus Y^i_{l_i} = 0$ is a non-trivial one. Therefore, in each cases the above system of equations along with the equation induced from the accident has rank at least 2 and hence, from Lemma 6, the probability of the event when number of accident is one is bounded by $\frac{1}{(2^n-3\ell+2)_2}$. Moreover, the number of structure graphs with exactly one accident among a triplet of messages is at most $9\ell^2/2$. Therefore,

$$\Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Sigma'_k \wedge |\mathsf{Coll}(G)| = 1] \leq \frac{9\ell^2}{2(2^n - 3\ell + 2)_2} \leq \frac{18\ell^2}{2^{2n}}, \tag{67}$$

with the assumption $\ell \leq (2^{n-1} + 2)/3$. Plug-in the bound of Eqn. (66) and Eqn. (67) into Eqn. (65), we have

$$\Pr[\Sigma'_i = \Lambda'_j, \Lambda'_i = \Sigma'_k, G \in \mathcal{G}_{23}(\mathcal{M})] \leq \frac{4}{2^{2n}} + \frac{18\ell^2}{2^{2n}} \leq \frac{2(9\ell^2+2)}{2^{2n}} \leq \frac{20\ell^2}{2^{2n}},$$

with the assumption $\ell \leq (2^{n-1} + 2)/3$.

A.5 Bound of $\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})]$

We have fixed two distinct messages M_i, M_j and M_k each of the length at most ℓ blocks. Let $\mathcal{G}(M_i, M_j)$ denotes the set of all structure graphs corresponding to the fixed pair of messages M_i and M_j. Now, we write

$$\begin{aligned}\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})] \quad &= \quad \Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_j \wedge |\mathsf{Coll}(G)| = 0] \\ &\quad + \Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_j \wedge \mathsf{Coll}(G)| = 1]. \end{aligned} \tag{68}$$

As argued before that when the number of accidents in the structure graph is zero, then $\Sigma'_i = \Sigma'_j$ is an impossible event and therefore, the probability of $\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_j$ is zero.

Number of Accident = 1. Let α and β be the length of the common suffix and prefix of M_i and M_j respectively. Then we have,

$$\Sigma'_i = \Sigma'_j \Rightarrow Y^i_{l_i-\alpha-1} \oplus Y^j_{l_j-\alpha-1} = M_i[l_i - \alpha] \oplus M_j[l_j - \alpha]. \tag{69}$$

Moreover, $\Lambda'_i = \Lambda'_j$ implies the following equation:

$$Y^i_{\beta+1} \oplus \ldots \oplus Y^i_{l_i} \oplus Y^j_{\beta+1} \oplus \ldots \oplus Y^j_{l_j} = 0. \tag{70}$$

Note that, the rank of Eqn. (69) and Eqn. (70) along with the equation induced from the accident is atleast 2. Therefore, from Lemma 6 we have

$$\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_j \wedge |\mathsf{Coll}(G)| = 1] \leq \frac{2\ell^2}{(2^n - 3\ell + 2)_2} \leq \frac{8\ell^2}{2^{2n}}, \tag{71}$$

where we assume $\ell \leq (2^{n-1} + 2)/3$ and the number of structure graphs with exactly one accident among a pair of messages is at most $2\ell^2$. Plug-in the bound of Eqn. (71) into Eqn. (68), we have

$$\Pr[\Sigma'_i = \Sigma'_j, \Lambda'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})] \leq 0 + \frac{8\ell^2}{2^{2n}} \leq \frac{8\ell^2}{2^{2n}},$$

with the assumption $\ell \leq (2^{n-1} + 2)/3$.

B Proof of Claim 3

In this section, we prove claim 3. Again, we first recall the statement of the claim:

Claim 3. *Let M_i, M_j be any two distinct messages such that the maximum number of message blocks among these two messages is ℓ. Then, we have,*

$$(a)\ \Pr[\Sigma'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})] \leq \frac{3\ell^2}{2^n};\ (b)\ \Pr[\Sigma'_i = \Sigma'_j, G \in \mathcal{G}_{01}(\mathcal{M})] \leq \frac{\ell^2}{2^n};$$
$$(c)\ \Pr[\Lambda'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})] \leq \frac{3\ell^2}{2^n},$$

where we assume $\ell \leq (2^{n-1}+1)/2$.

Like proof of claim 2, we analyse the probability of the events according to the structure graph notion. Hence, we bound the events as stated in claim 3 based on the randomness of the underlying permutation Π. Using the same notational convention as developed in Sect. A, we bound the following:

B.1 Bound of $\Pr[\Sigma'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})]$

We fix two distinct messages M_i and M_j. We denote the set of all structure graphs corresponding to M_i and M_j. by $\mathcal{G}(M_i, M_j)$. Now, we write

$$\begin{aligned}\Pr[\Sigma'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})] &= \Pr[\Sigma'_i = \Lambda'_j \wedge |\mathsf{Coll}(G)| = 0] + \Pr[\Sigma'_i = \Lambda'_j \wedge |\mathsf{Coll}(G)| = 1] \\ &\leq \Pr[\Sigma'_i = \Lambda'_j \wedge |\mathsf{Coll}(G)| = 0] + \Pr[|\mathsf{Coll}(G)| = 1] \\ &\leq \Pr[\Sigma'_i = \Lambda'_j \wedge |\mathsf{Coll}(G)| = 0] + \frac{\ell^2}{2^n}, \end{aligned} \tag{72}$$

where the last inequality follows from Proposition 2. Now, we analyse the probability of $\Sigma'_i = \Lambda'_j$, when number of accident in the structure graph is 0 as follows:

Number of Accident = 0. We analyse this case into different subcases as follows:

- (i) Without loss of generality we assume M_j is a prefix of M_i. In this case, the event $\Sigma'_i = \Lambda'_j$ implies the following non-trivial equation:

$$Y^i_1 \oplus \ldots Y^i_{l_j} \oplus Y^i_{l_i} = 0,$$

which holds with probability at most $\frac{1}{2^n - 2\ell + 1} \leq \frac{2}{2^n}$, follows from Lemma 1, with the assumption $\ell \leq (2^{n-1}+1)/2$.

- (ii) When none of the messages is a prefix of another. Without loss of generality, we assume $l_i \geq l_j$ and p be the length of the common prefix of M_i and M_j. Now, the event $\Sigma'_i = \Lambda'_j$ implies the following non-trivial equation:

$$Y^i_1 \oplus \ldots Y^i_p \oplus Y^j_{p+1} \oplus Y^j_{l_j} \oplus Y^i_{l_i} = 0,$$

which holds with probability at most $\frac{1}{2^n - 2\ell + 1} \leq \frac{2}{2^n}$, follows from Lemma 1, with the assumption $\ell \leq (2^{n-1}+1)/2$.

Plug-in the bound into Eqn. (72) we obtain

$$\Pr[\Sigma'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})] \leq \frac{1}{2^n - 2\ell + 1} + \frac{\ell^2}{2^n} \leq \frac{\ell^2 + 2}{2^n} \leq \frac{3\ell^2}{2^n},$$

where we assume $\ell \leq (2^{n-1}+1)/2$.

B.2 Bound of $\mathbf{Pr}[\Sigma'_i = \Sigma'_j, G \in \mathcal{G}_{01}(\mathcal{M})]$

Let us fix two distinct messages M_i and M_j. Let $\mathcal{G}(M_i, M_j)$ denotes the set of all structure graphs corresponding to M_i and M_j. Now, we write

$$\begin{aligned}\Pr[\Sigma'_i = \Sigma'_j, G \in \mathcal{G}_{01}(\mathcal{M})] &= \Pr[\Sigma'_i = \Sigma'_j \wedge |\mathsf{Coll}(G)| = 0] + \Pr[\Sigma'_i = \Sigma'_j \wedge |\mathsf{Coll}(G)| = 1] \\ &\leq \Pr[\Sigma'_i = \Sigma'_j \wedge |\mathsf{Coll}(G)| = 0] + \Pr[|\mathsf{Coll}(G)| = 1] \\ &\leq \Pr[\Sigma'_i = \Sigma'_j \wedge |\mathsf{Coll}(G)| = 0] + \frac{\ell^2}{2^n}, \end{aligned} \tag{73}$$

where the last inequality follows from Proposition 2. Now, we analyse the probability of $\Sigma'_i = \Sigma'_j$, when number of accident in the structure graph is 0 as follows:

Number of Accident = 0. As argued in Sect. A.1, when the number of accident is 0, then the probability of $\Sigma'_i = \Sigma'_j$ is 0 as the event itself implies either (a) at least one collision between a pair of messages or (b) a collision in either of the message walk of M_i or M_j. But since we condition on the number of accident is zero, $\Sigma'_i = \Sigma'_j$ is an impossible event. Therefore,

$$\Pr[\Sigma'_i = \Sigma'_j, G \in \mathcal{G}_{01}(\mathcal{M})] \leq 0 + \frac{\ell^2}{2^n} \leq \frac{\ell^2}{2^n}.$$

B.3 Bound of $\mathbf{Pr}[\Lambda'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})]$

We follow the similar analysis as we did for bounding $\Pr[\Sigma'_i = \Sigma'_j, G \in \mathcal{G}_{01}(\mathcal{M})]$. $\mathcal{G}(M_i, M_j)$ denotes the set of all structure graphs corresponding to the fixed pair of messages M_i and M_j. Now, we write

$$\begin{aligned}\Pr[\Lambda'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})] &= \Pr[\Lambda'_i = \Lambda'_j \wedge |\mathsf{Coll}(G)| = 0] + \Pr[\Lambda'_i = \Lambda'_j \wedge |\mathsf{Coll}(G)| = 1] \\ &\leq \Pr[\Lambda'_i = \Lambda'_j \wedge |\mathsf{Coll}(G)| = 0] + \Pr[|\mathsf{Coll}(G)| = 1] \\ &\leq \Pr[\Lambda'_i = \Lambda'_j \wedge |\mathsf{Coll}(G)| = 0] + \frac{\ell^2}{2^n}, \end{aligned} \tag{74}$$

where the last inequality follows from Proposition 2. Now, we analyse the probability of $\Lambda'_i = \Lambda'_j$, when number of accident in the structure graph is 0 as follows:

Number of Accident = 0. We analyse this case into different subcases as follows:

- (i) Without loss of generality we assume that M_j is a prefix of M_i. Then the event $\Lambda'_i = \Lambda'_j$ implies the following non-trivial equation:

$$Y^i_{l_j+1} \oplus \ldots \oplus Y^i_{l_i} = 0,$$

probability of which is bounded by $\frac{1}{2^n - 2\ell + 1}$, follows from Lemma 1, with the assumption $\ell \leq (2^{n-1} + 1)/2$.

- (ii) when none of the message is a prefix of another. Let us assume $l_i \geq l_j$. Let us assume, p is the length of the common prefix of M_i and M_j. Now, the event $\Lambda'_i = \Lambda'_j$ implies the following non-trivial equation

$$Y^j_{p+1} \oplus \ldots \oplus Y^j_{l_j} \oplus Y^i_{p+1} \ldots \oplus Y^i_{l_i} = 0,$$

probability of which is bounded by $\frac{1}{2^n - 2\ell}$, follows from Lemma 1, with the assumption $\ell \leq (2^{n-1} + 1)/2$. Note that, if $l_i = l_j$ then $p < l_j - 1$ otherwise the probaility would become zero.

Plug-in the bound into Eqn. (74) we obtain

$$\Pr[\Lambda'_i = \Lambda'_j, G \in \mathcal{G}_{01}(\mathcal{M})] \leq \frac{1}{2^n - 2\ell + 1} + \frac{\ell^2}{2^n} \leq \frac{\ell^2 + 2}{2^n} \leq \frac{3\ell^2}{2^n},$$

where we assume $\ell \leq (2^{n-1} + 1)/2$.