IACR Transactions on Symmetric Cryptology
ISSN 2519-173X, Vol. 0, No. 0, pp. 0–0. DOI:10.46586/tosc.v0.i0.0-0

# Improved Security Bound of (E/D)WCDM

Nilanjan Datta and Avijit Dutta and Kushankur Dutta

Institute for Advancing Intelligence, TCG-CREST, Kolkata, India.
nilanjan.datta@tcgcrest.org,avijit.dutta@tcgcrest.org,kushankur.dutta@tcgcrest.org

**Abstract.** In CRYPTO'16, Cogliati and Seurin proposed a block cipher based nonce based MAC, called *Encrypted Wegman-Carter with Davies-Meyer* (EWCDM), that gives $2n/3$ bit MAC security in the nonce respecting setting and $n/2$ bit security in the nonce misuse setting, where $n$ is the block size of the underlying block cipher. However, this construction requires two independent block cipher keys. In CRYPTO'18, Datta et al. came up with a single-keyed block cipher based nonce based MAC, called *Decrypted Wegman-Carter with Davies-Meyer* (DWCDM), that also provides $2n/3$ bit MAC security in the nonce respecting setting and $n/2$ bit security in the nonce misuse setting. However, the drawback of DWCDM is that it takes only $2n/3$ bit nonce. In fact, authors have shown that DWCDM cannot achieve beyond the birthday bound security with $n$ bit nonces. In this paper, we prove that DWCDM with $3n/4$ bit nonces provides MAC security up to $O(2^{3n/4})$ MAC queries against all nonce respecting adversaries. We also improve the MAC bound of EWCDM from $2n/3$ bit to $3n/4$ bit. The backbone of these two results is a refined treatment of extended mirror theory that systematically estimates the number of solutions to a system of bivariate affine equations and non-equations, which we apply on the security proofs of the constructions to achieve $3n/4$ bit security.

**Keywords:** Wegman Carter · Extended Mirror Theory · Nonce Based MAC · EWCDM · DWCDM.

## 1 Introduction

In the era of digital transmissions, cryptographic algorithms are used to authenticate the transmitted message over an insecure communication channel. Message Authentication Code, or in short MAC, is a popular symmetric key cryptographic primitive that plays an important role to enable two legitimate parties (having access to a shared secret key) to authenticate their transmissions. One of the natural approaches to authenticate a message $M$ is to generate a random string of a constant size, which is used to mask the hash of the message that needs to be authenticated. The disadvantage of the scheme is that for every message that needs to be authenticated, it requires generating fresh constant sized random strings. To eliminate this one-time authentication problem, Brassard [Bra82] suggested to use a pseudorandom generator that generates a sequence of pseudorandom strings from a short master key. But in some applications, messages may come in arbitrary order due to network latency. Therefore, a direct means of computing the pseudorandom string (instead of sequentially computing the string) is much desired. Although Brassard suggested the use of Blum-Blum Shub generator [Bra82] for directly computing the pseudorandom string, a pseudorandom function (PRF) was a natural choice for this purpose to directly compute the pseudorandom string out of a nonce, a non-repeating value. This construction is known as *Wegman-Carter* (WC) MAC, defined as follows:

$$\mathsf{WC}_{\mathsf{F},\mathsf{H}}(\nu, M) := \mathsf{F}_k(\nu) \oplus \mathsf{H}_{k_h}(M),$$

where $\nu$ is the nonce. WC is a powerful MAC that provides the security guarantee up to the differential probability of the underlying hash function (also known as *almost-xor-universal* advantage [1]) when a nonce does not repeat in the queries (also known as the *nonce respecting setting*). The primary disadvantage of the WC construction is that it is completely broken when nonce repeats at least once (in other words, *nonce misuse setting*). In fact one can mount universal forgery in the case of a single repetition of a nonce. Due to the lack of availability of practical PRFs, Shoup suggested that F can be replaced by a block cipher E. This resulting MAC is known as *Wegman-Carter-Shoup* (WCS). However, unlike WC MAC, the security of WCS drops down to the birthday limit in the number of queries when a nonce is not repeated, and it also suffers from the problem of providing adequate security in the nonce-misuse setting. To achieve security in the nonce misuse setting, Cogliati and Seurin [CS16] proposed *Encrypted Wegman-Carter* (EWC) construction that offers birthday bound security in the nonce misuse setting but provides a high security in the nonce respecting setting. EWC is defined as follows:

$$\mathsf{EWC}_{\mathsf{E},\mathsf{F},\mathsf{H}}(\nu, M) := \mathsf{E}_{k_2}(\mathsf{F}_{k_1}(\nu) \oplus \mathsf{H}_{k_h}(M)).$$

However, replacing the PRF F of EWC with a block cipher E makes its security drop to the birthday bound in the nonce respecting setting. To alleviate the problem, one can instantiate the PRF F of EWC construction with the xor of two permutations (XoP) construction [BKR98, Luc00]. Since XoP has been proved to be optimally secure [DHT17], the resulting construction provides optimal MAC security in the nonce respecting setting. Although the construction provides high MAC security, it requires three block cipher calls altogether. Interestingly, Cogliati and Seurin [CS16] were able to reduce the number of block cipher calls by 1 through their construction *Encrypted Wegman-Carter with Davies-Meyer* (EWCDM), where they have instantiated the PRF F with the Davies-Meyer construction. They have shown that EWCDM provides $2n/3$ bit MAC security in the nonce respecting setting and $n/2$ bit security in the nonce-misuse setting.

## 1.1 Encrypted Wegman-Carter with Davies-Meyer

In CRYPTO'16, Cogliati and Seurin [CS16] proposed EWCDM, a nonce-based MAC, defined as follows:

$$\mathsf{EWCDM}_{\mathsf{E},\mathsf{H}}(\nu, M) = \mathsf{E}_{k_2}\big(\mathsf{E}_{k_1}(\nu) \oplus \nu \oplus \mathsf{H}_{k_h}(M)\big),$$

where $\nu$ is the nonce and $M$ is the message. Note that EWCDM uses two independent block cipher keys, $k_1$ and $k_2$, and an independent hash key $k_h$ for the AXU hash function. Authors have proved that EWCDM is secure against all nonce-respecting adversaries[2] that make $q_m \ll 2^{2n/3}$ MAC queries and $q_v \ll 2^n$ verification queries.
They have also shown $n/2$ bit security of EWCDM against nonce-misuse adversaries. It is interesting to note here that, although the Davies-Meyer (DM) construction

$$\mathsf{DM}_{[\mathsf{E}]}(\nu) = \mathsf{E}_k(\nu) \oplus \nu,$$

is not a beyond birthday bound secure PRF, but encrypting its output after masking with the hash of a message makes the construction a beyond birthday bound secure MAC.
Later in CRYPTO'17, Mennink and Neves [MN17] proved $n$ bit PRF security of EWCDM in the nonce respecting setting using the result of *Mirror theory for general* $\xi_{\max}$ [3] [Pat05,

[1] An almost-xor universal (axu) hash function is a keyed hash function such that for any two distinct messages, the probability, over a random draw of a hash key, of the hash differential being equal to a specific output is small.

[2] Adversaries who never repeat the same value of $\nu$ in their MAC queries.

[3] $\xi_{\max}$ refers to the block maximality of a given system of bivariate affine equations [Pat10]. For a given system of bivariate affine equations, we say equation $i$ and equation $j$ are related if they have at least one common variable. Then $\xi_{\max}$ = the maximum number of related equations +1.

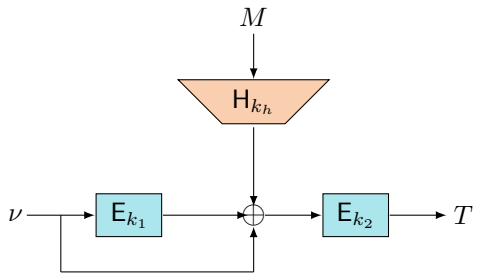


**Figure 1.1:** Encrypted Wegman-Carter with Davies-Meyer Construction.

Pat10], and mentioned that the analysis is extended to the analysis for the unforgeability of the construction. The trick involved in proving the optimal security of EWCDM is by replacing the last block cipher call with its inverse. This subtle change does not make any difference in the output distribution and as a bonus, it trivially allows one to view an evaluation of $T = \mathsf{EWCDM}(\nu, M)$ as the xor of two permutations in the middle of the function (or in general a bi-variate affine equation [4]), i.e.,

$$\mathsf{E}_{k_1}(\nu) \oplus \mathsf{E}_{k_2}(T) = \nu \oplus \mathsf{H}_{k_h}(M).$$

It is only this feature which is captured by the *mirror theory* to derive the security bound of the construction. However, as the construction requires two independent block cipher keys, reducing the number of block cipher keys to one was posed as an open problem.

## 1.2 Decrypted Wegman-Carter with Davies-Meyer

As an attempt to reduce the number of block cipher keys of EWCDM to one, Datta et al. [DDNY18] proposed a clever idea, where they replace the second block cipher call of EWCDM with the inverse of the first block cipher. This resulted in the construction called *Decrypted Wegman-Carter with Davies-Meyer* DWCDM, a nonce-based MAC, defined as follows:

$$\mathsf{DWCDM}_{\mathsf{E},\mathsf{H}}(\tilde{\nu}, M) = \mathsf{E}_k^{-1}\big(\mathsf{E}_k(\nu) \oplus \nu \oplus \mathsf{H}_{k_h}(M)\big),$$

where $\tilde{\nu} \in \{0,1\}^{2n/3}$ is the nonce, $M$ is the message and $\nu = \tilde{\nu}\|0^{n/3}$. Note that DWCDM uses a single block cipher key $k$ and another independent hash key $k_h$ for the AXU hash function. However, the main drawback of the construction is that DWCDM can only take $2n/3$ bit nonces.

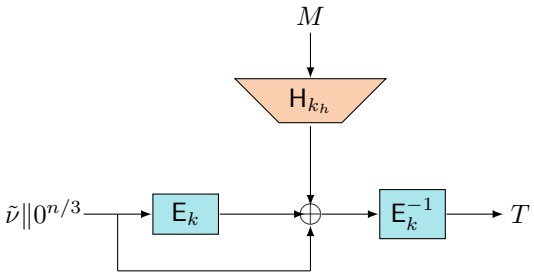


**Figure 1.2:** Decrypted Wegman-Carter with Davies-Meyer Construction.

In fact, authors have proved that DWCDM is not secured beyond the birthday limit with full $n$ bit nonces. They have shown that DWCDM is $2n/3$ bit secure against all nonce-respecting adversaries and $n/2$ bit secure against nonce-misuse adversaries. Moreover,

[4] For two variables, $P, Q$ and $\lambda \in \mathrm{GF}(2^n)$ we call an equation of the form $P \oplus Q = \lambda$, a bivariate affine equation.

the authors have also proposed a single-keyed nonce based MAC, dubbed 1K-DWCDM, where the hash key is derived using a block cipher evaluation on the input $0^{n-1}1$. This construction is secure up to $2n/3$ bits (resp. $n/2$ bits) in the nonce-respecting (resp. the nonce-misuse) setting. The nice property of DWCDM is that it allows one to view an evaluation of the construction as the xor of permutations in the middle of the string, i.e., $T = \mathsf{DWCDM}(\tilde{\nu}, M)$ can be equivalently viewed as

$$\mathsf{E}_k(\nu) \oplus \mathsf{E}_k(T) = \nu \oplus \mathsf{H}_{k_h}(M).$$

This feature allows the authors to use the mirror theory result for proving the security of their construction. However, to incorporate the verification attempts in the proof, they extended the mirror theory result by including univariate and bivariate affine non-equations along with bivariate affine equations. This result is known as the *Extended Mirror Theory* [DDNY18].

In the same paper [DDNY18], authors have mentioned that DWCDM can asymptotically achieves full $n$ bit security. They have given a sketchy proof that for a general $k$ with nonce space $\{0,1\}^{\frac{kn}{k+1}}$, DWCDM achieves $\frac{kn}{k+1}$ bit MAC security in the nonce respecting setting with the following condition and the conjecture

1. **Condition:** the underlying hash function must be $j$-way regular for all $3 \leq j \leq k$, i.e., for any $j$ distinct input points, the probability that sum of the the hash values evaluated at those points is non-zero, should be very low.

2. **Conjecture:** Proving $\frac{kn}{k+1}$ bits security for the extended mirror theory with $\xi_{\max} = k$.

Even though the above condition can be realized with a certain class of hash functions (e.g., Polyhash [MI11]), it is very difficult to prove the conjecture. In fact, in a follow-up work, Datta et al. [DDNY19] could only prove $2n/3$ bit MAC security of DWCDM with $n-1$ bit nonce space and left open for proving its security up to $3n/4$ bits. It is worth mentioning here that it is hard to improve the security of DWCDM beyond $2^{2n/3}$ with $2n/3$ bit nonce space. In general, improving the security of DWCDM beyond $2^{\frac{kn}{k+1}}$ with $\frac{kn}{k+1}$ bits of nonce space is a challenging task. In fact, we also do not know whether there exists an attack on DWCDM that uses $\frac{kn}{k+1}$ bit nonce with $2^{\frac{kn}{k+1}}$ MAC queries.

## 1.3 Mirror Theory and Its Relatable Debate

Mirror theory [Pat10] is an important combinatorial tool that provides a lower bound on the number of distinct solutions to a system of bivariate affine equations over any finite abelian group. Patarin stated this result as a conjecture in [Pat03] and proved in [Pat05]. This result was known as *Theorem* $P_i \oplus P_j$ *for* $\xi_{\max} = 2$ [Pat05], which was later renamed to *Mirror theory for* $\xi_{\max} = 2$ in [Pat10]. The result of *Mirror theory with* $\xi_{\max} = 2$ has been acknowledged in the community as a potential and a strong approach to establish the optimal security of XoP constructions [DHT17].

Besides the result of Mirror theory for $\xi_{\max} = 2$, Patarin [Pat05] also claimed that the number of distinct solutions to a system of $q$ bivariate affine equations with $\xi_{\max} > 2$ and with non-equality among the variables is always larger than the average number of solutions, provided $q \leq 2^n/67.(\xi_{\max} - 1)$. Patarin named this result the *Theorem* $P_i \oplus P_j$ *for any* $\xi_{\max}$. This result was also stated as a conjecture in [Pat03] (see Conjecture 8.1) in analyzing the security of the Feistel cipher. Only a couple of years later, this result was articulated in many follow-ups works for analyzing the security of the *xor of two permutations*, and it took a few articles [Pat05, Pat08b, Pat10, Pat13] for his result and security argument to evolve. Later, in 2017, this work culminated in a book [NPV17] called *Feistel Ciphers: Security Proofs and Cryptanalysis* by Nachef et al. However, the

proofs of this result in most of these works are very sketchy with plenty of giant equations and are missing most of the important details.

Theorem $P_i \oplus P_j$ for any $\xi_{\max}$ result plays a crucial role in deriving higher security bound of numerous cryptographic designs. Over the years, this general result has been applied in the context of deriving higher security bounds of numerous cryptographic constructions [DDNY18, DDNY19, DNT19, ML19, BDLN20, IMV16, MN17] that use XoP function as a component in their designs. The security proofs of most of these designs require a degeneration of the final outputs to get rid of the adaptive nature of the adversary. Hence the proof cannot use the fact that XoP function is a PRF. Instead, these security proofs require (by applying the H-Coefficient technique [Pat08a]) a good lower bound on the number of distinct solutions to a system of bivariate affine equations with a general $\xi_{\max}$, and therein comes the role of the result. As stated earlier, Mennink and Neves [MN17] used it to prove the optimal security bound of EWCDM. Iwata et al. [IMV16] also used this result to show the optimal security bound of CENC.

Despite the vivid applications of Theorem $P_i \oplus P_j$ for general $\xi_{\max}$, its proof is not very well understood in the community. The existing proofs of this result [Pat03, Pat05, Pat10] are very involved with lots of complicated equations. Moreover, the derivational process of these proofs has a lot of sloppiness in most of the crucial junctions. Hence, these proofs are practically not verifiable at all. Although the correctness of the proofs [Pat05, Pat10, NPV17] is debatable in the community, several authors have used this precarious result to derive an optimal bound for some constructions such as [IMV16, MN17, ZHY18]. Recently, Dutta et al. [DNS20] and Cogliati and Patarin [CP20] have independently developed a concrete and verifiable proof of Mirror theory for $\xi_{\max} = 2$. However, verifiable proof for Theorem $P_i \oplus P_j$ for any $\xi_{\max}$ result is still unavailable.

*Remark* 1. We would like to mention that applying the result of Theorem $P_i \oplus P_j$ for any $\xi_{\max}$ in deriving the optimal security of cryptographic constructions like EWCDM, CENC is technically correct. However, it may not be scientifically appropriate to apply a result whose correctness is still a matter of debate.

## 1.4 Our Contribution

In this paper, we prove that DWCDM with nonce space $\{0,1\}^{3n/4}$ is secure against all computationally bounded adversaries that make roughly $2^{3n/4}$ MAC queries and $2^n$ verification queries in nonce-respecting setting. We have also improved the MAC security bound of EWCDM from $2n/3$ bits to $3n/4$ bits in nonce-respecting setting. We would like to reiterate here that Mennink and Neves have already shown $n$ bit PRF security of EWCDM, leaving the proof of unforgeability open. However, as stated earlier that their analysis is solely based on the result of Theorem $P_i \oplus P_j$ for any $\xi_{\max}$, the correctness of the proof is a subject of debate. Inspired by the result of [KLL20, JN20], we have proved that the extended mirror theory for general $\xi_{\max}$ is secured roughly up to $3n/4$ bits. In particular, we have proved two versions of this result. In one version, the system of equations and non-equations of the extended mirror theory is based on the same permutation, whereas in the other version, the system of equations and non-equations is based on two independent random permutations. Our security proof of the constructions is based on the H-Coefficient technique [Pat08a]. Our first result of the extended mirror theory helps to bound the real interpolation probability for a good transcript of DWCDM, whereas the other one helps to bound the real interpolation for a good transcript of EWCDM. We would like to point out that the proof of EWCDM is similar to that of [CLLL20]. Moreover, the proof in establishing $3n/4$ bit security of EWCDM is less involved than proving $3n/4$ bit bound of nEHtM construction [CLLL20], as our construction deals with two independent random permutations whereas the latter one deals with a single random permutation. However, our non-trivial primary contribution in the paper is to establish $3n/4$ bit security of DWCDM.

As EWCDM is a close contender of DWCDM, and the proof of EWCDM was shown secure with less than $2^{2n/3}$ MAC queries by Cogliati and Seurin [CS16] (albeit the optimal PRF bound by Mennink and Neves [MN17]), we include the proof of the improved bound of EWCDM in the paper.

### 1.5 Proof Approach

Our MAC security proof of DWCDM and EWCDM fundamentally relies on Patarin's H-coefficient technique [Pat08a, Pat08b]. Similar to the technique of [CS16, DNT19], we cast the unforgeability game of MAC to an equivalent indistinguishability game, with a suitable choice of an ideal world, that allows us to apply the H-coefficient technique for bounding the distinguishing advantage of the construction of our concern.
One can express the evaluation of DWCDM (resp. EWCDM) as a sum of two identical permutations (resp. two independent permutations). Thus, $q$ many such evaluations of DWCDM gives us a system of $q$ many affine bi-variate equations as follows:

$$\mathsf{DWCDM} \Rightarrow \begin{cases} \mathsf{E}_k(\nu_1) \oplus \mathsf{E}_k(T_1) = \lambda_1 \\ \mathsf{E}_k(\nu_2) \oplus \mathsf{E}_k(T_2) = \lambda_2 \\ \qquad\qquad\qquad\quad \vdots \\ \mathsf{E}_k(\nu_q) \oplus \mathsf{E}_k(T_q) = \lambda_q \end{cases} \qquad \mathsf{EWCDM} \Rightarrow \begin{cases} \mathsf{E}_{k_1}(\nu_1) \oplus \mathsf{E}_{k_2}(T_1) = \gamma_1 \\ \mathsf{E}_{k_1}(\nu_2) \oplus \mathsf{E}_{k_2}(T_2) = \gamma_2 \\ \qquad\qquad\qquad\quad \vdots \\ \mathsf{E}_{k_1}(\nu_q) \oplus \mathsf{E}_{k_2}(T_q) = \gamma_q. \end{cases}$$

Here $\nu_i = \tilde{\nu}_i \| 0^{n/4}$, $\tilde{\nu}_i \in \{0,1\}^{3n/4}$ and $\lambda_i = \nu_i \oplus \mathsf{H}_{k_h}(M_i)$. Moreover, $\gamma_i = \nu_i \oplus \mathsf{H}_{k_h}(M_i)$, where $\nu_i \in \{0,1\}^n$. Along with this, we also need to ensure that the verification attempt of the adversary should fail (as a part of the good transcript), i.e., for a verification query $(\tilde{\nu}', M', T')$ (for DWCDM) and $(\nu', M', T')$ (for EWCDM), chosen by the adversary, we should always have

$$\begin{aligned} \mathsf{DWCDM} &\Rightarrow \mathsf{E}_k^{-1}(\mathsf{E}_k(\nu') \oplus \nu' \oplus \mathsf{H}_{k_h}(M')) \neq T', \\ \mathsf{EWCDM} &\Rightarrow \mathsf{E}_{k_2}(\mathsf{E}_{k_1}(\nu') \oplus \nu' \oplus \mathsf{H}_{k_h}(M')) \neq T'. \end{aligned}$$

Hence, it tells us that we also need to incorporate *bivariate affine non-equations* along with the system of bivariate affine equations. This leads us to *extend* the mirror theory technique incorporating the affine non-equations along with the affine bivariate equations. We use this extended mirror theory result while lower bounding the real interpolation probability for a good transcript.

## 2 Preliminaries

General Notations: For a set $\mathcal{X}$, we use the notation $\mathsf{X} \leftarrow\!\!\$\, \mathcal{X}$ to denote that $\mathsf{X}$ is sampled uniformly at random from $\mathcal{X}$ and independent of all random variables defined so far. We denote an empty set as $\emptyset$. For two mutually disjoint sets $\mathcal{X}$ and $\mathcal{Y}$, i.e., $\mathcal{X} \cap \mathcal{Y} = \emptyset$, we denote their union as $\mathcal{X} \sqcup \mathcal{Y}$, which we refer to as *disjoint union*. For a natural number $n$, $\{0,1\}^n$ denotes the set of all binary strings of length $n$ and $\{0,1\}^*$ denotes the set of all binary strings of arbitrary length. For a non-empty finite set $\mathcal{X} \subseteq \{0,1\}^n$ and an element $\lambda \in \{0,1\}^n$, we write $\mathcal{X} \oplus \lambda$ to denote the set $\{x \oplus \lambda : x \in \mathcal{X}\}$. For any binary string $x \in \{0,1\}^*$, $|x|$ denotes the length i.e. the number of bits in $x$. For $x, y \in \{0,1\}^n$, we write $z = x \oplus y$ to denote xor of $x$ and $y$. $\mathbf{0}$ denotes the element $0^n \in \{0,1\}^n$ and $\mathbf{1}$ denotes $0^{n-1}\|1 \in \{0,1\}^n$. 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 and for any natural number $q$, $[q]$ denotes the set $\{1, \dots, q\}$. We denote the set of all permutations over $\mathcal{X}$ as $\mathsf{Perm}(\mathcal{X})$. When $\mathcal{X} = \{0,1\}^n$, then we omit $\mathcal{X}$ and simply write $\mathsf{Perm}$ to denote the set of all permutations over $\{0,1\}^n$.

## 2.1 Security Definition of Block Cipher

A block cipher with key space $\mathcal{K}$ and domain $\{0,1\}^n$ is a mapping $\mathsf{E} : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ such that for all key $k \in \mathcal{K}$, $x \mapsto \mathsf{E}(k,x)$ is a permutation over $\{0,1\}^n$ and we denote $\mathsf{E}_k(x)$ for $\mathsf{E}(k,x)$. We consider a distinguisher $\mathbb{A}$ with oracle access to a permutation of $\{0,1\}^n$ that makes at most $q$ queries with running time at most $t$ and outputting a single bit after it finishes the interaction with the oracle. We define the pseudorandom permutation (prp)-advantage of $\mathbb{A}$ against the block cipher $\mathsf{E}$ as

$$\mathbf{Adv}^{\mathrm{prp}}_{\mathsf{E}}(\mathbb{A}) \stackrel{\Delta}{=} \left| \Pr[k \leftarrow\!\!\$\, \mathcal{K} : \mathbb{A}^{\mathsf{E}_k} \Rightarrow 1] - \Pr[\pi \leftarrow\!\!\$\, \mathsf{Perm} : \mathbb{A}^{\pi} \Rightarrow 1] \right|.$$

We say that $\mathsf{E}$ is $(q,t,\epsilon)$-secure prp if $\mathbf{Adv}^{\mathrm{prp}}_{\mathsf{E}}(q,t) \leq \epsilon$, where $\mathbf{Adv}^{\mathrm{prp}}_{\mathsf{E}}(q,t)$ is the maximum prp advantage in which the maximum is taken over all adversaries $\mathbb{A}$ that makes $q$ many queries with running time is at most $t$. Similar to the prp advantage, we say that $\mathbb{A}$ has strong pseudorandom permutation (sprp)-advantage against $\mathsf{E}$ if $\mathbb{A}$ is given an additional oracle access to the inverse of the permutation such that $\mathbb{A}$ makes at most $q^+$ queries (*forward*) to the permutation and $q^-$ queries (*backward*) to inverse permutation with running time at most $t$. We often merge the forward and backward queries and simply say $\mathbb{A}$ makes total $q$ queries including forward and backward queries.

## 2.2 Nonce Based MAC

Let $\mathsf{F} : \mathcal{K} \times \mathcal{N} \times \mathcal{M} \to \mathcal{T}$ be a keyed function where $\mathcal{K}, \mathcal{N}, \mathcal{M}$ and $\mathcal{T}$ are the key space, nonce space, message space and the tag space respectively. Based on $\mathsf{F}$, we define the nonce-based message authentication code $\mathcal{I} = (\mathcal{I}.\mathsf{KGen}, \mathcal{I}.\mathsf{TagGen}, \mathcal{I}.\mathsf{Ver})$ as follows: For $k \in \mathcal{K}$, the signing algorithm $\mathcal{I}.\mathsf{TagGen}_k$, takes as input $(\nu, M) \in \mathcal{N} \times \mathcal{M}$ and outputs $T \leftarrow \mathsf{F}(k,\nu,M)$ and the verification algorithm $\mathcal{I}.\mathsf{Ver}_k$, takes as input $(\nu, M, T) \in \mathcal{N} \times \mathcal{M} \times \mathcal{T}$ and outputs 1 if $\mathsf{F}_k(\nu, M) = T$; otherwise it outputs 0. Let $\mathbb{A}$ be a $(q_m, q_v, t)$-adversary against the unforgeability of $\mathcal{I}$ with oracle access of the signing algorithm $\mathcal{I}.\mathsf{TagGen}_k$ and the verification algorithm $\mathcal{I}.\mathsf{Ver}_k$ such that it makes $q_m$ signing and $q_v$ verification queries with running time at most $t$. $\mathbb{A}$ is said to be *nonce respecting* if she does not repeat a nonce in signing queries. However, $\mathbb{A}$ may repeat nonces in its verification queries. Moreover, the signing and the verification queries can be interleaved. $\mathbb{A}$ is said to *forge* $\mathcal{I}$ if for any of its verification query (not obtained through a previous signing query), the verification algorithm returns 1. The advantage of $\mathbb{A}$ against the unforgeability of the nonce based MAC $\mathcal{I}$ is defined as

$$\mathbf{Adv}^{\mathrm{nMAC}}_{\mathcal{I}}(\mathbb{A}) \stackrel{\Delta}{=} \Pr\left[\mathbb{A}^{\mathcal{I}.\mathsf{TagGen}_k, \mathcal{I}.\mathsf{Ver}_k} \text{ forges } \right],$$

where the randomness is defined over $k \leftarrow\!\!\$\, \mathcal{K}$ and the randomness of the adversary (if any). We write

$$\mathbf{Adv}^{\mathrm{nMAC}}_{\mathcal{I}}(q_m, q_v, t) \stackrel{\Delta}{=} \max_{\mathbb{A}} \mathbf{Adv}^{\mathrm{nMAC}}_{\mathcal{I}}(\mathbb{A}),$$

where the maximum is taken over all $(q_m, q_v, t)$-adversaries $\mathbb{A}$. In this paper, we skip the time parameter of the adversary as we will assume throughout the paper that the adversary is computationally unbounded. This will render us to assume that the adversary is deterministic. Moreover, $\mathbb{A}$ is non-trivial in the sense that it does not repeat any queries and does not make any queries whose output can be trivially computed.

Upper bound on $\mathbf{Adv}^{\mathrm{nMAC}}_{\mathcal{I}}(\mathbb{A})$. We obtain an upper bound for the nonce respecting MAC secuity of $\mathcal{I}$ in terms of the distinguishing advantage [DJN17], where the ideal world is comprised of a random oracle \$ that samples the tag $T$ independently and uniformly at random from $\{0,1\}^n$ for every nonce message pair $(\nu, M)$ and the reject oracle $\perp$

that always returns 0 for any $(\nu, M, T)$. Then, for any computationally unbounded and non-trivial nonce respecting adversary $\mathbb{A}$, $\mathbf{Adv}^{\mathrm{nMAC}}_{\mathcal{I}}(\mathbb{A})$ is upper bounded by

$$\max_{\mathbb{D}} \left| \Pr\left[\mathbb{D}^{\mathcal{I}.\mathsf{TagGen}_k, \mathcal{I}.\mathsf{Ver}_k} \Rightarrow 1\right] - \Pr\left[\mathbb{D}^{\$,\perp} \Rightarrow 1\right] \right|, \tag{1}$$

where $\mathbb{D}^{\mathbf{O}} \Rightarrow 1$ denotes that the distingisher $\mathbb{D}$ outputs 1 after interacting with its oracle $\mathbf{O}$.

## 2.3 H-Coefficient Technique for Nonce-Based MAC

Let $\mathcal{I} = (\mathcal{I}.\mathsf{KGen}, \mathcal{I}.\mathsf{TagGen}, \mathcal{I}.\mathsf{Ver})$ be a nonce-based MAC based on a keyed function $\mathsf{F} : \mathcal{K} \times \mathcal{N} \times \mathcal{M} \to \mathcal{T}$, where $\mathcal{K}, \mathcal{N}, \mathcal{M}$ and $\mathcal{T}$ are the key space, nonce space, message space and the tag space respectively. We fix a non-trivial and computationally unbounded distinguisher $\mathbb{D}$ that interacts with either of the two worlds: (1) in the real world it interacts with oracles $(\mathcal{I}.\mathsf{TagGen}_k, \mathcal{I}.\mathsf{Ver}_k)$ for a random key $k$ or (2) in the ideal world it interacts with oracles $(\$, \perp)$, making at most $q_m$ queries to its left (MAC) oracle and at most $q_v$ queries to its right (verification) oracle, and outputting a single bit. Let

$$\tau_m = \{(\nu_1, M_1, T_1), \ldots, (\nu_{q_m}, M_{q_m}, T_{q_m})\}$$

be the list of MAC queries and responses of $\mathbb{D}$ and

$$\tau_v = \{(\nu'_1, M'_1, T'_1, b'_1), (\nu'_2, M'_2, T'_2, b'_2), \ldots, (\nu'_{q_v}, M'_{q_v}, T'_{q_v}, b'_{q_v})\}$$

be the list of verification queries and responses of $\mathbb{D}$, where for all $j$, $b'_j \in \{0,1\}$ denotes the accept ($b'_j = 1$) or reject ($b'_j = 0$). We consider $\mathbb{D}$ to be stronger in the sense that it obtains some additional information after it made all its queries and obtains the corresponding responses but before it output its decision. If $\mathbb{D}$ interacts with the real world, then it obtains the key $k$ of the construction and if $\mathbb{D}$ interacts with the ideal world, then a dummy key $k$ is sampled uniformly at random from $\{0,1\}^n$ and released to the adversary. The triplet $\tau = (\tau_m, \tau_v, k)$ constitutes the query transcript of the attack. Let $\mathsf{X}_{\mathrm{re}}$ and $\mathsf{X}_{\mathrm{id}}$ denote the random variable of realizing a transcript $\tau$ in the real world and ideal world respectively. $\tau$ is said to be *attainable* (with respect to $\mathbb{D}$) if $\Pr[\mathsf{X}_{\mathrm{id}} = \tau] \neq 0$. $\Theta$ denotes the set of all attainable transcripts. Note that for an attainable transcript $\tau = (\tau_m, \tau_v, k)$, $b'_i = 0$, for every $i \in [q_v]$. Now, we state the main result of the H-coefficient technique (see e.g. [CS14] for the proof) as follows:

**Lemma 1.** *Let $\mathbb{D}$ be a fixed deterministic distinguisher and $\Theta = \Theta_{\mathrm{g}} \sqcup \Theta_{\mathrm{b}}$ be some partition of the set of all attainable transcripts. Suppose there exists $\epsilon_{\mathrm{ratio}} \geq 0$ such that for any $\tau \in \Theta_{\mathrm{g}}$,*

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

*and there exists $\epsilon_{\mathrm{bad}} \geq 0$ such that $\Pr[\mathsf{X}_{\mathrm{id}} \in \Theta_{\mathrm{b}}] \leq \epsilon_{\mathrm{bad}}$. Then, $\mathbf{Adv}(\mathbb{D}) \leq \epsilon_{\mathrm{ratio}} + \epsilon_{\mathrm{bad}}$.*

## 2.4 Universality and Regularity of Keyed Hash Functions

Let $\mathcal{K}_h$ and $\mathcal{X}$ be two non-empty finite sets and $\mathsf{H}$ be a keyed function $\mathsf{H} : \mathcal{K}_h \times \mathcal{X} \to \{0,1\}^n$. Then,

(i) ALMOST-XOR-UNIVERSALITY: $\mathsf{H}$ is said to be an $\epsilon_{\mathrm{axu}}$-almost xor universal (AXU) hash function, if for any distinct $x, x' \in \mathcal{X}$ and for any $\Delta \in \{0,1\}^n$,

$$\Pr\left[k_h \leftarrow\!\!\$\, \mathcal{K}_h : \mathsf{H}_{k_h}(x) \oplus \mathsf{H}_{k_h}(x') = \Delta\right] \leq \epsilon_{\mathrm{axu}}.$$

(ii) ALMOST REGULARITY: We say that $\mathsf{H}$ is an $\epsilon_{\text{reg}}$-almost regular (AR) hash function, if for any $x \in \mathcal{X}$ and for any $\Delta \in \{0,1\}^n$,

$$\Pr\left[k_h \xleftarrow{\$} \mathcal{K}_h : \mathsf{H}_{k_h}(x) = \Delta\right] \leq \epsilon_{\text{reg}}.$$

(iii) $r$-WAY REGULAR: We say that $\mathsf{H}$ is said to be an $\epsilon_{r\text{-reg}}$ $r$-way regular hash function if for any distinct $x_1, x_2, \ldots, x_r \in \mathcal{X}$ and for any non-zero $\Delta \in \{0,1\}^n$,

$$\Pr[k_h \xleftarrow{\$} \mathcal{K}_h : \mathsf{H}_{k_h}(x_1) \oplus \mathsf{H}_{k_h}(x_2) \oplus \ldots \oplus \mathsf{H}_{k_h}(x_r) = \Delta] \leq \epsilon_{r\text{-reg}}. \tag{2}$$

# 3 Extended Mirror Theory

We prove the MAC security of EWCDM and DWCDM using the H-Coefficient technique, where one is required to lower bound the probability of realizing a good transcript in the real and the ideal world. In order to compute this probability in the real world, we need to count the number of permutations such that the following system of bivariate affine equations and non-equations

$$(\mathcal{E}_m) = \begin{cases} \pi_1(\nu_1) \oplus \pi_2(T_1) = \lambda_1 \\ \pi_1(\nu_2) \oplus \pi_2(T_2) = \lambda_2 \\ \vdots \\ \pi_1(\nu_{q_m}) \oplus \pi_2(T_{q_m}) = \lambda_{q_m} \end{cases} \qquad (\mathcal{E}_v) = \begin{cases} \pi_1(\nu'_1) \oplus \pi_2(T'_1) \neq \lambda'_1 \\ \pi_1(\nu'_2) \oplus \pi_2(T'_2) \neq \lambda'_2 \\ \vdots \\ \pi_1(\nu'_{q_v}) \oplus \pi_2(T'_{q_v}) \neq \lambda'_{q_v} \end{cases}$$

hold. Note that $\pi_1$ and $\pi_2$ are two independent $n$-bit permutations for EWCDM, whereas $\pi_1 = \pi_2 = \pi$ for DWCDM. Moreover, $\lambda_i = \nu_i \oplus \mathsf{H}_{k_h}(M_i)$, where $\nu_i \in \{0,1\}^n$ for EWCDM, whereas the last $n/4$ bits of $\nu_i$ are set to zero for DWCDM. Therefore, it boils down to counting the number of solutions to the above system of bivariate affine equations and non-equations. This result is captured by the result of *Extended Mirror Theory* [DNT19].

Consider an undirected edge-labelled acylic (possibly bipartite) graph $\mathsf{G} = (\mathcal{V}, \mathcal{E} \sqcup \mathcal{E}', \mathcal{L})$ with edge labelling function $\mathcal{L} : \mathcal{E} \sqcup \mathcal{E}' \to \{0,1\}^n$, where $\mathcal{V} = \{Y_1, \ldots, Y_s\}$ be the set of vertices of the graph and the edge set is partitioned into two disjoint sets $\mathcal{E}$ and $\mathcal{E}'$. We call the edges of $\mathcal{E}$ as *equation edge* and the edges of $\mathcal{E}'$ as *non-equation edge*. For an equation edge $\{Y_i, Y_j\} \in \mathcal{E}$, we write $\mathcal{L}(\{Y_i, Y_j\}) = \lambda_{ij}$ (and so $\lambda_{ij} = \lambda_{ji}$) and $\mathcal{L}(\{Y_i, Y_j\}) = \lambda'_{ij}$ for all non-equation edges $\{Y_i, Y_j\} \in \mathcal{E}'$. For a bipartite graph $\mathsf{G}$, $\mathcal{V}$ is the disjoint union of two sets $\mathcal{V}_1 = \{Y_1, \ldots, Y_{s_\ell}\}$ and $\mathcal{V}_2 = \{Z_1, \ldots, Z_{s_\mathtt{r}}\}$ such that $s = s_\ell + s_\mathtt{r}$ be the total number of vertices in the graph. We write an edge of $\mathcal{E}$ as $\{Y_i, Z_j\}$, and we denote its label as $\mathcal{L}(\{Y_i, Z_j\}) = \lambda_{ij}$ (and so $\lambda_{ij} = \lambda_{ji}$). Moreover, we denote the label of $\{Y_i, Z_j\} \in \mathcal{E}'$ as $\mathcal{L}(\{Y_i, Z_j\}) = \lambda'_{ij}$.

Let $\mathsf{G}^{=} \triangleq (\mathcal{V}^{=}, \mathcal{E}, \mathcal{L}_{|\mathcal{E}})$ denote the subgraph of $\mathsf{G}$, where $\mathcal{V}^{=}$ is the set of vertices of $\mathcal{V}$ such that they are incident on at least one edge of $\mathcal{E}$ and $\mathcal{L}_{|\mathcal{E}}$ is the function $\mathcal{L}$ restricted over the set $\mathcal{E}$. For a path $\mathcal{P}$ in the graph $\mathsf{G}^{=}$, we define the label of the path as $\mathcal{L}(\mathcal{P}) \triangleq \sum_{e \in \mathcal{P}} \mathcal{L}(e)$. Similarly, for a cycle $\mathcal{C}$ in the graph $\mathsf{G}$, we define the label of the cycle as $\mathcal{L}(\mathcal{C}) \triangleq \sum_{e \in \mathcal{C}} \mathcal{L}(e)$. We say the graph $\mathsf{G}$ is **good** if it satisfies the following two conditions:

1. $\mathcal{L}(\mathcal{P}) \neq \mathbf{0}$, for all paths $\mathcal{P}$ in the graph $\mathsf{G}^{=}$, and
2. $\mathcal{L}(\mathcal{C}) \neq \mathbf{0}$, for all cycles $\mathcal{C}$ containing exactly one non-equation edge $e' \in \mathcal{E}'$ (i.e., all the remaining edges of $\mathcal{C}$ are elements of $\mathcal{E}$).

For a bipartite graph $\mathsf{G}$, we say that $\mathsf{G}$ is **good**, if it satisfies the following two conditions:

1. $\mathcal{L}(\mathcal{P}) \neq \mathbf{0}$, for all paths $\mathcal{P}$ of even length in the graph $\mathsf{G}^{=}$ and

2. $\mathcal{L}(\mathcal{C}) \neq \mathbf{0}$, for all cycles $\mathcal{C}$ of even length containing exactly one non-equation edge $e' \in \mathcal{E}'$ (i.e., all other edges of $\mathcal{C}$ are elements of $\mathcal{E}$).

WHY GOOD GRAPH IS CALLED GOOD? Let $\mathcal{P}$ be any path in $\mathsf{G}^{=}$ , and $Y_s$, $Y_t$ be the starting and the end vertex of the path respectively. Note that if $\mathcal{L}(\mathcal{P})$ is zero, then that implies $Y_s \oplus Y_t = 0$, which is nothing but the permutation collision. Regarding condition (2), let $\mathcal{C}$ be any cycle of $\mathsf{G}$ such that it contains exactly one non-equation edge $e'$ and let $x$ be the label of the path $\mathcal{P} = \mathcal{C}/e'$. Then, it implies that $Y_s \oplus Y_t = x$, where $Y_s$ is the starting and $Y_t$ is the ending vertex of $\mathcal{P}$ respectively. Note that, the label of the edge $\{Y_s, Y_t\}$ is $\mathcal{L}(e')$. Therefore,

$$\mathcal{L}(\mathcal{C}) = 0 \Rightarrow \mathcal{L}(\mathcal{P}) \oplus \mathcal{L}(e') = 0 \Rightarrow x = \mathcal{L}(e') \Rightarrow Y_s \oplus Y_t = \mathcal{L}(e'),$$

which contradicts the non-equation $Y_s \oplus Y_t \neq \mathcal{L}(e')$. This is why we exclude such graphs from the set of good graphs. Similarly, for a bipartite graph $\mathsf{G}$, we assume $\mathcal{P}$ to be a path of even length in $\mathsf{G}^{=}$, and $Y_s$, $Y_t$ are the starting and end vertex of the path respectively. Note that as the path length is even, starting and ending vertex is $Y_s$ and $Y_t$ respectively. In fact, the starting and the ending vertex could have been $Z_s$ and $Z_t$. However, if the path length is odd, then the stating and ending vertex would have been $Y_s$ and $Z_t$ respectively. Note that, for such a even length path $\mathcal{P}$, if $\mathcal{L}(\mathcal{P})$ is zero, then that implies $Y_s \oplus Y_t = 0$ or $Z_s \oplus Z_t = 0$ (if the starting and ending vertex would have been $Z_s$ and $Z_t$ respectively), which is nothing but the permutation collision. Regarding condition (2), let $\mathcal{C}$ be any cycle of $\mathsf{G}$ of even length such that it contains exactly one non-equation edge $e'$ and let $x$ be the label of the path $\mathcal{P} = \mathcal{C}/e'$. Then, it implies that $Y_s \oplus Z_t = x$, where we assume that $Y_s$ is the starting and $Z_t$ is the ending vertex of $\mathcal{P}$ respectively. Note that, the label of the edge $\{Y_s, Z_t\}$ is $\mathcal{L}(e')$, and hence,

$$\mathcal{L}(\mathcal{C}) = 0 \Rightarrow \mathcal{L}(\mathcal{P}) \oplus \mathcal{L}(e') = 0 \Rightarrow x = \mathcal{L}(e') \Rightarrow Y_s \oplus Z_t = \mathcal{L}(e').$$

This contradicts the non-equation $Y_s \oplus Z_t \neq \mathcal{L}(e')$, and hence we exclude such graphs from the set of good graphs. For such a good graph $\mathsf{G}$, we associate a system of bivariate affine equations and non-equations for the general graph and for the bipartite graph as follows:

$$\mathcal{E}_{\mathsf{G}}^{\mathtt{gen}} = \begin{cases} Y_i \oplus Y_j & = \lambda_{ij} \; \forall \; \{Y_i, Y_j\} \in \mathcal{E}, \\ Y_i \oplus Y_j & \neq \lambda'_{ij} \; \forall \; \{Y_i, Y_j\} \in \mathcal{E}', \end{cases} \quad \mathcal{E}_{\mathsf{G}}^{\mathtt{bi}} = \begin{cases} Y_i \oplus Z_j & = \lambda_{ij} \; \forall \; \{Y_i, Z_j\} \in \mathcal{E}, \\ Y_i \oplus Z_j & \neq \lambda'_{ij} \; \forall \; \{Y_i, Z_j\} \in \mathcal{E}'. \end{cases}$$

Note that, in the above system of bivariate affine equations and non-equations, the variables are the vertices of the associated graph. We say that two variables are involved in an equation, if the corresponding vertices are connected by an equation edge in the graph. Similarly, we say that two variables are involved in a non-equation, if the corresponding vertices are connected by a non-equation edge in the graph. The constants of the equation or non-equation are the label of the corresponding edges. Therefore, for $\mathcal{E}_{\mathsf{G}}^{\mathtt{gen}}$, the variables are $Y_i$'s and for $\mathcal{E}_{\mathsf{G}}^{\mathtt{bi}}$, the variables are $Y_i$'s and $Z_i$'s. For a subgraph $\mathsf{G}^{=} = (\mathcal{V}^{=}, \mathcal{E}, \mathfrak{L})$ of a good graph $\mathsf{G}$, two vertices in $\mathcal{V}^{=}$ are said to be related to each other if and only if they are connected by an edge in $\mathcal{E}$. This induces partitioning on $\mathcal{V}^{=}$ and each partition is called a component. The size of a component refers to the number of elements (i.e., the number of vertices) in the partition. The set of components in $\mathsf{G}^{=}$ is denoted by $\mathsf{comp}(\mathsf{G}^{=}) = (\mathsf{C}_1 \sqcup \ldots \sqcup \mathsf{C}_\alpha \sqcup \mathsf{D}_1 \sqcup \ldots \sqcup \mathsf{D}_\beta)$ where we assume that there are $\alpha$ many components of $\mathsf{G}^{=}$ (i.e., $\mathsf{C}_1, \ldots, \mathsf{C}_\alpha$) with component size greater than 2 and $\beta$ many components of $\mathsf{G}^{=}$ (i.e., $\mathsf{D}_1, \ldots, \mathsf{D}_\beta$) having component size exactly 2. We write $\mathsf{C}$ to denote $\mathsf{C}_1 \sqcup \ldots \sqcup \mathsf{C}_\alpha$ and $\mathsf{D}$ to denote $\mathsf{D}_1 \sqcup \ldots \sqcup \mathsf{D}_\beta$.

**Definition 1.** Let $\mathcal{E}_{\mathsf{G}}$ be a system of equations and non-equations corresponding to a good acyclic edge-labelled graph $\mathsf{G}$ (as defined above). An injective function $\Phi : \mathcal{V} \to \{0,1\}^n$, is said to be an *injective solution* to $\mathcal{E}_{\mathsf{G}}$ if $\Phi(Y_i) \oplus \Phi(Y_j) = \lambda_{ij}$ for all $\{Y_i, Y_j\} \in \mathcal{E}$ and $\Phi(Y_i) \oplus \Phi(Y_j) \neq \lambda'_{ij}$ for all $\{Y_i, Y_j\} \in \mathcal{E}'$. For a good acyclic edge-labelled bipartite graph $\mathsf{G}$, an injective function $\Phi : \mathcal{V}_1 \sqcup \mathcal{V}_2 \to \{0,1\}^n$, is said to be an *injective solution* to $\mathcal{E}_{\mathsf{G}}$ if $\Phi(Y_i) \oplus \Phi(Z_j) = \lambda_{ij}$ for all $\{Y_i, Z_j\} \in \mathcal{E}$ and $\Phi(Y_i) \oplus \Phi(Z_j) \neq \lambda'_{ij}$ for all $\{Y_i, Z_j\} \in \mathcal{E}'$.

In the following, we state and prove the following result of mirror theory which says that if $\mathsf{G}$ is a good acyclic edge-labelled (bipartite) graph such that its subgraph $\mathsf{G}^{=}$ can be decomposed into finitely many components of size greater than 2 and exactly 2, then the number of injective solutions to $\mathcal{E}_{\mathsf{G}}$ is very close to the average number of solutions until the number of edges in $\mathcal{E}$ is roughly $2^{3n/4}$.

**Disclaimer:** Although the way we define a component is a set of vertices, from now onwards, we also equivalently view a component as a graph with appropriate edges. Thus, $\mathsf{C} = \mathsf{C}_1 \sqcup \ldots \sqcup \mathsf{C}_\alpha$ alternatively denotes a disjoint collection of subgraphs of $\mathsf{G}^{=}$.

Now, we state the main theorem of *Extended Mirror Theory* that in principle estimates a lower bound on the number of solutions to the induced system of equations and non-equations for a good graph $\mathsf{G}$. We state two versions of the theorem, one is for a good acyclic general graph, and another is for a good acylic bipartite graph.

**Theorem 1 (General Graph).** *Let $\mathsf{G} = (\mathcal{V}, \mathcal{E} \sqcup \mathcal{E}', \mathcal{L})$ be a good graph with $s$ many vertices such that $|\mathcal{E}| = q_m, |\mathcal{E}'| = q_v$. Let $q_c$ denote the total number of edges in $\mathsf{C}$. Then the total number of injective solutions to $\mathcal{E}_{\mathsf{G}}$ which are chosen from $\{0,1\}^n$, is at least:*

$$\frac{(2^n)_s}{2^{nq_m}}\left(1 - \frac{9q_c^2}{4 \cdot 2^n} - \frac{9q_c^2 q_m + 24 q_c q_m^2 + 6 q_c q_m + 40 q_m^2}{2^{2n}} - \frac{16 q_m^4}{2^{3n}} - \frac{7q_v}{2^n}\right).$$

**Theorem 2 (Bipartite Graph).** *Let $\mathsf{G} = (\mathcal{V}_1 \sqcup \mathcal{V}_2, \mathcal{E} \sqcup \mathcal{E}', \mathcal{L})$ be a good bipartite graph with $s_\ell$ many vertices in $\mathcal{V}_1$ and $s_{\mathtt{r}}$ many vertices in $\mathcal{V}_2$, such that $|\mathcal{E}| = q_m, |\mathcal{E}'| = q_v$ and $s = s_\ell + s_{\mathtt{r}}$, the total number of vertices of the graph $\mathsf{G}$. Let $q_c$ denote the total number of edges in $\mathsf{C}$. Then the total number of injective solutions to $\mathcal{E}_{\mathsf{G}}$ which are chosen from $\{0,1\}^n$, is at least:*

$$\frac{(2^n)_{s_\ell}(2^n)_{s_{\mathtt{r}}}}{2^{nq_m}}\left(1 - \frac{9q_c^2}{4 \cdot 2^n} - \frac{9q_m q_c^2}{4 \cdot 2^{2n}} - \frac{3q_c q_m^2}{2 \cdot 2^{2n}} - \frac{q_m^2}{2^{2n}} - \frac{8q_m^4}{3 \cdot 2^{3n}} - \frac{5q_v}{2^n}\right).$$

**Notations:** Before we prove the above two theorems, we set up a few notations. Let $h(\mathsf{G})$ denote the number of solutions to the graph $\mathsf{G}$. Let $h_c(i)$ denote the number of solutions for the subgraph $\mathsf{C}_1 \sqcup \ldots \sqcup \mathsf{C}_i$ and $h_d(i)$ denotes the number of solutions for the subgraph $\mathsf{C} \sqcup \mathsf{D}^i$ where $\mathsf{D}^i \triangleq \mathsf{D}_1 \sqcup \mathsf{D}_2 \sqcup \ldots \sqcup \mathsf{D}_i$. Therefore, $h_d(0) = h_c(\alpha)$ and $h_d(\beta) = h(\mathsf{G}^{=})$. For the graph $\mathsf{G}$, a *blue dashed edge* represents a non-equation edge and hence belongs to the set $\mathcal{E}'$ and a *red continuous edge* represents an equation edge and hence belongs to the set $\mathcal{E}$. Moreover, $\mathcal{V}^{=}$ denotes the set of all vertices of the subgraph $\mathsf{G}^{=}$. We assume that there are $\widetilde{\mu}_{i,j}$ edges from $\mathcal{E}'$ connecting vertices of the $i$-th and $j$-th components of $\mathsf{G}^{=}$ where $j < i$. Moreover, let $|\mathcal{V} \setminus \mathcal{V}^{=}| = k'$ and for any vertex $v_i \in \mathcal{V} \setminus \mathcal{V}^{=}$, there are $\mu'_i$ many blue dashed edges incident on $v_i$.

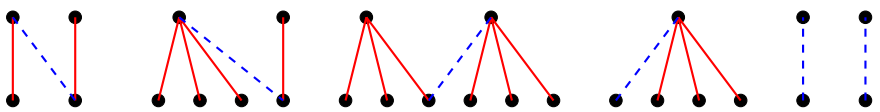

**Figure 3.1:** Blue dashed edges denote the verification non-equations and continious red edges denote MAC equations.

## 3.1 Proof of Theorem 1

We prove the result in a step by step manner. We first estimate a lower bound on $h_c(\alpha)$ and then we estimate a lower bound on $h_d(\beta)$, and finally, we estimate a lower bound on the number of solutions to $\mathsf{G} \setminus \mathsf{G}^{=}$. Let $\mathcal{V}_{\mathsf{C}}^{=}$ denote the set of vertices of $\mathsf{C}$ and $w_i \stackrel{\Delta}{=} |\mathsf{C}_i|$. For $1 \leq i \leq \alpha$, we write $\sigma_i = w_1 + \ldots + w_i$, with the convention that $\sigma_0 = 0$. Note that $q_c = \sigma_\alpha - \alpha$ as each component $\mathsf{C}_i$ is a tree.

### 3.1.1 Lower Bound on $h_c(\alpha)$.

To lower bound $h_c(\alpha)$, we count the number of solutions in each of the $\alpha$ components of $\mathsf{C}$. For the first component $\mathsf{C}_1$, there are $2^n$ ways to assign values to any one of the vertices of the component, and that uniquely determines the values to the rest of the variables in that component. For example, consider the graph as depicted in Fig. 3.2 and let us assume that we assign a value to vertex $v$ and let the assigned value ve $x$. Then the value at node $v_1$ is $x \oplus \lambda_1$, at node $v_2$ is $x \oplus \lambda_2$, at node $v_3$ is $x \oplus \lambda_3$ and at node $v_4$ is $x \oplus \lambda_4$. As the graph is good, none of the $\lambda$ values is zero, and all the $\lambda$ values are distinct. These two facts ensure the distinctness of the values assigned at node $v_1, v_2, v_3$ and $v_4$ by assigning the value to node $v$ which has $2^n$ choices.

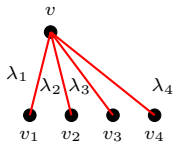


**Figure 3.2:** Component of a good graph with the label on the edges.

Once such a solution is fixed for the first component, we consider the second component. We consider any arbitrary vertex in the second component $\mathsf{C}_2$ of $\mathsf{G}^{=}$. Let $Y_{i_{w_1+1}} \in \mathcal{V}^{=}$ be a variable in $\mathsf{C}_2$. A valid solution for $Y_{i_{w_1+1}}$ should not take $w_1 w_2$ values. This is due to the fact that $Y_{i_{w_1+1}}$ cannot take $w_1$ values. Moreover, once an assignment is done to $Y_{i_{w_1+1}}$, it fixes the value of the rest of $w_2 - 1$ vertices of $\mathsf{C}_2$ such that each of the remaining vertices of $\mathsf{C}_2$ do not collide with the previous $w_1$ values. Therefore, a total of $w_1 + (w_2 - 1)w_1 = w_1 w_2$ values are discarded. Additionally, as there are $\widetilde{\mu}_{2,1}$ many blue dashed edges connecting the component $\mathsf{C}_1$ and $\mathsf{C}_2$, there are $\widetilde{\mu}_{2,1}$ many paths from the vertex $Y_{i_{w_1+1}}$ to the vertices of the component $\mathsf{C}_1$, and hence it cannot take $\widetilde{\mu}_{2,1}$ values that violate the non-equality conditions of $\widetilde{\mu}_{2,1}$ many blue dashed edges. As a result, there are at most $w_1 w_2 + \widetilde{\mu}_{2,1}$ forbidden values for assignment to the vertex $Y_{i_{w_1+1}}$. Hence, there are at least $(2^n - w_1 w_2 - \widetilde{\mu}_{2,1})$ valid choices for $Y_{i_{w_1+1}}$. Once a valid value is assigned to the variable $Y_{i_{w_1+1}}$, the remaining variables in the second component will be assigned uniquely.

In general, for the $i$-th component, once the injective solution is fixed for the previous $i-1$ components, there are at least $(2^n - \sigma_{i-1} w_i - \widetilde{\mu}_{i,1} - \ldots - \widetilde{\mu}_{i,i-1})$ ways for an injective solution for the $i$-th component. For the notational simplicity, we write $\delta_i = (\widetilde{\mu}_{i,1} + \ldots + \widetilde{\mu}_{i,i-1})$. Hence, we have

$$
\begin{aligned}
h_c(\alpha) &\geq \prod_{i=1}^{\alpha} \left(2^n - \sigma_{i-1} w_i - \delta_i\right) = 2^{n\alpha} \prod_{i=1}^{\alpha} \left(1 - \frac{\sigma_{i-1} w_i}{2^n} - \frac{\delta_i}{2^n}\right) \\
&\geq 2^{n\alpha} \left(1 - \sum_{i=1}^{\alpha} \frac{\sigma_{i-1} w_i}{2^n} - \sum_{i=1}^{\alpha} \frac{\delta_i}{2^n}\right) \stackrel{(1)}{\geq} 2^{n\alpha} \left(1 - \frac{1}{2^n} (\sum_{i=1}^{\alpha} w_i)^2 - \frac{q'_v}{2^n}\right) \\
&\stackrel{(2)}{\geq} 2^{n\alpha} \left(1 - \frac{9 q_c^2}{4 \cdot 2^n} - \frac{q'_v}{2^n}\right), \qquad (3)
\end{aligned}
$$

where (1) holds as $\delta_1 + \delta_2 + \ldots + \delta_\alpha = q'_v$, the total number of blue dashed edges across the components of $\mathsf{G}^{=}$ and (2) holds as $(w_1 + \ldots + w_\alpha) = \sigma_\alpha = q_c + \alpha$ and $\alpha \leq q_c/2$.

### 3.1.2 Lower Bound on $h_d(\beta)$.

Now we would like to find a lower bound on $h_d(i+1)$ in terms of $h_d(i)$. Let us denote the label of the edges in component $\mathsf{D}_i$ is $\lambda_i^*$ and recall that $\sigma_\alpha$ denotes the total number of vertices in $\mathsf{C}$. Now, we consider the component $\mathsf{D}_{i+1}$. For $(\alpha+i+1)$-th component $\mathsf{D}_{i+1}$, $\widetilde{\mu}_{\alpha+i+1,j}$ denotes the number of blue dashed edges connecting component $\mathsf{D}_{i+1}$ and the $j$-th component, where $j \in [\alpha+i]$. Note that, the one end vertex of each such edge is a vertex from component $\mathsf{D}_{i+1}$ and the other end vertex resides in the $j$-th component, where $j \in [\alpha+i]$. We represent the end vertex of each such edge which resides in the $j$-th component, $j \in [\alpha+i]$ as a single-ton set $A_k^j$, where $k \in [\widetilde{\mu}_{\alpha+i+1,j}]$. Now, we are interested in obtaining a lower bound on $h_d(i+1)$ as follows. Let $Y_{\sigma_\alpha+2i+1}$ be the vertex of $\mathsf{D}_{i+1}$. Then, $Y_{\sigma_\alpha+2i+1}$ must satisfy the following:

1. $Y_{\sigma_\alpha+2i+1} \notin \mathcal{V}_{\mathsf{C}}^{=} \sqcup \{Y_{\sigma_\alpha+1}, \ldots, Y_{\sigma_\alpha+2i}\} (\triangleq \mathcal{Z}_1)$
2. $Y_{\sigma_\alpha+2i+1} \notin \left(\mathcal{V}_{\mathsf{C}}^{=} \oplus \lambda_{i+1}^*\right) \sqcup \left(\{Y_{\sigma_\alpha+1}, \ldots, Y_{\sigma_\alpha+2i}\} \oplus \lambda_{i+1}^*\right) (\triangleq \mathcal{Z}_2)$
3. $Y_{\sigma_\alpha+2i+1} \notin A_j^k$ where $j \in [\alpha+i]$ and $k \in [\widetilde{\mu}_{\alpha+i+1,j}]$.

Since $|\mathcal{V}_{\mathsf{C}}^{=}| = \sigma_\alpha$, $|\mathcal{Z}_1| = |\mathcal{Z}_2| = \sigma_\alpha + 2i$. Applying the inclusion-exclusion principle, we have

$$
\begin{aligned}
h_d(i+1) &= \sum_{\text{soln to } \mathsf{C} \cup \mathsf{D}^i} \left(2^n - \Big|\mathcal{Z}_1 \bigcup \mathcal{Z}_2 \bigcup \Big(\bigcup_{j=1}^{\alpha+i} \bigcup_{k=1}^{\widetilde{\mu}_{\alpha+i+1,j}} A_j^k\Big)\Big|\right) \\
&\geq \left(2^n - 2\sigma_\alpha - 4i - \sum_{j=1}^{\alpha+i} \widetilde{\mu}_{\alpha+i+1,j}\right) h_d(i) + \sum_{\text{soln to } \mathsf{C} \cup \mathsf{D}^i} |\mathcal{Z}_1 \cap \mathcal{Z}_2| \\
&\geq \left(2^n - 2\sigma_\alpha - 4i - \sum_{j=1}^{\alpha+i} \widetilde{\mu}_{\alpha+i+1,j}\right) h_d(i) + \sum_{\substack{P \in \{Y_{\sigma_\alpha+1}, \ldots, Y_{\sigma_\alpha+2i}\} \\ Q \in \{Y_{\sigma_\alpha+1}, \ldots, Y_{\sigma_\alpha+2i}\}}} h'(P,Q), \quad (4)
\end{aligned}
$$

where $h'(P,Q)$ denotes the number of solutions to $\mathsf{C} \cup \widetilde{\mathsf{D}}^i$, where $\widetilde{\mathsf{D}}^i = \mathsf{D}^i \cup P \overset{\lambda_{i+1}^*}{\rightarrow} Q$. We say two indices $u, v \in \{\sigma_\alpha + 1, \ldots, \sigma_\alpha + 2i\}$ are in the same component of $\mathsf{D}^i$, if there is an edge between vertices $Y_u$ and $Y_v$ in the subgraph $\mathsf{D}^i$. Now, there are the following two cases:

- if $P = Y_u$ and $Q = Y_v$ such that $u$ and $v$ are in the same component of $\mathsf{D}^i$ and $\lambda_{i+1}^* = \lambda$, where $\lambda$ is the label of the edge connecting vertices $Y_u$ and $Y_v$, then $h'(P,Q) = h_d(i)$. Moreover, if $\lambda_{i+1}^* \neq \lambda$, then $h'(P,Q) = 0$.

- On the other hand, we consider the case when $u$ and $v$ are in different components and $\lambda_{i+1}^* \neq \lambda_a^*, \lambda_{i+1}^* \neq \lambda_b^*, \lambda_{i+1}^* \neq \lambda_a^* \oplus \lambda_b^*$, where $\lambda_a^*$ (resp $\lambda_b^*$) is the label of the edge whose end vertex is $Y_u$ (resp $Y_v$). In this case, we estimate a lower bound on $h'(P,Q)$. Note that when $u$ and $v$ are in different components and if any of the above conditions hold, then $h'(P,Q) = 0$.

The following lemma gives a lower bound on $h'(P,Q)$ in terms of $h_d(i)$, proof of which is postponed in Sect. 3.2.

**Lemma 2.** $h'(P,Q) \geq \frac{h_d(i)}{2^n}\left(1 - \frac{2(\sigma_\alpha+2i)}{2^n-2(\sigma_\alpha+2i)}\right)$.

Now, we define the following two sets:

- $\mathcal{S}_1 \stackrel{\Delta}{=} \{k \in [i] : \lambda^*_k = \lambda^*_{i+1}\}$ and let $\delta \stackrel{\Delta}{=} |\mathcal{S}_1|$
- $\mathcal{S}_2 \stackrel{\Delta}{=} \{(k,k') \in [i] \times [i] : k \neq k', \lambda^*_k \neq \lambda^*_{i+1}, \lambda^*_{k'} \neq \lambda^*_{i+1}, \lambda^*_{i+1} \neq \lambda^*_k \oplus \lambda^*_{k'}\}$.

Note that, $\delta$ is the number of multi-collisions of $\lambda^*_{i+1}$ and let $\Delta$ denote the maximum number of multi-collisions maximized over $\lambda^*$ values. Now, it is easy to see that

$$|\mathcal{S}_2| \geq 4i^2 - 8i - 4i\Delta - 8i\delta. \tag{5}$$

This is because $k$ and $k'$ are not in the same component, and hence we have $2i(2i-2)$ choices for $(k,k')$. However, out of these many choices, there are $2\delta(2i-2\delta)+(2i-2\delta)(2\delta)+4\delta(\delta-1)$ possibilities for $\lambda^*_{i+1} = \lambda^*_k$ or $\lambda^*_{k'}$. Moreover, out of these many choices for $(k,k')$, there are at most $2i(2\Delta-2)$ choices for $(k,k')$ such that $\lambda^*_{i+1} = \lambda^*_k \oplus \lambda^*_{k'}$. Therefore, from Eqn. (4), Eqn. (5), Lemma 2 and the above two cases where $(u,v)$ either belongs to the same component or in a different component, we have:

$$\frac{h_d(i+1)}{h_d(i)} \geq \left(2^n - 2\sigma_\alpha - 4i - \sum_{j=1}^{\alpha+i} \widetilde{\mu}_{\alpha+i+1,j} + 2\delta + \left(\frac{4i^2-8i-4i\Delta-8i\delta}{2^n}\right)\left(1 - \frac{2(\sigma_\alpha+2i)}{2^n-2(\sigma_\alpha+2i)}\right)\right). \tag{6}$$

Having the number of solutions for $\mathsf{D}_{i+1}$ in terms of the number of solutions to $\mathsf{D}^i$, we find out the number of solutions to the remaining variables as follows.

### 3.1.3 Lower Bound on the Number of Solutions to Remaining Variables.

Now, we lower bound the number of solutions for $\mathcal{V} \setminus \mathcal{V}^=$. Recall that $|\mathcal{V} \setminus \mathcal{V}^=| = k'$. Fix such a vertex $Y_{\sigma_\alpha+2\beta+i}$ and let us assume that $\mu'_{\sigma_\alpha+2\beta+i}$ many blue dashed edges are incident on $Y_{\sigma_\alpha+2\beta+i}$. Let $y$ be assigned to the variable $Y_{\sigma_\alpha+2\beta+i}$. For $y$ to be a valid assignment, it must satify the following:

- $y$ should be distinct from previous $\sigma_\alpha + 2\beta$ many assigned values,
- $y$ should be distinct from $(i-1)$ many assigned values to the variables of the set $\mathcal{V} \setminus \mathcal{V}^=$,
- $y$ should not take $\mu'_{\sigma_\alpha+2\beta+i}$ values such that it violates the non-equality conditions of $\mu'_{\sigma_\alpha+2\beta+i}$ many blue dashed edges.

Therefore, the number of valid choices of $y$ is at least $(2^n - \sigma_\alpha - 2\beta - i + 1 - \mu'_{\sigma_\alpha+2\beta+i})$. Summarizing everything, the total number of possible injective solutions for the remaining vertices is at least

$$\prod_{i \in [k']} (2^n - \sigma_\alpha - 2\beta - i + 1 - \mu'_{\sigma_\alpha+2\beta+i}). \tag{7}$$

Therefore, from Eqn. (3), Eqn. (6) and Eqn. (7), we have

$$h(\mathsf{G})\frac{2^{nq_m}}{(2^n)_s} \geq \underbrace{\frac{h_c(\alpha)2^{nq_c}}{(2^n)_{\sigma_\alpha}}}_{\mathsf{A.1}} \cdot \underbrace{\prod_{i=0}^{\beta-1} \frac{h_d(i+1)}{h_d(i)} \cdot \frac{2^n}{(2^n-\sigma_\alpha-2i)_2}}_{\mathsf{A.2}} \cdot \underbrace{\prod_{i \in [k']} \frac{(2^n - \sigma_\alpha - 2\beta - i + 1 - \mu'_{\sigma_\alpha+2\beta+i})}{(2^n - \sigma_\alpha - 2\beta - i + 1)}}_{\mathsf{A.3}} \tag{8}$$

### 3.1.4 Algebraic Calculation.

In this section, we individually bound A.1, A.2 and A.3. We begin with bounding A.1 as follows:

Bounding A.1: Recall that $\sigma_\alpha = q_c + \alpha$. By using Eqn. (3), we lower bound A.1 as follows:

$$\mathsf{A.1} \geq \frac{h_c(\alpha)}{2^{n\alpha}} \geq \left(1 - \frac{9q_c^2}{4 \cdot 2^n} - \frac{q'_v}{2^n}\right).$$

Bounding A.2: We would like to note that $(\sigma_\alpha + 2i) \leq q_m \leq 2^{n-2}$. Therefore, from Eqn. (6) we have

$$\begin{aligned}
\mathsf{A.2} &\overset{(5)}{\geq} \prod_{i=0}^{\beta-1} \frac{\left(2^{2n} - 2^{n+1}\sigma_\alpha - 2^n 4i - 2^n \sum_{j=1}^{\alpha+i} \widetilde{\mu}_{\alpha+i+1,j} + 4i^2 - 8i - \frac{16q_m i^2}{2^n}\right)}{(2^n - \sigma_\alpha - 2i)_2} \\
&\overset{(6)}{\geq} \prod_{i=0}^{\beta-1} \left(1 - \frac{4\sigma_\alpha^2 + 16i\sigma_\alpha + 4\sigma_\alpha + 40i}{2^{2n}} - \frac{16q_m i^2}{2^{3n}} - \sum_{j=1}^{\alpha+i} \frac{4\widetilde{\mu}_{\alpha+i+1,j}}{2^n}\right) \\
&\geq \left(1 - \sum_{i=0}^{\beta-1} \frac{4\sigma_\alpha^2 + 16i\sigma_\alpha + 4\sigma_\alpha + 40i}{2^{2n}} - \sum_{i=0}^{\beta-1} \frac{16q_m i^2}{2^{3n}} - \sum_{i=0}^{\beta-1} \sum_{j=1}^{\alpha+i} \frac{4\widetilde{\mu}_{\alpha+i+1,j}}{2^n}\right) \\
&\overset{(7)}{\geq} \left(1 - \frac{4q_m\sigma_\alpha^2 + 16\sigma_\alpha q_m^2 + 4\sigma_\alpha q_m + 40q_m^2}{2^{2n}} - \frac{16q_m^4}{2^{3n}} - \frac{4q''_v}{2^n}\right) \\
&\overset{(8)}{\geq} \left(1 - \frac{9q_c^2 q_m + 24q_c q_m^2 + 6q_c q_m + 40q_m^2}{2^{2n}} - \frac{16q_m^4}{2^{3n}} - \frac{4q''_v}{2^n}\right),
\end{aligned}$$

where (5) holds due to the fact that $2\delta \geq 4i\Delta/2^n + 8i\delta/2^n$ when $\delta = \Delta$ and $(\sigma_\alpha + 2\beta) \leq 2^n/6$. (6) holds due to the fact that $(2^n - \sigma_\alpha - 2i)_2 \geq (2^n - \sigma_\alpha - 2i - 1)^2$ and $(\sigma_\alpha + 2i + 1) \leq 2^{n-1}$. Moreover, (7) holds due to the fact that $\beta \leq q_m$ and we define

$$q''_v \triangleq \sum_{i=0}^{\beta-1} \sum_{j=1}^{\alpha+i} \frac{4\widetilde{\mu}_{\alpha+i+1,j}}{2^n}.$$

(8) holds as $\sigma_\alpha \leq 3q_c/2$.

Bounding A.3: For bounding A.3, we have

$$\begin{aligned}
\mathsf{A.3} &= \prod_{i=1}^{k'} \frac{(2^n - \sigma_\alpha - 2\beta - i + 1 - \mu'_{\sigma_\alpha+2\beta+i})}{(2^n - \sigma_\alpha - 2\beta - i + 1)} \geq \prod_{i=1}^{k'} \left(1 - \frac{\mu'_{\sigma_\alpha+2\beta+i}}{(2^n - \sigma_\alpha - 2\beta - i + 1)}\right) \\
&\overset{(9)}{\geq} \left(1 - \sum_{i=1}^{k'} \frac{2\mu'_{\sigma_\alpha+2\beta+i}}{2^n}\right) \overset{(10)}{\geq} \left(1 - \frac{2q'''_v}{2^n}\right),
\end{aligned}$$

where (9) follows due to the fact that $(\sigma_\alpha + 2\beta + i - 1) \leq 2^{n-1}$ and (10) follows as we denote $(\mu'_{\sigma_\alpha+2\beta+1} + \ldots + \mu'_{\sigma_\alpha+2\beta+k'}) = q'''_v$, the total number of blue dashed edges incident on the vertices $\mathcal{V} \setminus \mathcal{V}^{=}$.

Merging three bound: In the final step, we merge the bound that we obtained from A.1, A.2 and A.3. Therefore, by plug-in the lower bounds of A.1, A.2 and A.3 into Eqn. (8), we obtain

$$h(\mathsf{G}) \geq \frac{(2^n)_s}{2^{nq_m}} \left(1 - \frac{9q_c^2}{4 \cdot 2^n} - \frac{9q_c^2 q_m + 24q_c q_m^2 + 6q_c q_m + 40q_m^2}{2^{2n}} - \frac{16q_m^4}{2^{3n}} - \frac{7q_v}{2^n}\right),$$

where the above inequality follows as $q'_v + q''_v + q'''_v = q_v$, the total number of non-equation edges.

## 3.2 Proof of Lemma 2

In this section, we prove Lemma 2. Let $P = Y_u$ and $Q = Y_v$ such that $u$ and $v$ are in the different components and $\lambda^*_{i+1} \neq \lambda^*_a, \lambda^*_{i+1} \neq \lambda^*_b, \lambda^*_{i+1} \neq \lambda^*_a \oplus \lambda^*_b$, where $\lambda^*_a$ (resp $\lambda^*_b$) is the label of the edge whose end vertex is $Y_u$ (resp $Y_v$). By removing the two equations whose constant part is $\lambda^*_a$ and $\lambda^*_b$, we derive

$$h'(P,Q) \geq (2^n - 2(\sigma_\alpha + 2i - 4))h_d(i-2) \geq (2^n - 2(\sigma_\alpha + 2i))h_d(i-2).$$

Moreover, we have

$$(2^n - (\sigma_\alpha + 2i - 4))(2^n - (\sigma_\alpha + 2i - 2))h_d(i-2) \geq h_d(i).$$

Therefore, from the above two equations with the trivial inequality that $2^{2n} \geq (2^n - (\sigma_\alpha + 2i - 4))(2^n - (\sigma_\alpha + 2i - 2))$, we obtain the result. □

## 3.3 Proof of Theorem 2

We prove the result in the same way as we proved Theorem 1, i.e., we lower bound on $h_c(\alpha)$, and then we estimate a lower bound on $h_d(\beta)$, and finally we estimate a lower bound on the number of solutions to $\mathsf{G} \setminus \mathsf{G}^{=}$. For the $i$-th component of $\mathsf{C}$, i.e., $\mathsf{C}_i$, which is acyclic and labelled bipartite graph, let $\mathcal{V}^{=}_{\mathsf{C}_i}$ be the set of vertices of the component $\mathsf{C}_i$. Let $\mathcal{V}^{\uparrow}_i = \mathcal{V}_1 \cap \mathcal{V}^{=}_{\mathsf{C}_i}$ be the set of vertices of one part of $\mathsf{C}_i$ and $\mathcal{V}^{\downarrow}_i = \mathcal{V}_2 \cap \mathcal{V}^{=}_{\mathsf{C}_i}$ be the set of vertices of the other part of $\mathsf{C}_i$. Let $s_{\ell,i} = |\mathcal{V}^{\uparrow}_i|$ and $s_{\mathtt{r},i} = |\mathcal{V}^{\downarrow}_i|$. For $1 \leq i \leq \alpha$, we write $\sigma_i = (s_{\ell,1} + s_{\mathtt{r},1}) + \ldots + (s_{\ell,i} + s_{\mathtt{r},i})$, with the convention that $\sigma_0 = 0$. Note that $q_c = \sigma_\alpha - \alpha$ as each component $\mathsf{C}_i$ is a tree.

### 3.3.1 Lower Bound on $h_c(\alpha)$.

We lower bound $h_c(\alpha)$ by counting the number of solutions in each of the $\alpha$ components of $\mathsf{C}$. For the first component, $\mathsf{C}_1$, there are $2^n$ ways to assign values to any one of the vertices of $\mathcal{V}^{\uparrow}_1$, which uniquely determines the values of all the vertices of $\mathcal{V}^{\downarrow}_1 \cup \mathcal{V}^{\uparrow}_1$. For assigning values to a vertex of $\mathcal{V}^{\uparrow}_2$ of the second component $\mathsf{C}_2$, it cannot take $s_{\ell,1}s_{\ell,2} + s_{\mathtt{r},1}s_{\mathtt{r},2}$ values. Additionally, as there are $\widetilde{\mu}_{2,1}$ many blue dashed edges connecting the component $\mathsf{C}_1$ and $\mathsf{C}_2$, there are $\widetilde{\mu}_{2,1}$ many paths from the assigned vertex to the vertices of the component $\mathsf{C}_1$ and hence it cannot take $\widetilde{\mu}_{2,1}$ values that violate the non-equality conditions of $\widetilde{\mu}_{2,1}$ many blue dashed edges. As a result, there are at least $(2^n - s_{\ell,1}s_{\ell,2} - s_{\mathtt{r},1}s_{\mathtt{r},2} - \widetilde{\mu}_{2,1})$ valid choices. In general, for the $i$-th component, there are at least $(2^n - (s_{\ell,1} + \ldots + s_{\ell,(i-1)})s_{\ell,i} - (s_{\mathtt{r},1} + \ldots + s_{\mathtt{r},(i-1)})s_{\mathtt{r},i} - \widetilde{\mu}_{i,1} - \ldots - \widetilde{\mu}_{i,i-1})$ injective solutions for the $i$-th component. For notational simplicity, we write $\delta_i = (\widetilde{\mu}_{i,1} + \ldots + \widetilde{\mu}_{i,i-1})$. Hence, we have

$$\begin{aligned}
h_c(\alpha) &\geq \prod_{i=1}^{\alpha} \Big( (2^n - (s_{\ell,1} + \ldots + s_{\ell,(i-1)})s_{\ell,i} - (s_{\mathtt{r},1} + \ldots + s_{\mathtt{r},(i-1)})s_{\mathtt{r},i} - \delta_i \Big) \\
&= 2^{n\alpha} \prod_{i=1}^{\alpha} \Big( 1 - \frac{1}{2^n} \sum_{j=1}^{i-1} (s_{\ell,j}s_{\ell,i} + s_{\mathtt{r},j}s_{\mathtt{r},i}) - \frac{\delta_i}{2^n} \Big) \\
&\geq 2^{n\alpha} \Big( 1 - \frac{1}{2^n} \sum_{1 \leq i < j \leq \alpha} ((s_{\ell,i}s_{\ell,j} + s_{\mathtt{r},i}s_{\mathtt{r},j})) - \sum_{i=1}^{\alpha} \frac{\delta_i}{2^n} \Big) \\
&\overset{(1)}{\geq} 2^{n\alpha} \Big( 1 - \frac{1}{2^n} \Big( \sum_{i=1}^{\alpha} (s_{\ell,i} + s_{\mathtt{r},i}) \Big)^2 - \frac{q'_v}{2^n} \Big) \\
&\overset{(2)}{\geq} 2^{n\alpha} \Big( 1 - \frac{9q_c^2}{4 \cdot 2^n} - \frac{q'_v}{2^n} \Big), \qquad (9)
\end{aligned}$$

where (1) holds as $\delta_1 + \delta_2 + \ldots + \delta_\alpha = q'_v$, the total number of blue dashed edges across the components of $\mathsf{G}^{=}$ and (2) holds as $(s_{\ell,1} + s_{\mathtt{r},1} \ldots + s_{\ell,\alpha} + s_{\mathtt{r},\alpha}) = q_c + \alpha$ and $\alpha \leq q_c/2$.

### 3.3.2 Lower Bound on $h_d(\beta)$.

We want to have a lower bound of $h_d(i+1)$ in terms of $h_d(i)$. Let us denote the label of the edges in the component $\mathsf{D}_i$ is $\lambda_i^*$ and recall that

$$|\mathcal{V}_1^{\uparrow} \sqcup \ldots \sqcup \mathcal{V}_\alpha^{\uparrow}| = s_{\ell,1} + \ldots s_{\ell,\alpha}, \quad |\mathcal{V}_1^{\downarrow} \sqcup \ldots \sqcup \mathcal{V}_\alpha^{\downarrow}| = s_{\mathtt{r},1} + \ldots s_{\mathtt{r},\alpha}.$$

Let $\mathcal{V}^{\uparrow}$ denotes the set $\mathcal{V}_1^{\uparrow} \sqcup \ldots \sqcup \mathcal{V}_\alpha^{\uparrow}$ and $\mathcal{V}^{\downarrow}$ denotes the set $\mathcal{V}_1^{\downarrow} \sqcup \ldots \sqcup \mathcal{V}_\alpha^{\downarrow}$. Let $s^{\uparrow} \stackrel{\Delta}{=} |\mathcal{V}^{\uparrow}| = s_{\ell,1} + \ldots s_{\ell,\alpha}$ and $s^{\downarrow} \stackrel{\Delta}{=} |\mathcal{V}^{\downarrow}| = s_{\mathtt{r},1} + \ldots s_{\mathtt{r},\alpha}$. We write the vertex of one part of $\mathsf{D}_i$ as $Y_{s^{\uparrow}+i}$ and the other part as $Z_{s^{\downarrow}+i}$, i.e.,

$$Y_{s^{\uparrow}+i} \oplus Z_{s^{\downarrow}+i} = \lambda_i^*.$$

Now, we consider the component $\mathsf{D}_{i+1}$. For $(\alpha+i+1)$-th component $\mathsf{D}_{i+1}$, $\widetilde{\mu}_{\alpha+i+1,j}$ denotes the number of blue dashed edges connecting component $\mathsf{D}_{i+1}$ and the $j$-th component, where $j \in [\alpha+i]$. Note that the one end vertex of each such edge is a vertex from component $\mathsf{D}_{i+1}$ and the other end vertex resides in the $j$-th component, where $j \in [\alpha+i]$. We represent the end vertex of each such edge which resides in the $j$-th component, $j \in [\alpha+i]$ as a single-ton set $A_k^j$, where $k \in [\widetilde{\mu}_{\alpha+i+1,j}]$. Now, we are interested in obtaining a lower bound on $h_d(i+1)$ as follows. Let $Y_{s^{\uparrow}+i+1}$ be the vertex of $\mathsf{D}_{i+1}$. Then, $Y_{s^{\uparrow}+i+1}$ must satisfy the followings:

1. $Y_{s^{\uparrow}+i+1} \notin \mathcal{V}^{\uparrow} \sqcup \{Y_{s^{\uparrow}+1}, \ldots, Y_{s^{\uparrow}+i}\} (\stackrel{\Delta}{=} \mathcal{Z}_1)$
2. $Y_{s^{\uparrow}+i+1} \notin \left(\mathcal{V}^{\downarrow} \oplus \lambda_{i+1}^*\right) \sqcup \left(\{Z_{s^{\downarrow}+1}, \ldots, Z_{s^{\downarrow}+i}\} \oplus \lambda_{i+1}^*\right) (\stackrel{\Delta}{=} \mathcal{Z}_2)$
3. $Y_{s^{\uparrow}+i+1} \notin A_j^k$ where $j \in [\alpha+i]$ and $k \in [\widetilde{\mu}_{\alpha+i+1,j}]$.

Note that $|\mathcal{Z}_1| = (s^{\uparrow} + i)$ and $|\mathcal{Z}_2| = (s^{\downarrow} + i)$. Applying the inclusion-exclusion principle, we have

$$h_d(i+1) \quad \geq \quad \left(2^n - \sigma_\alpha - 2i - \sum_{j=1}^{\alpha+i} \widetilde{\mu}_{\alpha+i+1,j}\right) h_d(i) + \sum_{\substack{P \in \{Y_{s^{\uparrow}+1}, \ldots, Y_{s^{\uparrow}+i}\} \\ Q \in \{Z_{s^{\downarrow}+1}, \ldots, Z_{s^{\downarrow}+i}\}}} h'(P,Q), \quad (10)$$

where $h'(P,Q)$ denotes the number of solutions to $\mathsf{C} \cup \widetilde{\mathsf{D}}^i$, where $\widetilde{\mathsf{D}}^i = \mathsf{D}^i \cup P \stackrel{\lambda_{i+1}^*}{\rightarrow} Q$. We say two indices $u \in \{s^{\uparrow}+1, \ldots, s^{\uparrow}+i\}, v \in \{s^{\downarrow}+1, \ldots, s^{\downarrow}+i\}$ are in the same component of $\mathsf{D}^i$, if there is an edge between vertices $Y_u$ and $Z_v$ in the subgraph $\mathsf{D}^i$. Now, there are the following two cases:

- If $P = Y_u$ and $Q = Z_v$ such that $u$ and $v$ are in the same component of $\mathsf{D}^i$, and $\lambda_{i+1}^* = \lambda$ ($\lambda$ is the label of the edge connecting vertices $Y_u$ and $Z_v$), then $h'(P,Q) = h_d(i)$. Moreover, if $\lambda_{i+1}^* \neq \lambda$, then $h'(P,Q) = 0$.

- Otherwise, we consider the case when $u$ and $v$ are in different components, and $\lambda_{i+1}^* \neq \lambda_a^*, \lambda_{i+1}^* \neq \lambda_b^*, \lambda_{i+1}^* \neq \lambda_a^* \oplus \lambda_b^*$, where $\lambda_a^*$ (resp $\lambda_b^*$) is the label of the edge whose end vertex is $Y_u$ (resp $Z_v$). In this case, we estimate a lower bound on $h'(P,Q)$. Note that when $u$ and $v$ are in different components and if any of the above conditions hold, then $h'(P,Q) = 0$.

The following lemma gives a lower bound on $h'(P,Q)$ in terms of $h_d(i)$, proof of which can be found in Eqn. (4) of [KLL20].

**Lemma 3.** $h'(P,Q) \geq \frac{h_d(i)}{2^n}\left(1 - \frac{2(\sigma_\alpha+2i)}{2^n}\right)$.

Now, we define the following two sets:

- $\mathcal{S}_1 \triangleq \{k \in [i] : \lambda^*_k = \lambda^*_{i+1}\}$ and let $\delta \triangleq |\mathcal{S}_1|$
- $\mathcal{S}_2 \triangleq \{(k,k') \in [i] \times [i] : k \neq k', \lambda^*_k \neq \lambda^*_{i+1}, \lambda^*_{k'} \neq \lambda^*_{i+1}, \lambda^*_{i+1} \neq \lambda^*_k \oplus \lambda^*_{k'}\}$.

Recall that, $\delta$ is the number of multi-collisions of $\lambda^*_{i+1}$. Now, using the similar argument, one can check that

$$|\mathcal{S}_2| \geq i(i-1) - 2i\delta. \tag{11}$$

Therefore, Eqn. (10), Eqn. (11), Lemma 3, and the above two cases where $(u,v)$ either belongs to the same component or in different component lead us to the following inequality:

$$\begin{aligned}\frac{h_d(i+1)}{h_d(i)} &\geq \left(2^n - \sigma_\alpha - 2i - \sum_{j=1}^{\alpha+i}\widetilde{\mu}_{\alpha+i+1,j} + \delta + \frac{i(i-1)-2i\delta}{2^n}\left(1 - \frac{2(\sigma_\alpha+2i)}{2^n}\right)\right)\\ &\overset{(1)}{\geq} \left(2^n - \sigma_\alpha - 2i - \sum_{j=1}^{\alpha+i}\widetilde{\mu}_{\alpha+i+1,j} + \frac{i(i-1)}{2^n}\left(1 - \frac{2(\sigma_\alpha+2i)}{2^n}\right)\right),\end{aligned} \tag{12}$$

where (1) holds as $i \leq 2^{n-1}$.Having the number of solutions for $\mathsf{D}_{i+1}$ in terms of the number of solutions to $\mathsf{D}^i$, we find out the number of solutions to the remaining variables as follows:

### 3.3.3 Lower Bound on the Number of Solutions to Remaining Variables.

Now, we lower bound the number of solutions for $\mathcal{V} \setminus \mathcal{V}^=$. Recall that $|\mathcal{V} \setminus \mathcal{V}^=| = k'$. Fix such a vertex and let us assume that $\mu'_{\sigma_\alpha+2\beta+i}$ many blue dashed edges are incident on it. Let $y$ be assigned to the variable. For $y$ to be a valid assignment, it must have the following:

- $y$ should be distinct from previous $\sigma_\alpha + 2\beta$ many assigned values.
- $y$ should be distinct from $(i-1)$ many assigned values to the variables of the set $\mathcal{V} \setminus \mathcal{V}^=$.
- $y$ should not take $\mu'_{\sigma_\alpha+2\beta+i}$ values such that it violates the non-equality conditions of $\mu'_{\sigma_\alpha+2\beta+i}$ many blue dashed edges.

Therefore, the number of valid choices of $y$ is at least $(2^n - \sigma_\alpha - 2\beta - i + 1 - \mu'_{\sigma_\alpha+2\beta+i})$. Summarizing above, the total number of possible injective solutions for the remaining vertices is at least

$$\prod_{i\in[k']}(2^n - \sigma_\alpha - 2\beta - i + 1 - \mu'_{\sigma_\alpha+2\beta+i}). \tag{13}$$

From Eqn. (9), Eqn. (12) and Eqn. (13), we have

$$\begin{aligned}h(\mathsf{G})\frac{2^{nq_m}}{(2^n)_{s_\ell}(2^n)_{s_r}} \geq\ & \underbrace{\frac{h_c(\alpha)2^{nq_c}}{(2^n)_{s^\uparrow}(2^n)_{s^\downarrow}}}_{\mathsf{A.1}} \cdot \underbrace{\prod_{i=0}^{\beta-1}\frac{h_d(i+1)}{h_d(i)} \cdot \frac{2^n}{(2^n - s^\uparrow - i)(2^n - s^\downarrow - i)}}_{\mathsf{A.2}}\\ & \cdot \underbrace{\prod_{i\in[k']}\frac{(2^n - \sigma_\alpha - 2\beta - i + 1 - \mu'_{\sigma_\alpha+2\beta+i})}{(2^n - \sigma_\alpha - 2\beta - i + 1)}}_{\mathsf{A.3}}.\end{aligned} \tag{14}$$

### 3.3.4 Algebraic Calculation.

In this section, we individually bound A.1, A.2 and A.3. We begin with bounding A.1 as follows:

Bounding A.1: Recall that $\sigma_\alpha = s^{\uparrow} + s^{\downarrow}$ and $\sigma_\alpha = q_c + \alpha$. By using Eqn. (9), we lower bound A.1 as follows:

$$\mathsf{A.1} \geq \frac{h_c(\alpha)}{2^{n\alpha}} \geq \left(1 - \frac{9q_c^2}{4 \cdot 2^n} - \frac{q'_v}{2^n}\right).$$

Bounding A.2: From Eqn. (12) we have

$$\begin{aligned}
\mathsf{A.2} &\geq \prod_{i=0}^{\beta-1}\left(\frac{2^{2n} - 2^n\sigma_\alpha - 2^n 2i - 2^n \sum_{j=1}^{\alpha+i} \widetilde{\mu}_{\alpha+i+1,j} + (i^2 - i)(1 - \frac{2(\sigma_\alpha+2i)}{2^n})}{(2^n - s^{\uparrow} - i)(2^n - s^{\downarrow} - i)}\right) \\
&= \prod_{i=0}^{\beta-1}\left(1 - \frac{(s^{\uparrow}+i)(s^{\downarrow}+i) + 2^n \sum_{j=1}^{\alpha+i} \widetilde{\mu}_{\alpha+i+1,j} - (i^2 - i)(1 - \frac{2(\sigma_\alpha+2i)}{2^n})}{2^{2n} - 2^n(\sigma_\alpha + 2i) + (s^{\uparrow}+i)(s^{\downarrow}+i)}\right) \\
&\geq \prod_{i=0}^{\beta-1}\left(1 - \frac{(s^{\uparrow}+i)(s^{\downarrow}+i) + 2^n \sum_{j=1}^{\alpha+i} \widetilde{\mu}_{\alpha+i+1,j} - (i^2 - i) + \frac{2(\sigma_\alpha+2i)i^2}{2^n}}{2^{2n}/2}\right) \\
&\geq \prod_{i=0}^{\beta-1}\left(1 - \frac{2s^{\uparrow}s^{\downarrow}}{2^{2n}} - \frac{2(\sigma_\alpha+1)i}{2^{2n}} - \frac{4(\sigma_\alpha+2i)i^2}{2^{3n}} - \sum_{j=1}^{\alpha+i}\frac{2\widetilde{\mu}_{\alpha+i+1,j}}{2^n}\right) \\
&\overset{(2)}{\geq} \left(1 - \sum_{i=0}^{\beta-1}\frac{9q_c^2}{4 \cdot 2^{2n}} - \sum_{i=0}^{\beta-1}\frac{3q_c i + 2i}{2^{2n}} - \sum_{i=0}^{\beta-1}\frac{8q_m i^2}{2^{3n}} - \sum_{i=0}^{\beta-1}\sum_{j=1}^{\alpha+i}\frac{2\widetilde{\mu}_{\alpha+i+1,j}}{2^n}\right) \\
&\overset{(3)}{\geq} \left(1 - \frac{9q_m q_c^2}{4 \cdot 2^{2n}} - \frac{3q_c q_m^2}{2 \cdot 2^{2n}} - \frac{q_m^2}{2^{2n}} - \frac{8q_m^4}{3 \cdot 2^{3n}} - \frac{2q''_v}{2^n}\right),
\end{aligned}$$

where (2) holds due to the fact that $\sigma_\alpha = s^{\uparrow} + s^{\downarrow} \leq 3q_c/2$ and (3) holds due to the fact that $\beta \leq q_m$ and $q''_v \triangleq ((\widetilde{\mu}_{\alpha+1,1} + \ldots + \widetilde{\mu}_{\alpha+1,\alpha}) + \ldots + (\widetilde{\mu}_{\alpha+\beta,1} + \ldots + \widetilde{\mu}_{\alpha+\beta,\alpha+\beta-1}))$.

Bounding A.3: For bounding A.3, we have

$$\begin{aligned}
\mathsf{A.3} &= \prod_{i=1}^{k'}\frac{(2^n - \sigma_\alpha - 2\beta - i + 1 - \mu'_{\sigma_\alpha+2\beta+i})}{(2^n - \sigma_\alpha - 2\beta - i + 1)} \geq \prod_{i=1}^{k'}\left(1 - \frac{\mu'_{\sigma_\alpha+2\beta+i}}{(2^n - \sigma_\alpha - 2\beta - i + 1)}\right) \\
&\overset{(4)}{\geq} \left(1 - \sum_{i=1}^{k'}\frac{2\mu'_{\sigma_\alpha+2\beta+i}}{2^n}\right) \overset{(5)}{\geq} \left(1 - \frac{2q'''_v}{2^n}\right),
\end{aligned}$$

where (4) follows due to the fact that $(\sigma_\alpha + 2\beta + i - 1) \leq 2^{n-1}$ and (5) follows as we denote $(\mu'_{\sigma_\alpha+2\beta+1} + \ldots + \mu'_{\sigma_\alpha+2\beta+k'}) = q'''_v$, the total number of blue dashed edges incident on the vertices $\mathcal{V} \setminus \mathcal{V}^{=}$.

Merging three bound: In the final step we merge the bound that we obtained from A.1, A.2 and A.3. With the fact that $q'_v + q''_v + q'''_v = q_v$, the total number of non-equation edges and by plugging-in the lower bounds of A.1, A.2 and A.3 into Eqn. (14), we obtain

$$h(\mathsf{G}) \geq \frac{(2^n)_{s_\ell}(2^n)_{s_r}}{2^{nq_m}}\left(1 - \frac{9q_c^2}{4 \cdot 2^n} - \frac{9q_m q_c^2}{4 \cdot 2^{2n}} - \frac{3q_c q_m^2}{2 \cdot 2^{2n}} - \frac{q_m^2}{2^{2n}} - \frac{8q_m^4}{3 \cdot 2^{3n}} - \frac{5q_v}{2^n}\right).$$

*Remark* 2. We would like to note here that the proof of Theorem 1 and Theorem 2 differs from that of [KLL20] as the proof in [KLL20] takes care of only lower bounding the number of solutons to a system of bivariate affine equations, whereas our result takes care in lower bounding the number of solutions to a system of bivariate affine equations and non-equations. That is why, while counting the number of solutions in Sect. 3.1.1 and Sect. 3.3.1 for extended mirror theory, we discarded the choices which violated the non-equality conditions. Such restrictions were not present in [KLL20].

# 4 Security Result of EWCDM

In this section we state and prove that EWCDM can be secured up to $2^{3n/4}$ MAC queries and $2^n$ verification queries against nonce respecting adversaries. The following result bounds the MAC advantage of EWCDM against nonce respecting adversaries.

**Theorem 3.** *Let $\mathcal{M}$ and $\mathcal{K}$ be finite and non-empty sets. Let $\mathsf{E} : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher and $\mathsf{H} : \mathcal{K}_h \times \mathcal{M} \to \{0,1\}^n$ be an $\epsilon_{\text{axu}}$-AXU hash function. Then, the MAC advantage for any $(q_m, q_v, t)$ nonce respecting adversary against $\mathsf{EWCDM}[\mathsf{E}, \mathsf{H}]$ is given by,*

$$\begin{aligned}\mathbf{Adv}^{\text{nMAC}}_{\mathsf{EWCDM}}(q_m, q_v, t) \quad \leq \quad & 2\mathbf{Adv}^{\text{PRP}}_{\mathsf{E}}(q_m + q_v, t') + \frac{4q_m^{4/3}}{2^n} + q_v\epsilon_{\text{axu}} + \frac{q_m^2\epsilon_{\text{axu}}}{2^n} + \frac{9q_m^{7/3}}{4 \cdot 2^{2n}} \\ & + \frac{3q_m^{8/3}}{2 \cdot 2^{2n}} + \frac{q_m^2}{2^{2n}} + \frac{8q_m^4}{3 \cdot 2^{3n}} + \frac{5q_v}{2^n},\end{aligned}$$

*where $t' = O(t + (q_m + q_v)t_H)$, $t_H$ be the time for computing the hash function. Assuming $\epsilon_{\text{axu}} \approx 2^{-n}$, EWCDM is secured up to roughly $q_m \approx 2^{3n/4}$ MAC queries and $q_v \approx 2^n$ verification queries.*

## 4.1 Proof of Theorem 3

For the sake of notational simplicity, we refer to the construction $\mathsf{EWCDM}[\mathsf{E}, \mathsf{H}]$ simply as EWCDM when the primitives are understood from the context. As the first step of the proof, we replace two independent block ciphers of the construction with two independently sampled $n$-bit uniform random permutations $\pi_1$ and $\pi_2$ at the cost of the sprp advantage of E and denote the resulting construction as $\mathsf{EWCDM}^*[\pi_1, \pi_2, \mathsf{H}]$, i.e.,

$$\mathbf{Adv}^{\text{nMAC}}_{\mathsf{EWCDM}}(q_m, q_v, t) \leq 2\mathbf{Adv}^{\text{PRP}}_{\mathsf{E}}(q_m + q_v, t') + \mathbf{Adv}^{\text{nMAC}}_{\mathsf{EWCDM}^*}(q_m, q_v).$$

Instead of arguing the security of $\mathsf{EWCDM}^*$, we argue the security of $\mathsf{EWCDM}^*[\pi_1, \pi_2^{-1}, \mathsf{H}]$, which we denote as $\mathsf{EWCDM}^+$. Note that the distinguishing advantage of the adversary $\mathbb{D}$ for the latter is identical to the former as $\pi_1, \pi_2$ are mutually independent. The advantage of analysing the security of the latter construction is that it is convenient to argue the security of $\mathsf{EWCDM}^+$ as one can view an evaluation $T = \mathsf{EWCDM}^+(\nu, M)$ as the xor of two permutations in the middle of the function, i.e.,

$$\pi_1(\nu) \oplus \pi_2(T) = \nu \oplus \mathsf{H}_{k_h}(M).$$

Our goal is to upper bound the information-theoretic MAC security of $\mathsf{EWCDM}^+$. For doing this, we resort to the Eqn.(1) which allows us to bound the MAC security of $\mathsf{EWCDM}^+$ in terms of the distingusihing advantage in distinguishing $\mathsf{EWCDM}^+$ from an ideal world consisting of a random oracle \$ that outputs a random tag on every input $(\nu, M) \in \{0,1\}^n \times \mathcal{M}$ and a reject oracle $\perp$ that always outputs 0 on every query $(\nu, M, T)$. At the end of the interaction, the real world releases the hash key $k_h$ and the ideal world releases a random dummy key $k_h$. As a result of it, we apply the H-Coefficient Technique [Pat08a] to bound the distinguishing advantage of $\mathsf{EWCDM}^+$. We

can represent an attainable transcript $\tau = (\tau_m, \tau_v, k_h)$ in terms of the following equations, where $\lambda_i = \nu_i \oplus \mathsf{H}_{k_h}(M_i)$.

$$(\mathcal{E}_m) = \begin{cases} \pi_1(\nu_1) \oplus \pi_2(T_1) = \lambda_1 \\ \pi_1(\nu_2) \oplus \pi_2(T_2) = \lambda_2 \\ \vdots \\ \pi_1(\nu_{q_m}) \oplus \pi_2(T_{q_m}) = \lambda_{q_m} \end{cases} \qquad (\mathcal{E}_v) = \begin{cases} \pi_1(\nu'_1) \oplus \pi_2(T'_1) \neq \lambda'_1 \\ \pi_1(\nu'_2) \oplus \pi_2(T'_2) \neq \lambda'_2 \\ \vdots \\ \pi_1(\nu'_{q_v}) \oplus \pi_2(T'_{q_v}) \neq \lambda'_{q_v} \end{cases}$$

## 4.2 Definition and Probability of Bad Transcripts

In this section, we define and bound the probability of bad transcripts in the ideal world. We say a transcript $\tau = (\tau_m, \tau_v, k_h)$ is $\mathsf{bad}$ if it satisfies either of the following conditions:

- $\mathsf{B.1} : |\{i \neq j \in [q_m] : T_i = T_j\}| \geq q_m^{2/3}$.
- $\mathsf{B.2} : \exists i \in [q_m], a \in [q_v]$ such that $\nu_i = \nu'_a, T_i = T'_a, \nu_i \oplus \mathsf{H}_{k_h}(M_i) = \nu'_a \oplus \mathsf{H}_{k_h}(M'_a)$.
- $\mathsf{B.3} : \exists i \neq j \in [q_m]$ such that $\nu_i \oplus \mathsf{H}_{k_h}(M_i) = \nu_j \oplus \mathsf{H}_{k_h}(M_j), T_i = T_j$.

Having defined the bad transcripts, we bound the probability of realizing bad transcripts in the ideal world as follows.

**Lemma 4.** *Let* $\mathsf{X}_{\mathrm{id}}$ *and* $\Theta_b$ *be defined as above. Then, we have*

$$\Pr[X_{\mathrm{id}} \in \Theta_b] \leq \frac{q_m^{4/3}}{2^n} + q_v \epsilon_{\mathrm{axu}} + \frac{q_m^2 \epsilon_{\mathrm{axu}}}{2^n}.$$

We defer the proof of the Lemma in Sect. 5.

## 4.3 Analysis of good transcripts

Let us consider $\tau = (\tau_m, \tau_v, k_h)$ be a good transcript and we show that realizing $\tau$ is almost as likely in the real world as in the ideal world. In particular, we prove the following result.

**Lemma 5.** *Let* $\tau = (\tau_m, \tau_v, k_h)$ *be a good transcript. Then*

$$\frac{\Pr[\mathsf{X}_{\mathrm{re}} = \tau]}{\Pr[\mathsf{X}_{\mathrm{id}} = \tau]} \geq \left(1 - \frac{9q_m^{4/3}}{4 \cdot 2^n} - \frac{9q_m^{7/3}}{4 \cdot 2^{2n}} - \frac{3q_m^{8/3}}{2 \cdot 2^{2n}} - \frac{q_m^2}{2^{2n}} - \frac{8q_m^4}{3 \cdot 2^{3n}} - \frac{5q_v}{2^n}\right).$$

**Proof.** Since the MAC oracle in the ideal world is perfectly random and the verification oracle always outputs 0, one simply has

$$\Pr[\mathsf{X}_{\mathrm{id}} = \tau] = \frac{1}{|\mathcal{K}_h|} \cdot \frac{1}{2^{nq_m}}. \tag{15}$$

To lower bound the real interpolation probability, we say that a pair of permutations $(\pi_1, \pi_2)$ is compatible with $\tau$ if

$$\begin{cases} \pi_1(\nu_i) \oplus \pi_2(T_i) = \nu_i \oplus \mathsf{H}_{k_h}(M_i), \forall i \in [q_m], \\ \pi_1(\nu'_a) \oplus \pi_2(T'_a) \neq \nu'_a \oplus \mathsf{H}_{k_h}(M'_a), \forall a \in [q_v]. \end{cases}$$

Let $\mathsf{Comp}(\tau)$ denotes the set of all pair of permutations $(\pi_1, \pi_2)$ that are compatible with $\tau$. Therefore, we have

$$\mathsf{p}_{\mathrm{re}}(\tau) \stackrel{\Delta}{=} \Pr[\mathsf{X}_{\mathrm{re}} = \tau] = \frac{1}{|\mathcal{K}_h|} \cdot \underbrace{\Pr[\pi_1, \pi_2 \leftarrow\!\!\$\, \mathsf{Perm} : (\pi_1, \pi_2) \in \mathsf{Comp}(\tau)]}_{\mathsf{P}_{mv}} \tag{16}$$

Lower bounding $\mathsf{P}_{mv}$ implies lower bounding the probability of the number of solutions to the system of $q_m$ many bivariate affine MAC equations and $q_v$ many bivariate affine verification non-equations $\mathcal{E}_m \cup \mathcal{E}_v$. From the above system of bivariate affine equations and non-equations, one can induce an edge-labelled undirected bipartite graph $\mathsf{G}_\tau = (\mathcal{V} = \mathcal{V}_1 \sqcup \mathcal{V}_2, \mathcal{E} \sqcup \mathcal{E}', \mathfrak{L})$, where the set of nodes $\mathcal{V}$ is partitioned into two sets, $\mathcal{V}_1 = \{Y_1, \ldots, Y_{s_\ell}\}$ and $\mathcal{V}_2 = \{Z_1, \ldots, Z_{s_r}\}$, $\mathcal{E}$ is the set of edges corresponding to each MAC equation, and $\mathcal{E}'$ is the set of edges corresponding to each verification non-equation. Therefore, $q_m = |\mathcal{E}|$ and $q_v = |\mathcal{E}'|$. Moreover, if there is a MAC equation $Y_u \oplus Z_v = T_i$, then the corresponding edge $\{Y_u, Z_v\} \in \mathcal{E}$ is labelled as $T_i$. Similarly, if there is a verification non-equation $Y_u \oplus Z_v \neq T'_i$, then the corresponding edge $\{Y_u, Z_v\} \in \mathcal{E}'$ is labelled as $T'_i$. Moreover, $\mathsf{G}^{=}_\tau = (\mathcal{V}^{=}, \mathcal{E}, \mathfrak{L}_{|\mathcal{E}})$ is the subgraph of $\mathsf{G}_\tau$. Now, it is easy to argue the following.

**Claim 1.** *For a good transcript $\tau$, the induced graph $\mathsf{G}_\tau$ is a good graph.*

**Proof.** For a good transcript $\tau$, note that the subgraph $\mathsf{G}^{=}_\tau$ contains only two types of components as follows: (a) star-type component and (b) component of a single edge as depicted in Fig. 4.1.

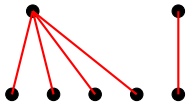

**Figure 4.1:** (a) star type component and (b) the component with a single edge

Note that for the star type component, if we consider any path of $\mathsf{G}^{=}_\tau$ of even length, then the label of the path is non-zero, otherwise bad condition B.3 would have been satisfied. Moreover, the graph is acyclic by construction, which proves the claim. □

Resuming the proof of Lemma 5. Since $\mathsf{G}_\tau$ is good, the graph is acyclic. Therefore, let us assume that there are $\alpha + \beta$ components in the subgraph $\mathsf{G}^{=}_\tau$ such that the size of each of the first $\alpha$ components are greater than 2 and the remaining $\beta$ components are of size 2 each. Moreover, due to the construction, each of the first $\alpha$ components is star type graph. As the transcript $\tau$ is good, the total number of edges in the first $\alpha$ components is at most $q_m^{2/3}$. Therefore, we apply Theorem 2 to obtain

$$\mathsf{P}_{mv} \geq \frac{1}{2^{nq_m}}\left(1 - \frac{9q_m^{4/3}}{4 \cdot 2^n} - \frac{9q_m^{7/3}}{4 \cdot 2^{2n}} - \frac{3q_m^{8/3}}{2 \cdot 2^{2n}} - \frac{q_m^2}{2^{2n}} - \frac{8q_m^4}{3 \cdot 2^{3n}} - \frac{5q_v}{2^n}\right), \tag{17}$$

Therefore, from Eqn. (16) and Eqn. (17), we have

$$\mathsf{p}_{\mathrm{re}}(\tau) \geq \frac{1}{|\mathcal{K}_h|} \cdot \frac{1}{2^{nq_m}} \cdot \left(1 - \frac{9q_m^{4/3}}{4 \cdot 2^n} - \frac{9q_m^{7/3}}{4 \cdot 2^{2n}} - \frac{3q_m^{8/3}}{2 \cdot 2^{2n}} - \frac{q_m^2}{2^{2n}} - \frac{8q_m^4}{3 \cdot 2^{3n}} - \frac{5q_v}{2^n}\right). \tag{18}$$

Finally, by taking the ratio of Eqn. (18) to Eqn. (15), we obtain the result. □

# 5 Proof of Lemma 4

Using the union bound, we write

$$\Pr[\mathsf{X}_{\mathrm{id}} \in \Theta_b] \leq \Pr[\mathsf{B}.1] + \Pr[\mathsf{B}.2] + \Pr[\mathsf{B}.3]. \tag{19}$$

In the following, we bound the probabilities of all the bad events individually.

**Bounding B.1:** Let $X$ denotes the cardinality of the set $\{\exists i \neq j \in [q_m] : T_i = T_j\}$ and for all $i \neq j \in [q_m]$, let $\mathbb{I}_{ij}$ be the indicator random variable that takes the value 1 if $T_i = T_j$,

otherwise it takes the value 0. Therefore,

$$X = \sum_{i \neq j} \mathbb{I}_{ij}.$$

By using the linearity of expectation, we have

$$\mathbf{E}[X] = \sum_{i \neq j} \mathbf{E}[\mathbb{I}_{ij}] = \sum_{i \neq j} 1.\Pr[\mathbb{I}_{ij} = 1] = \sum_{i \neq j} 1.\Pr[T_i = T_j] \leq \frac{q_m^2}{2^n}. \tag{20}$$

By applying Markov's inequality on Eqn. (20), we have

$$\Pr[\mathsf{B}.1] = \Pr[X \geq q_m^{2/3}] \leq \mathbf{E}[X]/q_m^{2/3} \leq q_m^{4/3}/2^n. \tag{21}$$

**Bounding B.2:** Recall that the event B.2 holds if $\exists i \in [q_m], a \in [q_v]$ such that $\nu_i = \nu'_a, T_i = T'_a, \nu_i \oplus \mathsf{H}_{k_h}(M_i) = \nu'_a \oplus \mathsf{H}_{k_h}(M'_a)$. Note that, for a fixed choice of $i, a$, the probability of the above event is at most $\epsilon_{\text{axu}}$ due to the randomness of the hash key. However, the number of choices for $i$ is at most one as the adversary is nonce-respecting. Therefore, by varying over all possible choices of $a$, we have

$$\Pr[\mathsf{B}.2] \leq q_v \epsilon_{\text{axu}}. \tag{22}$$

**Bounding B.3:** Recall that the event B.3 holds if $\exists i \neq j \in [q_m]$ such that $\nu_i \oplus \mathsf{H}_{k_h}(M_i) = \nu_j \oplus \mathsf{H}_{k_h}(M_j), T_i = T_j$. Since, in the ideal oracle, the hash key is sampled independently to all previously sampled MAC responses $T_i$, we write

$$\Pr[\mathsf{B}.3] \leq \sum_{i,j} \Pr[\mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M_j) = \nu_i \oplus \nu_j] \cdot \Pr[T_i = T_j] \leq \frac{q_m^2 . \epsilon_{\text{axu}}}{2^n}. \tag{23}$$

Finally, Lemma 4 follows from Eqn. (19)-(23). □

# 6 Security Result of DWCDM

In this section, we state and prove that DWCDM is secure up to $2^{3n/4}$ MAC queries and $2^n$ verification queries against nonce respecting adversaries. The following result bounds the MAC advantage of DWCDM against nonce respecting adversaries. For the sake of notational simplicity, we refer DWCDM as $\Pi$.

**Theorem 4.** *Let $\mathcal{M}$ and $\mathcal{K}$ be finite and non-empty sets. Let $\mathsf{E} : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher and $\mathsf{H} : \mathcal{K}_h \times \mathcal{M} \to \{0,1\}^n$ be an $\epsilon_{\text{reg}}$-regular and $\epsilon_{\text{axu}}$-AXU, $\epsilon_{3\text{-reg}}$-3-way regular, and $\epsilon_{4\text{-reg}}$-4-way regular. Then, the MAC advantage for any $(q_m, q_v, t)$ nonce respecting adversary against $\Pi[\mathsf{E}, \mathsf{E}^{-1}, \mathsf{H}]$ is given by,*

$$\begin{aligned}
\mathbf{Adv}^{\text{nMAC}}_{\Pi}(q_m, q_v, t) \leq\ & \mathbf{Adv}^{\text{SPRP}}_{\mathsf{E}}(q_m + q_v, t') + \frac{2q_m}{2^n} + \frac{q_m}{2^{3n/4}} + \frac{q_m^{1/3}}{2^{n/4}} + \frac{4q_m^{4/3}}{2^n} + \frac{2q_m^2 \epsilon_{\text{axu}}}{2^n} \\
& + \frac{q_m \epsilon_{\text{axu}}}{2^{3n/4}} + q_m \epsilon_{\text{reg}} + \max\{3q_v\epsilon_{4\text{-reg}}, 3q_v\epsilon_{3\text{-reg}}, 3q_v\epsilon_{\text{axu}}, q_v\epsilon_{\text{reg}}, \frac{q_m}{2^{3n/4}}\} \\
& + \frac{9q_m^{7/3} + 24q_m^{8/3} + 6q_m^{5/3} + 40q_m^2}{2^{2n}} + \frac{16q_m^4}{2^{3n}} + \frac{7q_v}{2^n} + \frac{q_m^2}{2^{2n}} + \frac{q_m}{2^{5n/4}} \\
& + \left(\frac{q_m^3}{2^{2n}} + \frac{q_m^2}{2^{5n/4}} + \frac{q_m}{2^{n/2}}\right) \cdot \epsilon_{3\text{-reg}},
\end{aligned}$$

*where $t' = O(t + (q_m + q_v)t_H)$, $t_H$ be the time for computing the hash function. Assuming $\epsilon_{\text{reg}}, \epsilon_{\text{axu}}, \epsilon_{3\text{-reg}}, \epsilon_{4\text{-reg}} \approx 2^{-n}$, we obtain the desired bound for* DWCDM.

We want to point out that the hash function's 3-way regular and 4-way regular properties do not necessarily demand longer hash keys. For example, 3-way and 4-way regular bound of Polyhash [MI11] function with $n$-bit key is $\ell/2^n$ [DDNY19, Proposition 1], where $\ell$ denotes the maximum number of message blocks.

## 6.1 Proof of Theorem 4

For the sake of simplicity, we will refer to the construction $\mathsf{\Pi}[\mathsf{E}, \mathsf{E}^{-1}, \mathsf{H}]$ as $\mathsf{\Pi}$ when the underlying primitives are understood from the context. As the first step of the proof, we replace the block cipher $\mathsf{E}$ and its inverse with an $n$-bit uniform random permutation $\pi$ and its inverse respectively. This comes at the cost of the sprp advantage of $\mathsf{E}$, and we denote the resulting construction as $\mathsf{\Pi}^*$. Our goal is to upper bound the information-theoretic MAC security of $\mathsf{\Pi}^*$. For doing this, we resort to the Eqn.(1), which allows us to bound the MAC security of $\mathsf{\Pi}^*$ in terms of the distinguishing advantage in distinguishing $\mathsf{\Pi}^*$ from an ideal world consisting of a random oracle \$ that outputs a random tag on every input $(\tilde{\nu}, M) \in \{0,1\}^{3n/4} \times \mathcal{M}$ and a reject oracle $\perp$ that always outputs 0 on every query $(\tilde{\nu}, M, T)$. As a result, we apply the H-Coefficient Technique [Pat08a] to bound the distinguishing advantage of $\mathsf{\Pi}^*$. For the sake of notational simplicity, we write $\nu = \tilde{\nu} \| 0^{n/4}$. As before, we can represent an attainable transcript $\tau = (\tau_m, \tau_v, k_h)$ in terms of the following equations:

$$(\mathcal{E}_m) = \begin{cases} \pi(\nu_1) \oplus \pi(T_1) = \lambda_1 \\ \pi(\nu_2) \oplus \pi(T_2) = \lambda_2 \\ \vdots \\ \pi(\nu_{q_m}) \oplus \pi(T_{q_m}) = \lambda_{q_m} \end{cases} \qquad (\mathcal{E}_v) = \begin{cases} \pi(\nu'_1) \oplus \pi(T'_1) \neq \lambda'_1 \\ \pi(\nu'_2) \oplus \pi(T'_2) \neq \lambda'_2 \\ \vdots \\ \pi(\nu'_{q_v}) \oplus \pi(T'_{q_v}) \neq \lambda'_{q_v}, \end{cases}$$

where $\lambda_i = \nu_i \oplus \mathsf{H}_{k_h}(M_i)$ and $\lambda'_i = \nu'_i \oplus \mathsf{H}_{k_h}(M_i)$. We associate an undirected, edge-labelled graph $\mathsf{G}_\tau = (\mathcal{V}, \mathcal{E} \cup \mathcal{E}', \mathfrak{L})$ corresponding to a transcript $\tau = (\tau_m, \tau_v, k_h)$ as follows: the set of vertices $\mathcal{V}$ of the graph is the set of all variables of the equations $\mathcal{E}_m \cup \mathcal{E}_v$. If any two variables are same then they correspond to the same vertices. $\mathcal{E}$ denotes the set of all edges $\{a, b\}$ with $a$ and $b$ both as variables in $\mathcal{E}_m$. The equation $a \oplus b = \lambda_\star \in \mathcal{E}_m$, for some $\lambda_\star$, serves as the label of the edge $\{a, b\}$. Moreover, $\mathcal{E}'$ denotes the set of all edges $\{a, b\}$ with $a$ and $b$ are both variables in $\mathcal{E}_m \cup \mathcal{E}_v$, and the equation $a \oplus b \neq \lambda_\star \in \mathcal{E}_v$, for some $\lambda_\star$, serves as the label of the edge $\{a, b\}$.

We say a cycle $C^= = (i_1, i_2, \ldots, i_p)$ of length $p$ in the graph $\mathsf{G}^= = (\mathcal{V}^=, \mathcal{E}, \mathfrak{L}_{|\mathcal{E}})$ is *valid* if the imposed equality pattern of $(\nu, T)$, generated out of $C^=$, derives the following equation:

$$\bigoplus_{i \in C^=} \left( \nu_i \oplus \mathsf{H}_{k_h}(M_i) \right) = \mathbf{0}.$$

We also consider a cycle $C^\neq = (i_1, i_2, \ldots, i_p)$ of length $p$ in $\mathsf{G}^=$, containing exactly one non-equation edge $e' \in \mathcal{E}'$ (i.e., all other edges of $C^\neq$ are elements of $\mathcal{E}$). We call $C^\neq$ to be *valid* if the imposed equality pattern of $(\nu, T)$ and $(\nu', T')$, generated out of $C^\neq$, derives the equation

$$\bigoplus_{i \in C^\neq \setminus e'} \left( \nu_i \oplus \mathsf{H}_{k_h}(M_i) \right) \oplus \left( \nu' \oplus \mathsf{H}_{k_h}(M') \right) = \mathbf{0},$$

where $e'$ represents the equation $\pi(\nu') \oplus \pi(T') = \nu' \oplus \mathsf{H}_{k_h}(M')$. Now, we state a simple result for MAC queries.

**Lemma 6.** *In the ideal world, for two fixed MAC queries, $(\nu_i, M_i, T_i)$ and $(\nu_j, M_j, T_j)$, we have*

$$(a)\ \textit{if } i < j, \Pr[T_j = \nu_i] = \frac{1}{2^n};\ (b)\ \textit{if } i > j, \Pr[T_j = \nu_i] = \frac{1}{2^{n/4}};\ (c)\ \Pr[T_i = T_j] = \frac{1}{2^n}.$$

**Proof.** If $(\nu_j, M_j, T_j)$ appears after the $i$-th query $(\nu_i, M_i, T_i)$, then the event that $T_j$ collides with $\nu_i$ holds with probability $2^{-n}$. Moreover, if $(\nu_j, M_j, T_j)$ appears before the $i$-th query $(\nu_i, M_i, T_i)$, then the event that $T_j$ collides with $\nu_i$ holds with probability exactly $1/2^{n/4}$. Here we use the fact that the probability of the last $n/4$ bits of $T_i$ set to all zero is $1/2^{n/4}$. □

## 6.2 Definition and Probability of Bad Transcripts

In this section, we define and bound the probability of bad transcripts in the ideal world. We say a transcript $\tau = (\tau_m, \tau_v, k_h)$ is $\mathsf{bad}$ if its associated graph $\mathsf{G}_\tau$ satisfies either of the following conditions:

- $\mathsf{B}.1$ : $\exists i \in [q_m]$ such that $T_i = \mathbf{0}$.
- $\mathsf{B}.2$ : $\mathsf{G}^{=}$ has a component of size at least 5.
- $\mathsf{B}.3$ : $|\{i \neq j \in [q_m] : \nu_i = T_j \vee T_i = T_j\}| \geq q_m^{2/3}$.
- $\mathsf{B}.4$ : $\mathsf{G}^{=}$ contains a valid cycle $C^{=}$ of any arbitrary length.
- $\mathsf{B}.5$ : $\mathsf{G}^{=}$ contains a valid cycle $C^{\neq}$ of any arbitrary length.

Moreover, $\tau$ is also said to be bad if

- $\mathsf{B}.6$ : $\exists i \neq j \in [q_m]$ such that $\nu_i \oplus \mathsf{H}_{k_h}(M_i) = \nu_j \oplus \mathsf{H}_{k_h}(M_j), T_i = T_j$.
- $\mathsf{B}.7$: $\exists i \neq j \in [q_m]$ such that $\nu_i = T_j, \nu_i \oplus \mathsf{H}_{k_h}(M_i) = \nu_j \oplus \mathsf{H}_{k_h}(M_j)$.
- $\mathsf{B}.8$ : $\exists i \in [q_m]$ such that $\mathsf{H}_{k_h}(M_i) = \nu_i$.
- $\mathsf{B}.9$ : $\mathsf{G}^{=}$ has a path of length 3 such that its label is zero.

Having defined the bad transcripts, we bound the probability of realizing bad transcripts in the ideal world as follows.

**Lemma 7.** *Let* $\mathsf{X}_{\mathrm{id}}$ *and* $\Theta_b$ *be defined as above. Then, we have*

$$\begin{aligned}\Pr[X_{\mathrm{id}} \in \Theta_b] \quad \leq \quad & \frac{2q_m}{2^n} + \frac{q_m}{2^{3n/4}} + \frac{q_m^{1/3}}{2^{n/4}} + \frac{q_m^{4/3}}{2^n} + \frac{2q_m^2\epsilon_{\mathrm{axu}}}{2^n} + \frac{q_m\epsilon_{\mathrm{axu}}}{2^{3n/4}} + q_m\epsilon_{\mathrm{reg}} + \frac{q_m^2}{2^{2n}} \\ & + \frac{q_m}{2^{5n/4}} + \max\{3q_v\epsilon_{4\text{-reg}}, 3q_v\epsilon_{3\text{-reg}}, 3q_v\epsilon_{\mathrm{axu}}, q_v\epsilon_{\mathrm{reg}}, \frac{q_m}{2^{3n/4}}\} \\ & + \left(\frac{q_m^3}{2^{2n}} + \frac{q_m^2}{2^{5n/4}} + \frac{q_m}{2^{n/2}}\right) \cdot \epsilon_{3\text{-reg}}.\end{aligned}$$

We defer the proof of the Lemma in Sect. 7.

## 6.3 Analysis of good transcripts

Let us consider $\tau = (\tau_m, \tau_v, k_h)$ a good transcript, and we show that realizing $\tau$ is almost as likely in the real world as in the ideal world. In particular, we prove the following result.

**Lemma 8.** *Let* $\tau = (\tau_m, \tau_v, k_h)$ *be a good transcript. Then*

$$\frac{\Pr[\mathsf{X}_{\mathrm{re}} = \tau]}{\Pr[\mathsf{X}_{\mathrm{id}} = \tau]} \geq \left(1 - \frac{9q_m^{4/3}}{4 \cdot 2^n} - \frac{9q_m^{7/3} + 24q_m^{8/3} + 6q_m^{5/3} + 40q_m^2}{2^{2n}} - \frac{16q_m^4}{2^{3n}} - \frac{7q_v}{2^n}\right).$$

**Proof.** Since the MAC oracle in the ideal world is perfectly random and the verification oracle always outputs 0, we obtain

$$\Pr[\mathsf{X}_{\mathrm{id}} = \tau] = \frac{1}{|\mathcal{K}_h|} \cdot \frac{1}{2^{nq_m}}. \tag{24}$$

To lower bound the real interpolation probability, we say that a permutation $\pi$ is compatible with $\tau$ if

$$\begin{cases} \pi(\nu_i) \oplus \pi(T_i) = \nu_i \oplus \mathsf{H}_{k_h}(M_i), \forall i \in [q_m], \\ \pi(\nu'_a) \oplus \pi(T'_a) \neq \nu'_a \oplus \mathsf{H}_{k_h}(M'_a), \forall a \in [q_v]. \end{cases}$$

Let $\mathsf{Comp}(\tau)$ denotes the set of permutations that are compatible with $\tau$. Therefore, we have

$$\mathsf{p}_{\mathrm{re}}(\tau) \triangleq \Pr[\mathsf{X}_{\mathrm{re}} = \tau] = \frac{1}{|\mathcal{K}_h|} \cdot \underbrace{\Pr[\pi \xleftarrow{\$} \mathsf{Perm} : \pi \in \mathsf{Comp}(\tau)]}_{\mathsf{P}_{mv}}. \tag{25}$$

Lower bounding $\mathsf{P}_{mv}$ implies lower bounding the probability of the number of solutions to the system consisting (i) $q_m$ many bivariate affine MAC equations, and (ii) $q_v$ many bivariate affine verification non-equations $\mathcal{E}_m \cup \mathcal{E}_v$. From the above system of bivariate affine equations and non-equations, one can induce an edge-labelled undirected graph $\mathsf{G}_\tau = (\mathcal{V}, \mathcal{E} \cup \mathcal{E}', \mathfrak{L})$, where the set of nodes $\mathcal{V}$ is the set of variables $\{Y_1, \ldots, Y_s\}$, $\mathcal{E}$ is the set of edges corresponding to each MAC equation and $\mathcal{E}'$ is the set of edges corresponding to each verification non-equation. Therefore, $q_m = |\mathcal{E}|$ and $q_v = |\mathcal{E}'|$. Moreover, if there is a MAC equation $Y_u \oplus Y_v = T_i$, then the corresponding edge $\{Y_u, Y_v\} \in \mathcal{E}$ is labelled as $T_i$. Similarly, if there is a verification non-equation $Y_u \oplus Y_v \neq T'_i$, then the corresponding edge $\{Y_u, Y_v\} \in \mathcal{E}'$ is labelled as $T'_i$. Moreover, $\mathsf{G}^{=}_\tau = (\mathcal{V}^{=}, \mathcal{E}, \mathfrak{L}_{|\mathcal{E}})$ is the subgraph of $\mathsf{G}_\tau$. Now, it is easy to argue the following.

**Claim 1.** *For a good transcript $\tau$, the induced graph $\mathsf{G}_\tau$ is a good graph.*

**Proof.** For a good transcript $\tau$, as the component size of $\mathsf{G}^{=}_\tau$ is at most 4, it is to be noted that the subgraph $G^{=}_\tau$ contains only three types of components as follows in Fig. 6.1.

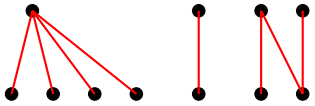

**Figure 6.1:** (a) star type component, (b) the component with a single edge and (c) component with a path of length three

(a) for the first type of component, if we consider any path, then the label of the path is non-zero; otherwise, either one of the bad conditions $\mathsf{B}.6$ or $\mathsf{B}.7$ would have been satisfied. (b) For the second type of component, the label of the edge is non-zero; otherwise, bad condition $\mathsf{B}.8$ would have been satisfied. Finally, if we consider the path of the third type of component, then the label of the path is non-zero due to $\overline{\mathsf{B}.9}$. Moreover, the graph is acyclic due to condition $\overline{\mathsf{B}.4}$ and $\overline{\mathsf{B}.5}$, which proves the claim. □

Resuming the proof of Lemma 8. Suppose there are $\alpha + \beta$ components in the subgraph $\mathsf{G}^{=}_\tau$ such that the size of each of the first $\alpha$ components is greater than two and the remaining $\beta$ components are of size two each. As the transcript $\tau$ is good, the total number of edges in the first $\alpha$ components is at most $q_m^{2/3}$. Now applying Theorem 1, we obtain

$$\mathsf{P}_{mv} \geq \frac{1}{2^{nq_m}} \left(1 - \frac{9q_m^{4/3}}{4 \cdot 2^n} - \frac{9q_m^{7/3} + 24q_m^{8/3} + 6q_m^{5/3} + 40q_m^2}{2^{2n}} - \frac{16q_m^4}{2^{3n}} - \frac{7q_v}{2^n}\right). \tag{26}$$

Therefore, from Eqn. (25) and Eqn. (26), we have

$$\mathsf{p}_{\mathrm{re}}(\tau) \geq \frac{1}{|\mathcal{K}_h|} \cdot \frac{1}{2^{nq_m}} \cdot \left(1 - \frac{9q_m^{4/3}}{4 \cdot 2^n} - \frac{9q_m^{7/3} + 24q_m^{8/3} + 6q_m^{5/3} + 40q_m^2}{2^{2n}} - \frac{16q_m^4}{2^{3n}} - \frac{7q_v}{2^n}\right). \quad (27)$$

Finally, by taking the ratio of Eqn. (27) to Eqn. (24), we obtain the result. □

Note that the hash key is not derived using $\pi$, it is sampled independent to the block cipher keys.

# 7 Proof of Lemma 7

Using the union bound, we write

$$\Pr[X_{\mathrm{id}} \in \Theta_b] \leq \sum_{\mathsf{v} \in \{1,2,3,6,7,8,9\}} \Pr[\mathsf{B.v}] + \Pr[\mathsf{B.4} \mid \overline{\mathsf{B.2}}] + \Pr[\mathsf{B.5} \mid \overline{\mathsf{B.1}} \wedge \overline{\mathsf{B.2}} \wedge \overline{\mathsf{B.4}}]. \quad (28)$$

In the following, we bound the probabilities of all the bad events individually.

**Bounding B.1.** Event B.1 occurs if there exists a MAC query whose response is all zero. For a fixed MAC query, the probability of this event in the ideal world is $2^{-n}$ as the responses are sampled uniformly and independently to all other sampled random variables. Now, varying over all such MAC queries, we obtain the bound to be

$$\Pr[\mathsf{B.1}] \leq \frac{q_m}{2^n} \quad (29)$$

**Bounding B.2.** Event B.2 occurs if there exists a component of size at least 5 in $\mathsf{G}^{=}$. Depending on the collision pattern of the vertices, we have the following cases, each of which is analyzed one by one as follows:

Bounding Case-I. $\exists i, j, k, l \in [q_m]$ such that $T_i = T_j = T_k = T_l$. For a fixed set of $i, j, k, l \in [q_m]$, this event is bounded by $2^{-3n}$ as each $T_i$ is sampled uniformly at random from $\{0,1\}^n$. Summing over all possible choices of $i, j$ and $k$, we obtain the bound $\frac{q_m^4}{24 \cdot 2^{3n}} \leq \frac{q_m}{2^{3n/4}}$, assuming $q_m \leq 2^{3n/4}$.

Bounding Case-II. $\exists i, j, k, l \in [q_m]$ such that $T_i = T_j = T_k = \nu_l$ or $T_i = T_j = T_k, \nu_k = T_l$. For a fixed set of $i, j, k$ $T_i = T_j = T_k$ is bounded by $2^{-2n}$. Now, we have the following two subcases for $\nu_l = T_i = T_j = T_k$:

- Case (a): If $l < i, j, k$, then the probability of $\nu_l = T_i$ is bounded by $2^{-n}$. Thus the overall probability becomes $2^{-3n}$. Summing over all possible choices of $i, j, k, l$, we obtain $\frac{q_m^4}{24 \cdot 2^{3n}}$.
- Case (b): Otherwise, without loss of generality, we assume that $l > i$. In that case, the probability that $\nu_l$ can be set to $T_i$ is the probability that $T_i$ is a valid nonce. Applying Lemma 6, the overall probability is $2^{-9n/4}$, and if we sum over all possible choices of $i, j, k$, we obtain the bound $\frac{q_m^3}{6 \cdot 2^{9n/4}} \leq \frac{q_m}{2^{3n/4}}$.

For the other case, i.e., $T_i = T_j = T_k, \nu_k = T_l$, we have the following two subcases:

- Case (a): If $l < k$, then due to Lemma 6, the probability of $\nu_k = T_l$ is bounded by $2^{-n/4}$, and thus the overall probability becomes $2^{-9n/4}$. The number of choices for each of $i, j, l$ is $q_m$, and the number of choices for $k = 1$. Summing over all possible choices of $i, j, k, l$, we obtain $\frac{q_m^3}{6 \cdot 2^{9n/4}}$.

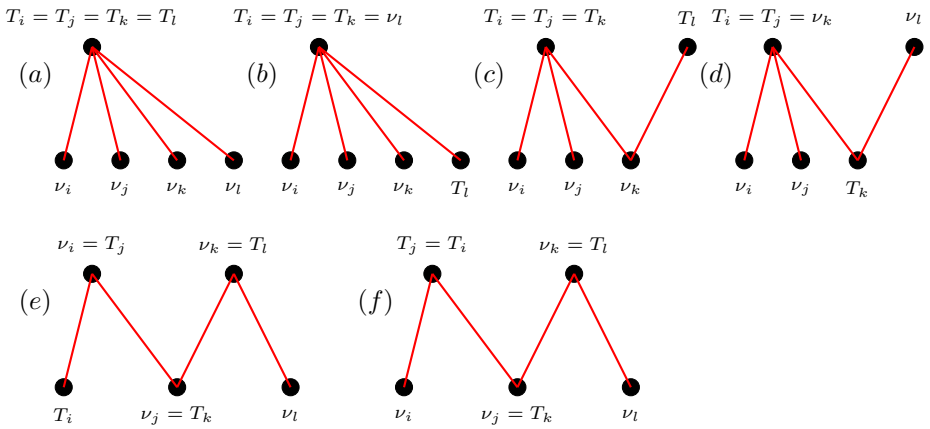


**Figure 7.1:** Different components of size of five. (a) $T_i = T_j = T_k = T_l$, (b) $T_i = T_j = T_k = \nu_l$, (c) $T_i = T_j = T_k, \nu_k = T_l$, (d) $T_i = T_j = \nu_k$ and $T_k = \nu_l$, (e) $\nu_i = T_j, \nu_j = T_k, \nu_k = T_l$, and (f) $T_i = T_j, \nu_j = T_k, \nu_k = T_l$.

- Case (b): If $l > k$, then the probability of the event $\nu_k = T_l$ is $2^{-n}$, and therefore, the overall probability becomes $2^{-3n}$. Summing over all possible choices of $i, j, k, l$, we have $\frac{q_m^4}{24 \cdot 2^{3n}}$.

Bounding Case-III. $\exists i, j, k \in [q_m]$ such that $T_i = T_j = \nu_k$ and $T_k = \nu_l$.

- Case (a): If $l < k < i, j$, then the probability of $\nu_k = T_i$ is bounded by $2^{-n}$, and thus the overall probability becomes $2^{-3n}$. Summing over all possible choices of $i, j, k, l$, we obtain the bound $\frac{q_m^4}{2^{3n}}$.
- Case (b): If $l < k$, $k > i$ or $k > j$, then without loss of generality, we assume that $k > i$, and in that case the probability that $\nu_k$ is set to $T_i$ is the probability that $T_i$ is a valid nonce, and applying Lemma 6, the overall probability becomes $2^{-9n/4}$. Summing over all possible choices of $j, k, l$, we have $\frac{q_m^3}{2^{9n/4}} \leq \frac{q_m}{2^{3n/4}}$.
- Case (c): If $l > k$ and $k < i, j$, then the probability of $\nu_k = T_i$ is bounded by $2^{-n}$, $\nu_l$ is set to $T_k$ is the probability that $T_k$ is a valid nonce, and thus due to Lemma 6, the overall probability is $2^{-9n/4}$. Summing over all possible choices of $i, j, k, l$, we obtain the bound $\frac{q_m^3}{2^{9n/4}}$.
- Case (d): If $l > k$, $k > i$ or $k > j$, then without loss of generality, we assume that $k > i$, and in that case the probability that $\nu_k$ is set to $T_i$ is the probability that $T_i$ is a valid nonce, and applying Lemma 6, the overall probability becomes $2^{-3n/2}$. Summing over all possible choices of $j, k, l$, we have $\frac{q_m^2}{2^{3n/2}} \leq \frac{q_m}{2^{3n/4}}$.

Bounding Case-IV. $\exists i, j, k, l \in [q_m]$ such that $\nu_i = T_j, \nu_j = T_k, \nu_k = T_l$. We bound this event using different subcases.

- Case (a): If $i < j < k < l$, then due to Lemma 6, we obtain $2^{-3n}$ bound, and varying over all possible choices of $i, j, k, l$, we obtain $\frac{q_m^4}{2^{3n}}$ bound.
- Case(b): If $i < j < k$ and $k > l$, then due to Lemma 6, we obtain $2^{-9n/4}$ bound, but there is exactly one choice of $k$ and $q_m$ many choices for $i, j, l$. Hence, by summing over all possible choices of $i, j, k, l$, we have $\frac{q_m^3}{2^{9n/4}} \leq \frac{q_m}{2^{3n/4}}$.

- Case (c): If $i < j$ and $j > k > l$, then due to Lemma 6, we obtain $2^{-3n/2}$ bound, but there is exactly one choice of $k, l$ and $q_m$ many choices for $i, j$. Hence, by summing over all possible choices of $i, j, k, l$, we obtain $\frac{q_m^2}{2^{3n/2}} \leq \frac{q_m}{2^{3n/4}}$ bound.
- Case (d): If $i > j > k > l$, then due to Lemma 6, the probability of the event is bounded by $2^{3n/4}$, and there is exactly one choice of $i, j, k$, leaving $q_m$ choices for $l$, which eventually gives $\frac{q_m}{2^{3n/4}}$ bound.

Bounding Case-V. $\exists i, j, k, l \in [q_m]$ such that $T_i = T_j, \nu_j = T_k, \nu_k = T_l$. For a fixed set of $i, j$ $T_i = T_j$ is bounded by $2^{-n}$. Now, we have the following three subcases for $\nu_j = T_k,\ \nu_k = T_l$:

- Case (a): If $l < k < j$, then the probability of both $\nu_j = T_k$, and $\nu_k = T_l$ are bounded by the probability that $T_k$ and $T_l$ are valid nonce respectively, both of which is equal to $2^{-n/4}$. Thus, the overall probability becomes $2^{-3n/2}$, and summing over all possible choices of $i, j, k, l$, we obtain $\frac{q_m^2}{2^{3n/2}}$.
- Case (b): If $l > k > j$, then the probability of both $\nu_j = T_k$, and $\nu_k = T_l$ is bounded by $2^{-n}$, and thus the overall probability becomes $2^{-3n}$. Summing over all possible choices of $i, j, k, l$, we obtain $\frac{q_m^4}{2^{3n}}$.
- Case (c): If $l, j > k$ or $l, j < k$, then with similar argument as above, the overall probability becomes $2^{-9n/4}$, and the number of choices for $i, j, k, l$ is $q_m^3$, and hence we obtain $\frac{q_m^3}{2^{9n/4}}$.

For each of the above cases, we obtain the maximum bound to be $\frac{q_m}{2^{3n/4}}$, and therefore, we have

$$\Pr[\mathsf{B}.2] \leq \frac{q_m}{2^{3n/4}}. \tag{30}$$

**Bounding B.3:** Let $X$ denotes the cardinality of the set $\{i \neq j \in [q_m] : \nu_i = T_j \vee T_i = T_j\}$ and for all $i \neq j \in [q_m]$, let $\mathbb{I}_{ij}$ be the indicator random variable that takes the value 1 if $\nu_i = T_j \vee T_i = T_j$, otherwise it takes the value 0. Therefore,

$$X = \sum_{i \neq j} \mathbb{I}_{ij}.$$

By using the linearity of expectation, we have

$$\mathbf{E}[X] = \sum_{i \neq j} \mathbf{E}[\mathbb{I}_{ij}] = \sum_{i \neq j} 1.\Pr[\mathbb{I}_{ij} = 1] = \sum_{i \neq j} 1.\Pr[\nu_i = T_j \vee T_i = T_j] \leq \frac{q_m}{2^{n/4}} + \frac{q_m^2}{2^n}. \tag{31}$$

By applying Markov's inequality on Eqn. (31), we have

$$\Pr[\mathsf{B}.3] = \Pr[X \leq q_m^{2/3}] \leq \mathbf{E}[X]/q_m^{2/3} \leq q_m^{1/3}/2^{n/4} + q_m^{4/3}/2^n. \tag{32}$$

**Bounding B.4 |$\overline{\mathsf{B}.2}$:** Recall that event $\mathsf{B}.4$ holds if there exists any cycle in $\mathsf{G}^{=}$. But, as we conditioned on $\overline{\mathsf{B}.2}$, it is enough to bound the existence of a cycle of length one (self loop), two (parallel edges), three (triangle), or four (square).

Bounding Self Loop. A self loop or a cycle of length 1 in $\mathsf{G}^{=}$ implies that $\exists i \in [q_m]$ such that $\nu_i = T_i$. For a fixed choice of $i$, the probability of $\nu_i = T_i$ is bounded by $2^{-n}$ due to the randomness of $T_i$. Summing over all choices of $i$, we obtain $\frac{q_m}{2^n}$ bound.

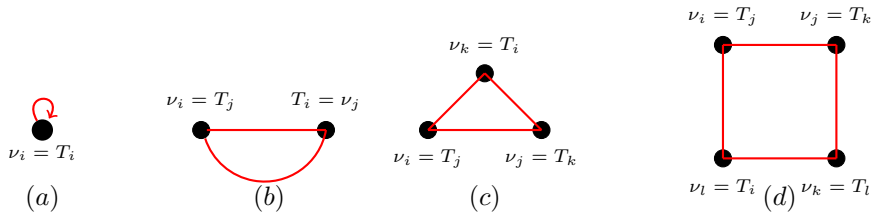


**Figure 7.2:** (a) Self Loop: when $\nu_i = T_i$, (b) Parallel Edges: $\nu_i = T_j, \nu_j = T_i$, (c) Triangle: $\nu_i = T_j, \nu_j = T_k, \nu_k = T_i$, (d) Square: $\nu_i = T_j, \nu_j = T_k, \nu_k = T_l, \nu_l = T_i$.

Bounding Parallel Edges. A parallel edge or a cycle of length 2 in $\mathsf{G}^{=}$ implies that $\exists i \neq j \in [q_m]$ such that $\nu_i = T_j, \nu_j = T_i$. For a fixed choice of $i, j$ (w.l.o.g, assume $i < j$), the probability of $\nu_i = T_j, \nu_j = T_i$ is bounded by $2^{-5n/4}$. This is because of Lemma 6, the probability of $\nu_i = T_j$ is bounded by $2^{-n}$ and the probability of $\nu_j = T_i$ is bounded by $2^{-n/4}$. As there exists only one choice of $j$ and $q_m$ many choices of $i$, summing over all possible choices of $i$ and $j$, we obtain $\frac{q_m}{2^{5n/4}}$ bound.

Bounding Triangle. A triangle in $\mathsf{G}^{=}$ implies that $\exists i \neq j \neq k \in [q_m]$ such that $\nu_i = T_j, \nu_j = T_k, \nu_k = T_i$. If $i < j < k$, the probability of $\nu_i = T_j, \nu_j = T_k, \nu_k = T_i$ is bounded by $2^{-9n/4}$. Note that, due to Lemma 6, the probability of $\nu_i = T_j$, and $\nu_j = T_k$ can be bounded by $2^{-n}$ each, and the probability of $\nu_k = T_i$ is bounded by $2^{-n/4}$. As there exists only one choice of $k$ and $q_m$ many choices for $i, j$, summing over all possible choices of $i$ and $j$, we obtain $\frac{q_m^2}{2^{9n/4}}$ bound. On the other hand, if $i > j > k$, the probability of $\nu_i = T_j, \nu_j = T_k, \nu_k = T_i$ is bounded by $2^{-3n/2}$, and the number of choices for each of $i, j$ is 1, and the choice of $k$ is $q_m$, and hence we obtain a bound of $\frac{q_m}{2^{3n/2}}$. All the other ordering of $i, j, k$ leads to similar analysis as done in the above two cases, and hence this bad case can be bounded by $\frac{q_m}{2^{3n/2}}$.

Note that any other way of forming the triangle involving only the MAC queries immediately implies that the triangle either contains a self loop or contains a parallel edge. Since these two events (i.e., self loop and parallel edge) have already bounded, we have deliberately skipped the analysis for these cases.

Bounding Square. A square in $\mathsf{G}^{=}$ implies that $\exists i \neq j \neq k \neq l \in [q_m]$ such that $\nu_i = T_j, \nu_j = T_k, \nu_k = T_l, \nu_l = T_i$. If $i < j < k < l$, the probability of $\nu_i = T_j, \nu_j = T_k, \nu_k = T_l, \nu_l = T_i$ is bounded by $2^{-13n/4}$. This is because of Lemma 6, the probability of $\nu_i = T_j, \nu_j = T_k$, and $\nu_k = T_l$ are bounded by $2^{-n}$ each, and the probability of $\nu_l = T_i$ is bounded by $2^{-n/4}$. As there exists only one choice of $l$ and $q_m$ many choices for $i, j, k$, summing over all possible choices of $i, j$ and $k$, we obtain $\frac{q_m^3}{2^{13n/4}}$ bound. On the other hand, if $i > j > k > l$, the probability of the event $\nu_i = T_j, \nu_j = T_k, \nu_k = T_l, \nu_l = T_i$ is bounded by $2^{-7n/4}$. Moreover, the number of choices for each of the $i, j, k$ is 1, and the number of choice for $l$ is $q_m$. Hence, we obtain a bound of $\frac{q_m}{2^{7n/4}}$. All the other ordering of $i, j, k$ leads to similar analysis as done in the above two cases, and hence this bad case can be bounded by $\frac{q_m}{2^{7n/4}}$.

Note that any other way of forming the square involving only the MAC queries immediately implies that the square either contains a self loop or contains a parallel edge, or contains a triangle. Since these events have already been bounded, we have deliberately skipped these cases from our analysis.

Therefore, from the above four cases, we obtain the maximum bound to be $\frac{q_m}{2^n}$, and thus we write

$$\Pr[\mathsf{B.4} \mid \overline{\mathsf{B.2}}] \leq \frac{q_m}{2^n}. \tag{33}$$

**Bounding B.5 | $\overline{\mathsf{B.1}} \wedge \overline{\mathsf{B.2}} \wedge \overline{\mathsf{B.4}}$.** Recall that event B3 holds if there exists a valid cycle $C^{\neq}$ in $\mathsf{G}^{=}$, which implies that the sum of the labels of the cycle is zero. However, as we conditioned on $\overline{\mathsf{B.1}} \wedge \overline{\mathsf{B.2}} \wedge \overline{\mathsf{B.4}}$, it is enough to bound the existence of a valid cycle of length one (self loop), two (parallel-edges), three (triangle) and four (square) involving a verification query.

Bounding Self-Loop. A self loop or cycle of length 1 in $\mathsf{G}^{=}$ implies that $\exists a \in [q_v]$ such that $\nu'_a = T'_a$ and $\mathsf{H}_{k_h}(M'_a) = \nu'_a$. Note that, for a fixed choice of $a$, the above event holds with probability at most $\epsilon_{\text{reg}}$, as we have assumed the hash function to be $\epsilon_{\text{reg}}$ regular. Summing over all choices of $a$, we obtain $q_v\epsilon_{\text{reg}}$ bound.

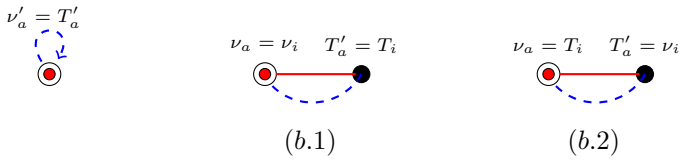


**Figure 7.3:** $(a)$ Self Loop: when $\nu'_a = T'_a$, $(b)$ Parallel Edges: $(b.1)$ $\nu'_a = \nu_i, T'_a = T_i$, $(b.2)$ $\nu'_a = T_i, T'_a = \nu_i$. Node with concentric circle denotes the verification query node.

Bounding Parallel Edges. A parallel edge or cycle of length 2 in $\mathsf{G}^{=}$ that involves a verification query implies either of the following two conditions: $\exists i \in [q_m], a \in [q_v]$ :

$$\begin{cases} \text{Case-I} : \nu_i = \nu'_a, T_i = T'_a, \mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M'_a) = \nu_i \oplus \nu'_a \\ \text{Case-II} : \nu'_a = T_i, T'_a = \nu_i, \mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M'_a) = \nu_i \oplus \nu'_a \end{cases}$$

- Bounding Case-I. For a fixed choice of $i$ and $a$, the probability of the event $\nu_i = \nu'_a, T_i = T'_a, \mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M'_a) = \nu_i \oplus \nu'_a$ is bounded by $\epsilon_{\text{axu}}$. This is due to the randomness of hash key $k_h$ (note that $M_i \neq M'_a$, as we have assumed a non-trivial distinguisher). As there exists only one choice of $i$ for which the above probability is bounded by $\epsilon_{\text{axu}}$, summing over all possible choices of $i$ and $a$, we obtain the bound $q_v\epsilon_{\text{axu}}$.

- Bounding Case-II. Similarly, for a fixed choice of $i, a$, the probability of $\nu'_a = T_i, T'_a = \nu_i, \mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M'_a) = \nu_i \oplus \nu'_a$ is bounded by $\epsilon_{\text{axu}}$ due to the randomness of hash key $k_h$, Note that, in this case there exists at most three choices of $i$ (as we have at most three collision of $T$) for which the above probability is bounded by $\epsilon_{\text{axu}}$. Summing over all possible choices of $i$ and $a$, we obtain the bound $3q_v\epsilon_{\text{axu}}$.

Thus, from the above two cases, the probability of forming a parallel edge can be bounded by $3q_v\epsilon_{\text{axu}}$.

Bounding Triangle. For the case of a triangle or cycle of length 3 in $\mathsf{G}^{=}$ that involves the valid cycle $C^{\neq}$ implies that $\exists i, j \in [q_m], a \in [q_v]$ such that either of the following holds:

$$\begin{cases} \text{Case-I} : \nu_i = T_j, \nu'_a = \nu_j, T'_a = T_i \text{ and} \\ \qquad\qquad \mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M_j) \oplus \mathsf{H}_{k_h}(M'_a) = T_j \\ \text{Case-II} : \nu_i = T_j, T'_a = \nu_j, \nu'_a = T_i \text{ and} \\ \qquad\qquad \mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M_j) \oplus \mathsf{H}_{k_h}(M'_a) = T_i \oplus T_j \oplus T'_a. \\ \text{Case-III} : T_i = T_j, \nu'_a = \nu_j, T'_a = \nu_i \text{ and} \\ \qquad\qquad \mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M_j) \oplus \mathsf{H}_{k_h}(M'_a) = T_i \end{cases}$$

We bound each of these events as follows:

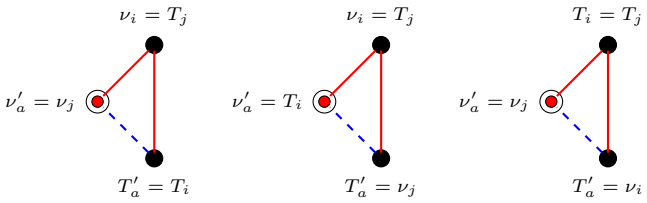


**Figure 7.4:** Cycles of length 3 including the verification query which is denoted by the concentric circle node.

- Bounding Case-I. For a fixed choice of $i, j, a$, the probability of the event is bounded by $\epsilon_{3\text{-reg}}$ using the randomness of the hash key as we have assumed the hash function is $\epsilon_{3\text{-reg}}$-3-way regular (note that we have conditioned on $\overline{\mathsf{B.1}}$ and therefore $T_j \neq \mathbf{0}$). Note that, the number of choice for $j$ and $i$ is restricted to one and three respectively. Hence, summing over all possible choices of indices, we obtain the bound to be $3q_v\epsilon_{3\text{-reg}}$.

- Bounding Case-II. We analyze this case in two different subcases:
  - (a) For a fixed choice of $i, j$ and $a$, if $\nu_j \neq T_i \oplus T_j$, then $\mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M_j) \oplus \mathsf{H}_{k_h}(M'_a) \neq \mathbf{0}$, and thus we can bound the event by $\epsilon_{3\text{-reg}}$. In this case, choices of $i$ and $j$ is 3 an 1 respectively, and hence we obtain the bound to be $3q_v\epsilon_{3\text{-reg}}$.
  - (b) For a fixed choice of $i, j$ and $a$, if $\nu_j = T_i \oplus T_j$, then $\mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M_j) \oplus \mathsf{H}_{k_h}(M'_a) = \mathbf{0}$, and in that case we again consider two different subcases: (i) if $i < j$, then $T_j = \nu_i$ holds with probability $2^{-n}$ and $T_i$ is to be valid (i.e. last $n/4$ bits of $T_i$ has to be zero), which holds with probability $2^{-n/4}$. Moreover, the number of choices of $i, j$ in this case are $q_m$ and 1 (as $\nu_j = \nu_i \oplus T_i$) resp. and thus we obtain the bound to be $\frac{q_m}{2^{5n/4}}$. (ii) If $i > j$, then $T_j = \nu_i$ holds with probability $2^{-n/4}$, and $T_i$ should be $\nu_i \oplus \nu_j$ which holds with probability $2^{-n}$. In this case, the number of choices of $j$ is $q_m$ and $i$ in 1, resulting in probability of the event to be bounded by $\frac{q_m}{2^{5n/4}}$.

- Bounding Case-III. For a fixed choice of $i, j$ and $a$, the probability of the event is bounded by $\epsilon_{3\text{-reg}}$ as $\mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M_j) \oplus \mathsf{H}_{k_h}(M'_a) = T_i$ holds with probability at most $\epsilon_{3\text{-reg}}$ (by the assumption that the hash function is $\epsilon_{3\text{-reg}}$-3-way regular). Note that, in this case choice of $j$ and $i$ is one. Hence, summing over all possible choices of indices, we obtain the bound to be $q_v\epsilon_{3\text{-reg}}$.

Therefore, we see that for all the above cases, the maximum probability of forming a closed triangle is $\max\{3q_v\epsilon_{3\text{-reg}}, \frac{q_m}{2^{5n/4}}\}$. Note that the other way of forming the valid cycle $C^{\neq}$ in $\mathsf{G}^{=}$ that involves the verification query immediately implies the existence of a self loop or parallel edges.

Bounding Square. For the case of a square or a valid cycle $C^{\neq}$ of length 4 in $\mathsf{G}^{=}$ implies that $\exists i, j, k \in [q_m], a \in [q_v]$ such that either of the following holds:

$$\begin{cases} \text{Case-I} : \nu'_a = \nu_i, T_i = \nu_j, T_j = \nu_k, T'_a = T_k \text{ and} \\ \qquad\qquad \mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M_j) \oplus \mathsf{H}_{k_h}(M_k) \oplus \mathsf{H}_{k_h}(M'_a) = T_i \oplus T_j \\ \text{Case-II} : \nu'_a = \nu_i, T_i = T_j, \nu_j = T_k, T'_a = \nu_k \text{ and} \\ \qquad\qquad \mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M_j) \oplus \mathsf{H}_{k_h}(M_k) \oplus \mathsf{H}_{k_h}(M'_a) = T_k \oplus T'_a \\ \text{Case-III} : \nu'_a = T_i, \nu_i = T_j, \nu_j = T_k, T'_a = \nu_k \text{ and} \\ \qquad\qquad \mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M_j) \oplus \mathsf{H}_{k_h}(M_k) \oplus \mathsf{H}_{k_h}(M'_a) = T_i \oplus T_j \oplus T_k \oplus T'_a \end{cases}$$

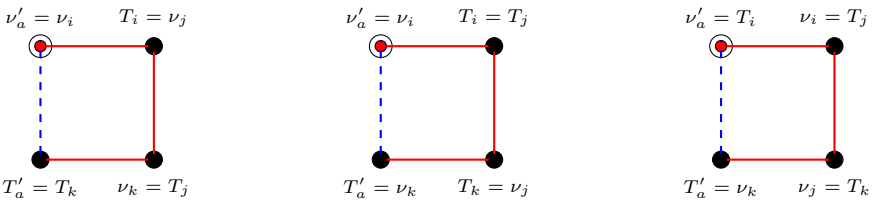


**Figure 7.5:** Cycles of length 4 including the verification query which is denoted by the concentric circle node.

We bound each of these events as follows.

- Bounding Case-I. Note that, $\mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M_j) \oplus \mathsf{H}_{k_h}(M_k) \oplus \mathsf{H}_{k_h}(M'_a) \neq \mathbf{0}$ as $T_i = T_j$ implies $\nu_j = \nu_k$, which is not possible, and thus we can bound the event by $q_v\epsilon_{4\text{-reg}}$, as the choices of $i$, $j$ and $k$ are one each.

- Bounding Case-II. Here also we have, $\mathsf{H}_{k_h}(M_i)\oplus\mathsf{H}_{k_h}(M_j)\oplus\mathsf{H}_{k_h}(M_k)\oplus\mathsf{H}_{k_h}(M'_a) \neq \mathbf{0}$ as $T_k = T'_a$ implies $\nu_k = T_k$ which essentially gives a self loop. Thus, we can bound the event by $q_v\epsilon_{4\text{-reg}}$, as the choices of $i$, $j$ and $k$ are one each.

- Bounding Case-III. We analyze this case in two different subcases depending on whether $\nu_k$ is equals to $T_i \oplus T_j \oplus T_k$ or not. For a fixed choice of $i, j$ and $a$, if $\nu_k \neq T_i \oplus T_j \oplus T_k$, then $\mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M_j) \oplus \mathsf{H}_{k_h}(M_k) \oplus \mathsf{H}_{k_h}(M'_a) \neq \mathbf{0}$, and thus we can bound the event by $\epsilon_{4\text{-reg}}$. In this case, choices of $i$, $j$ and $k$ is 1, 1, and 1 respectively, and hence we obtain the bound to be $q_v\epsilon_{4\text{-reg}}$.

  On the other hand, if $\nu_k = T_i \oplus T_j \oplus T_k$, then $\mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M_j) \oplus \mathsf{H}_{k_h}(M'_a) = \mathbf{0}$, and in that case we consider the following subcases:

  - Case (a): If $k > i, j$, then $T_k = \nu_j$ holds with probability $2^{-n}$. If $i < j$, then $T_j = \nu_i$ holds with probability $2^{-n}$, and the number of choices of $i, j, k$ in this case are $q_m$, $q_m$, and 1 (as $\nu_k = T_i \oplus \nu_i \oplus \nu_j$) resp., and thus we obtain the bound to be $\frac{q_m^2}{2^{2n}}$. On the other hand, if $i > j$, then $T_j = \nu_i$ holds with probability $2^{-n/4}$, and the choices for $i$ would become 1, obtaining a bound of $\frac{q_m}{2^{5n/4}}$.
  - Case (b): If $j > i, k$, then $T_j = \nu_i$ holds with probability $2^{-n}$, and $T_k = \nu_j$ holds with probability $2^{-n/4}$. The number of choices for $i, j, k$ are $q_m$, 1 and 1, and hence resulting in probability of the event to be bounded by $\frac{q_m}{2^{5n/4}}$.
  - Case (c): If $i > j, k$, then if $j > k$, both $T_j = \nu_i$ and $T_k = \nu_j$ holds with probability $2^{-n/4}$, and $T_i = \nu_i \oplus \nu_j \oplus \nu_k$ holds with probability $2^{-n}$, hence bounding the overall probability by $\frac{q_m}{2^{3n/2}}$. If $k > j$, $T_k = \nu_j$ holds with probability $\frac{1}{2^n}$, while the choices for $j$ can be at most $q_m$. Hence, the probability can be bounded by $\frac{q_m^2}{2^{9n/4}}$.

Therefore, we see that for all the above cases, the maximum probability of forming a closed square is $\max\{3q_v\epsilon_{4\text{-reg}}, \frac{q_m}{2^{3n/4}}\}$. Therefore, we see from all of the above cases the maximum probability of forming a valid cycle $C^{\neq}$ in $\mathsf{G}^{=}$ is

$$\max\{3q_v\epsilon_{4\text{-reg}}, 3q_v\epsilon_{3\text{-reg}}, 3q_v\epsilon_{\text{axu}}, q_v\epsilon_{\text{reg}}, \frac{q_m}{2^{3n/4}}\}.$$

Thus, we have

$$\Pr[\mathsf{B}.5 \mid \overline{\mathsf{B}.1} \wedge \overline{\mathsf{B}.2} \wedge \overline{\mathsf{B}.4}] \leq \max\{3q_v\epsilon_{4\text{-reg}}, 3q_v\epsilon_{3\text{-reg}}, 3q_v\epsilon_{\text{axu}}, q_v\epsilon_{\text{reg}}, \frac{q_m}{2^{3n/4}}\}. \tag{34}$$

**Bounding B.6:** Recall that, the event B.6 holds if $\exists i \neq j \in [q_m]$ such that $\nu_i \oplus \mathsf{H}_{k_h}(M_i) = \nu_j \oplus \mathsf{H}_{k_h}(M_j), T_i = T_j$. Since, in the ideal oracle the hash key is sampled independent to all previously sampled MAC responses $T_i$, we deduce

$$\Pr[\mathsf{B.6}] \leq \sum_{i,j} \Pr[\mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M_j) = \nu_i \oplus \nu_j] \cdot \Pr[T_i = T_j] \leq \frac{q_m^2 \cdot \epsilon_{\text{axu}}}{2^n}. \tag{35}$$

**Bounding B.7:** Recall that, the event B.7 holds if $\exists i \neq j \in [q_m]$ such that $\nu_i \oplus \mathsf{H}_{k_h}(M_i) = \nu_j \oplus \mathsf{H}_{k_h}(M_j), \nu_i = T_j$. Now, we consider two subcases:

- (a) For fixed $i$ and $j$, if $i < j$ then $\nu_i = T_j$ holds with probability $2^{-n}$ (due to Lemma 6), and $\nu_i \oplus \mathsf{H}_{k_h}(M_i) = \nu_j \oplus \mathsf{H}_{k_h}(M_j)$ holds with probability $\epsilon_{\text{axu}}$. Summing over all possible choices of $i$ and $j$, we obtain the bound to be $\frac{q_m^2 \epsilon_{\text{axu}}}{2^n}$.
- (b) When $i > j$, then $\nu_i = T_j$ holds with probability $2^{-n/4}$ (due to Lemma 6), and as before $\nu_i \oplus \mathsf{H}_{k_h}(M_i) = \nu_j \oplus \mathsf{H}_{k_h}(M_j)$ holds with probability $\epsilon_{\text{axu}}$. In this case, possible choices of $i$ and $j$ is 1 and $q_m$ respectively and therefore by summing over all possible choices of indices, we obtain the bound to be $\frac{q_m \epsilon_{\text{axu}}}{2^{n/4}}$.

Therefore, from each of the above cases we have

$$\Pr[\mathsf{B.7}] \leq \frac{q_m^2 \epsilon_{\text{axu}}}{2^n} + \frac{q_m \epsilon_{\text{axu}}}{2^{3n/4}}. \tag{36}$$

**Bounding B.8:** Recall that the event B.8 holds if $\exists i \in [q_m]$ such that $\nu_i = \mathsf{H}_{k_h}(M_i)$. For a fixed $i \in [q_m]$, the event holds with probability $\epsilon_{\text{reg}}$ due to the regular property of the hash function. Summing over all choices of $i$, we obtain the bound

$$\Pr[\mathsf{B.8}] \leq q_m \epsilon_{\text{reg}}. \tag{37}$$

**Bounding B.9:** The event B.9 occurs if there exists a path of length three in $\mathsf{G}^{=}$ such that the label of the path is zero. Depending on the collision pattern of the vertices, we have the following cases, each of which is analyzed one by one as follows:

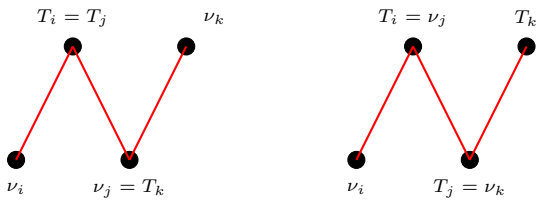


**Figure 7.6:** Different paths of length 3. (a) $T_i = T_j$ and $\nu_j = T_k$, (b) $\nu_j = T_i$ and $\nu_k = T_j$.

Bounding Case-I. $\exists i, j, k \in [q_m]$ such that $T_i = T_j, \nu_j = T_k$. For a fixed set of $i, j \in [q_m]$, the probability of the event $T_i = T_j$ can be bounded by $2^{-n}$ as each $T_i$ is sampled uniformly at random from $\{0,1\}^n$. Now, we first consider the case $\nu_i \oplus \nu_j \oplus \nu_k \neq \mathbf{0}$. In this case, we can bound the probability of the event $\mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M_j) \oplus \mathsf{H}_{k_h}(M_k) = \nu_i \oplus \nu_j \oplus \nu_k$ by $\epsilon_{3\text{-reg}}$ property of the hash function. Now we have the following subcases:

- Case (a): If $k < j$, the probability of the event $\nu_j = T_k$ is bounded by $2^{-n/4}$. Moreover, the total number of choices for $i, j, k$ is at most $q_m^2$. Therefore, summing over all possible choices of $i, j$ and $k$, we obtain the bound $\max\{\frac{q_m^2}{2^{5n/4}} \cdot \epsilon_{3\text{-reg}}$.

- Case (b): If $k > j$, the probability of the event $T_k = \nu_j$ is bounded by $2^{-n}$. The total number of choices for $i, j, k$ is at most $q_m^3$, and hence we obtain the bound $\frac{q_m^3}{2^{2n}} \cdot \epsilon_{3\text{-reg}}$.

On the other hand, if $\nu_i \oplus \nu_j \oplus \nu_k = \mathbf{0}$, then we have the following equalities: $T_i = T_j$, $T_k = \nu_i \oplus \nu_k$, $\nu_i \oplus \nu_k \oplus \nu_j = \mathbf{0}$. Note that, irrespective of the ordering of $i, j, k$, the first two equalities hold with probability $2^{-2n}$, and the choice of $i, j, k$ is at most $q_m^2$. Therefore, we obtain the bound $\frac{q_m^2}{2^{2n}}$.

Bounding Case-II. $\exists i, j, k \in [q_m]$ such that $\nu_j = T_i, \nu_k = T_j$. First, we consider the case where $\nu_i \oplus \nu_j \oplus \nu_k \neq \mathbf{0}$, and the event $\mathsf{H}_{k_h}(M_i) \oplus \mathsf{H}_{k_h}(M_j) \oplus \mathsf{H}_{k_h}(M_k) = \nu_i \oplus \nu_j \oplus \nu_k$ is bounded by $\epsilon_{3\text{-reg}}$ property of the hash function. Now, we consider the two subcases:

- Case (a): If $k < j < i$, the probability of both the events $\nu_j = T_i$, and $\nu_k = T_j$ can be bounded by $2^{-n/4}$. As the total number of choices for $i, j, k$ is at most $q_m$, we obtain the bound $\frac{q_m}{2^{n/2}} \cdot \epsilon_{3\text{-reg}}$.
- Case (b): If $i > j > k$, the probability of both the events $\nu_j = T_i$ and $\nu_k = T_j$ is bounded by $2^{-n}$. Summing over the total number of choices for $i, j, k$, which is at most $q_m^3$, we obtain the bound $\frac{q_m^3}{2^{2n}} \cdot \epsilon_{3\text{-reg}}$.
- Case (c): In all other subcases, we have $i < j$ or $j < k$, but not both. Hence, the probability of exactly one of the events $\nu_j = T_i$ and $\nu_k = T_j$ are bounded by $2^{-n}$, and the other by $2^{-n/4}$. As the choice of $j$ or $k$ is fixed, we obtain the bound $\frac{q_m^2}{2^{5n/4}} \cdot \epsilon_{3\text{-reg}}$.

Now, observe that if $\nu_i \oplus \nu_j \oplus \nu_k = \mathbf{0}$, then we obtain the sub event $T_i = \nu_j, \nu_i = T_i \oplus T_j$, which can be bounded by $\frac{q_m}{2^{5n/4}}$, if $i < j$ and $\frac{q_m^2}{2^{2n}}$, if $i > j$. Now, summing up all the subcases, we obtain

$$\Pr[\mathsf{B}.9] \leq \left(\frac{q_m^3}{2^{2n}} + \frac{q_m^2}{2^{5n/4}} + \frac{q_m}{2^{n/2}}\right) \cdot \epsilon_{3\text{-reg}} + \left(\frac{q_m^2}{2^{2n}} + \frac{q_m}{2^{5n/4}}\right). \tag{38}$$

Finally, Lemma 7 follows from Eqn. (28)-(38). □

# 8 How Our Proof Differs from the Original DWCDM?

Here we briefly mention how our proof differs from the one used in [DDNY18]. In the proof of DWCDM, authors cast a system of bivariate affine equations in a graph-theoretic setup, where each vertex represents an equation. Two vertices are joined with an edge if their corresponding equations share at least one variable. In this setup, under the H-Coefficient technique [Pat08a], authors defined a transcript to be *bad* if

1. the induced graph from the transcript contains a cycle,
2. or any of the components of the induced graph contains a path of length two or more.

They also rejected the transcript if any of its verification queries form a cycle with some MAC queries of the transcript. However, in our setup, we represent a system of bivariate affine equations in the form of a graph, where each vertex of the graph represents a variable of the equations. Two vertices are joined with an edge if their corresponding variables are involved in an equation. We refer to this representation as a *dual representation* of [DDNY18]. Note that that if we represent the system of equations of [DDNY18] in our graph-theoretic setting, then in the proof of DWCDM, a transcript would have been defined to be bad if

1. the induced graph from the transcript contains a cycle,
2. or any of the components of the induced graph contains *a path of length* 3 *or more.*

For example, consider the following system of equations

$$\mathcal{E}_m = \begin{cases} (i). \;\; \pi(\nu_1) \oplus \pi(T_1) = \lambda_1 \\ (ii). \;\; \pi(\nu_2) \oplus \pi(T_2) = \lambda_2 \\ (iii). \;\; \pi(\nu_3) \oplus \pi(T_3) = \lambda_3 \end{cases}$$

with the equality that $\nu_1 = T_2$ and $T_1 = T_3$. Now, if we represent the above system of equations in the form of a graph as defined in [DDNY18], then it would result in a graph $(a)$ as depicted in Fig. 8, where the path length is 2. As the number of equations is 3, the graph in $(a)$ contains three nodes. An edge joins node $(i)$ with node $(ii)$ as the equation $(i)$ and $(ii)$ has a common variable $\pi(\nu_1) = \pi(T_2)$. Similarly, node $(i)$ is joined by an edge with node $(iii)$ because the equation $(i)$ and $(iii)$ has a common variable $\pi(T_1) = \pi(T_3)$. However, if we represent $\mathcal{E}_m$ in our graph-theoretic setup, then that would result in graph $(b)$ as depicted in Fig. 8, where the path length is three.


**Figure 8.1:** (a) Graph representation of $\mathcal{E}_m$ as defined in [DDNY18]. (b) Graph representation of $\mathcal{E}_m$ as defined in this paper.

In order to improve the security bound of DWCDM with $3n/4$ bit nonce, we allow the transcripts whose induced graph contains a component having path length of at most 3. In other words, we reject all the transcripts that result in a graph with components of path length four or more. As before, we also avoid the presence of any cycles in the components. We also reject the transcript if any verification query forms a cycle with some MAC queries of the transcript. These restrictions immediately lead us to have one extra level of assumption on the hash function, which says that it is not sufficient to have only the 3-way regular property of the hash function, but it should be 4-way regular as well. Now, it is natural to wonder whether one can get $kn/k+1$ bit security with $kn/k+1$ bit nonce by allowing the components to have a path of length at most $k$ and rejecting the transcripts that induce components of path length $k+1$ or more. In fact, this result was stated as a conjecture in [DDNY18]. However, the bottleneck of proving this result is the good transcript analysis which stands on the availability of verifiable proof for Theorem $P_i \oplus P_j$ for any $\xi_{\max}$ result.

## References

[BDLN20] Arghya Bhattacharjee, Avijit Dutta, Eik List, and Mridul Nandi. Cencpp* - beyond-birthday-secure encryption from public permutations. Cryptology ePrint Archive, Report 2020/602, 2020. https://eprint.iacr.org/2020/602.

[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.

[Bra82] Gilles Brassard. On computationally secure authentication tags requiring short secret shared keys. In David Chaum, Ronald L. Rivest, and Alan T. Sherman, editors, *Advances in Cryptology: Proceedings of CRYPTO '82, Santa Barbara, California, USA, August 23-25, 1982*, pages 79–86. Plenum Press, New York, 1982.

[CLLL20] Wonseok Choi, ByeongHak Lee, Yeongmin Lee, and Jooyoung Lee. Improved security analysis for nonce-based enhanced hash-then-mask macs. In Shiho Moriai and Huaxiong Wang, editors, *Advances in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7-11, 2020, Proceedings, Part I*, volume 12491 of *Lecture Notes in Computer Science*, pages 697–723. Springer, 2020.

[CP20] Benoît Cogliati and Jacques Patarin. Mirror theory: A simple proof of the pi+pj theorem with xi_max=2. *IACR Cryptol. ePrint Arch.*, 2020:734, 2020.

[CS14] Shan Chen and John P. Steinberger. Tight security bounds for key-alternating ciphers. In *Advances in Cryptology - EUROCRYPT 2014,*, pages 327–350, 2014.

[CS16] Benoît Cogliati and Yannick Seurin. EWCDM: an efficient, beyond-birthday secure, nonce-misuse resistant MAC. In *CRYPTO 2016, Proceedings, Part I*, pages 121–149, 2016.

[DDNY18] Nilanjan Datta, Avijit Dutta, Mridul Nandi, and Kan Yasuda. Encrypt or decrypt? to make a single-key beyond birthday secure nonce-based MAC. In *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I*, pages 631–661, 2018.

[DDNY19] Nilanjan Datta, Avijit Dutta, Mridul Nandi, and Kan Yasuda. Dwcdm+: A BBB secure nonce based MAC. *Adv. Math. Commun.*, 13(4):705–732, 2019.

[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.

[DJN17] Avijit Dutta, Ashwin Jha, and Mridul Nandi. Tight security analysis of ehtm MAC. *IACR Trans. Symmetric Cryptol.*, 2017(3):130–150, 2017.

[DNS20] Avijit Dutta, Mridul Nandi, and Abishanka Saha. Proof of mirror theory for $\xi_{\max}=2$. *IACR Cryptol. ePrint Arch.*, 2020:669, 2020.

[DNT19] Avijit Dutta, Mridul Nandi, and Suprita Talnikar. Beyond birthday bound secure MAC in faulty nonce model. In *Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part I*, pages 437–466, 2019.

[IMV16] Tetsu Iwata, Bart Mennink, and Damian Vizár. CENC is optimally secure. *IACR Cryptology ePrint Archive*, 2016:1087, 2016.

[JN20] Ashwin Jha and Mridul Nandi. Tight security of cascaded LRW2. *J. Cryptol.*, 33(3):1272–1317, 2020.

[KLL20] Seongkwang Kim, ByeongHak Lee, and Jooyoung Lee. Tight security bounds for double-block hash-then-sum macs. In Anne Canteaut and Yuval Ishai, editors, *Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part I*, volume 12105 of *Lecture Notes in Computer Science*, pages 435–465. Springer, 2020.

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

[MI11] Kazuhiko Minematsu and Tetsu Iwata. Building blockcipher from tweakable blockcipher: Extending FSE 2009 proposal. In *Cryptography and Coding - 13th IMA International Conference, IMACC 2011, Oxford, UK, December 12-15, 2011. Proceedings*, pages 391–412, 2011.

[ML19] Alexander Moch and Eik List. Parallelizable macs based on the sum of prps with security beyond the birthday bound. In *Applied Cryptography and Network Security - 17th International Conference, ACNS 2019, Bogota, Colombia, June 5-7, 2019, Proceedings*, pages 131–151, 2019.

[MN17] Bart Mennink and Samuel Neves. Encrypted davies-meyer and its dual: Towards optimal security using mirror theory. In *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III*, pages 556–583, 2017.

[NPV17] Valérie Nachef, Jacques Patarin, and Emmanuel Volte. *Feistel Ciphers - Security Proofs and Cryptanalysis*. Springer, 2017.

[Pat03] Jacques Patarin. Luby-rackoff: 7 rounds are enough for $2^{n(1\text{-epsilon})}$security. In *Advances in Cryptology - CRYPTO 2003, 23rd Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 2003, Proceedings*, pages 513–529, 2003.

[Pat05] Jacques Patarin. On linear systems of equations with distinct variables and small block size. In *Information Security and Cryptology - ICISC 2005, 8th International Conference, Seoul, Korea, December 1-2, 2005, Revised Selected Papers*, pages 299–321, 2005.

[Pat08a] Jacques Patarin. The "coefficients h" technique. In *Selected Areas in Cryptography, 15th International Workshop, SAC 2008, Sackville, New Brunswick, Canada, August 14-15, Revised Selected Papers*, pages 328–345, 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.

[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.

[ZHY18] Ping Zhang, Honggang Hu, and Qian Yuan. Close to optimally secure variants of GCM. *Security and Communication Networks*, 2018:9715947:1–9715947:12, 2018.

# A Cheat-Sheet for Symbols and Their Definitions in Proofs

| Symbols | Definitions |
|---|---|
| $h(\mathsf{G})$ | # of solutions to the graph $\mathsf{G}$ |
| $q_m$ | # of equations edges |
| $q_v$ | # of non-equations edges |
| $q_c$ | # of vertices in $\mathsf{C}$ components |
| $s$ | # of vertices in the graph |
| $s_\ell$ | # of vertices in the left partite of a bipartite graph |
| $s_{\mathbf{r}}$ | # of vertices in the right partite of a bipartite graph |
| $h_c(i)$ | # of solutions for the subgraph $\mathsf{C} \triangleq \mathsf{C}_1 \sqcup \cdots \mathsf{C}_i$ |
| $h_d(i)$ | # of solutions for the subgraph $\mathsf{C} \sqcup \mathsf{D}_1 \sqcup \ldots \sqcup \mathsf{D}_i$ |
| $\widetilde{\mu}_{i,j}$ | # of edges from $\mathcal{E}'$ connecting vertices of $i$th and $j$-th component of $\mathsf{G}^{=}$ |
| $\mu'_i$ | # of blue dashed edges incident on $v_i$ |
| $w_i$ | # of vertices in the $i$-th component of $\mathsf{G}^{=}$ |
| $\sigma_i$ | # of vertices up to the $i$-th component of $\mathsf{G}^{=}$ |
| $\xi_{\max}$ | block maximality of a given system of bivariate affine equations |
| $\nu$ | $n$ bit nonce |
| $\tilde{\nu}$ | nonce input to the DWCDM construction, which could be the most significant $2n/3$ bits or $3n/4$ bits of the nonce |