Offset-Based BBB-Secure Tweakable Block-ciphers with Updatable Caches

Arghya Bhattacharjee[1], Ritam Bhaumik[2⋆], and Mridul Nandi[1]

[1] Indian Statistical Institute, Kolkata
[2] Inria, Paris
{bhattacharjeearghya29 bhaumik.ritam mridul.nandi}@gmail.com

Abstract. A nonce-respecting tweakable blockcipher is the building-block for the OCB authenticated encryption mode. An XEX-based TBC is used to process each block in OCB. However, XEX can provide at most birthday bound privacy security, whereas in Asiacrypt 2017, beyond-birthday-bound (BBB) forging security of OCB3 was shown in [15]. In this paper we study how at a small cost we can construct a nonce-respecting BBB-secure tweakable blockcipher. We propose the OTBC-3 construction, which maintains a cache that can be easily updated when used in an OCB-like mode. We show how this can be used in a BBB-secure variant of OCB with some additional keys and a few extra blockcipher calls but roughly the same amortised rate.

Keywords: OCB, tweakable block-cipher, authenticated encryption, updatable offsets, beyond-birthday-bound security

1 Introduction

Authenticated encryption (AE) is a symmetric-key cryptographic function for providing a combined guarantee of privacy (or confidentiality) and authenticity (or integrity) of plaintexts. Beginning with the formalisation by Katz and Yung [37] and Bellare and Namprempre [12,11], and the constructions by Jutla [35,36], the practical significance of AE has been accepted in the community, and over the last decade or so the design and analysis of AE modes has been a very active area of research in symmetric-key cryptography.

Associated data (AD) is the data that is not confidential but contributes to the authentication of the message, and AE with associated data (AEAD), formalised by Rogaway [46], takes both a plaintext and some AD as input. AEAD ensures confidentiality of plaintexts and authenticity of both plaintexts and AD. The most popular form of AEAD is based on a nonce, and is called nonce-based AEAD (NAEAD). A nonce is a non-repeating value for each encryption, and can be realised for instance with a counter. NAEAD is commonly built as a mode

⋆ This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement no. 714294 - acronym QUASYModo).

of operation of a blockcipher. However, there is often an inherent limitation on the security caused by the birthday paradox on the input or output of a blockcipher, which ensures only $(n/2)$-bit security of NAEAD if a blockcipher with n-bit blocks is used. The $(n/2)$-bit security is commonly referred to as birthday-bound (BB) security. Possible solutions to break this barrier exist, i.e., NAEAD with beyond-birthday-bound (BBB) security. However, they come with an extra computational cost.

One way to get around this obstacle is to use a tweakable blockcipher (TBC) as the underlying primitive instead of classical blockciphers. A TBC was formalised by Liskov, Rivest and Wagner [40,41], and it has an extra t-bit tweak input to provide variability, i.e., it provides a family of 2^t independent blockciphers indexed by the tweak. Starting from the early Hasty Pudding Cipher [51], many TBC designs have been proposed, including Threefish (in Skein [22]), Deoxys-BC [34], Joltik-BC [33], and KIASU-BC from the TWEAKEY framework [32], and Scream [24], where the last four schemes were submitted to CAESAR (Competition for Authenticated Encryption: Security, Applicability, and Robustness) [1]. We also see other examples including SKINNY [8,9], QARMA [6], CRAFT [10], the TBCs in the proposals for the NIST Lightweight Cryptography project [3], OPP [23] for permutation-based instantiations of OCB3 that uses a (tweakable) Even-Mansour construction, and a construction by Naito [43].

One of the most popular TBC-based NAEAD schemes is OCB. There are three main variants of OCB. The first, now called OCB1 (2001) [49], was motivated by Charanjit Jutla's IAPM [35,36]. A second version, now called OCB2 (2004) [2,47], added support for associated data (AD) and redeveloped the mode using the idea of a tweakable blockcipher. Later OCB2 was found to have a disastrous bug [28,29]. The final version of OCB, called OCB3 (2011) [39], corrected some missteps taken with OCB2 and achieved the best performance yet. OCB3 is simple, parallelisable, efficient, provably secure with BB security, and its security is well analysed [4,5,48]. It is specified in RFC 7253 [38] and was selected for the CAESAR final portfolio.

In recent times, OCB has been analysed in much detail from various perspectives. A blockcipher-based NAEAD scheme OTR and its TBC-based counterpart $\mathbb{OTR}$ were designed by Minematsu [42] which improve OCB by removing the necessity of the decryption routine of the underlying blockcipher or TBC (this property is often called as the inverse-freeness). Bhaumik and Nandi [15] showed that when the number of encryption query blocks is not more than birthday-bound (an assumption without which the privacy guarantee of OCB3 disappears), even an adversary making forging attempts with the number of blocks in the order of $2^n/\ell_{\mathrm{MAX}}$ (n being the block-size and ℓ_{MAX} being the length of the longest block) may fail to break the integrity of OCB3. Zhang et al. [53,54] described a new notion, called plaintext or ciphertext checksum (PCC), which is a generalisation of plaintext checksum (used to generate the tag of OCB), and proved that all authenticated encryption schemes with PCC are insecure in the INT-RUP security model. Then they fixed the weakness of PCC, and described a new approach called intermediate (parity) checksum (I(P)C for short). Based on the

I(P)C approach, they provided two modified schemes OCB-IC and OCB-IPC to settle the INT-RUP of OCB in the nonce-misuse setting. They proved that OCB-IC and OCB-IPC are INT-RUP up to the birthday bound in the nonce-misuse setting if the underlying tweakable blockcipher is a secure mixed tweakable pseudorandom permutation (MTPRP). The security bound of OCB-IPC is proved to be tighter than OCB-IC. To improve their speed, they utilised a "prove-then-prune" approach: prove security and instantiate with a scaled-down primitive (e.g., reducing rounds for the underlying primitive invocations). Bao et al. [7] introduced a scheme called XTX*, based on previous tweak extension schemes for TBCs, and defined ZOCB and ZOTR for nonce-based authenticated encryption with associated data. While ΘCB and $\mathbb{OTR}$ have an independent part to process AD, their schemes integrated this process into the encryption part of a plaintext by using the tweak input of the TBC, and thus achieved full absorption and full parallelisability simultaneously.

OCB has also found its place in other domains of cryptology like lightweight cryptology and quantum cryptology. Chakraborti et al. [16] proposed a lightweight authenticated encryption (AE) scheme, called Light-OCB, which can be viewed as a lighter variant of OCB as well as a faster variant of LOCUS-AEAD [17] which has been a Round 2 candidate of the NIST Lightweight Cryptography project. Bhaumik et al. [14] proposed a new rate-one parallelisable mode named QCB inspired by TAE and OCB and prove its security against quantum superposition queries.

There are two limitations on OCB that we would like to emphasise. The first is that OCB's security crucially depends on the encrypting party not repeating a nonce. The mode should never be used in situations where that can't be assured; one should instead employ a misuse-resistant AE scheme [50]. These include AES-GCM-SIV [25,26], COLM, and Deoxys-II. A second limitation of OCB is its birthday-bound degradation in provable security. This limitation implies that, given OCB's 128-bit block-size, one must avoid operating on anything near 2^{64} blocks of data. The RFC on OCB [38] asserts that a given key should be used to encrypt at most 2^{48} blocks (4 petabytes), including the associated data. Practical AE modes that avoid the birthday-bound degradation in security are now known [1,26,30,31,45].

1.1 Our Contributions

In this paper we explore ways of designing an offset-based tweakable blockcipher that can be used to obtain an OCB-like authenticated encryption mode with better security guarantees. First we show that when using an n-bit nonce (where n is the width of the block-cipher) it is difficult to go beyond the birthday-bound if we use the same offset to mask the input and the output (OTBC-0). Next we show that if we take fully independent offsets for masking inputs and outputs for each message, we get full security in the nonce-respecting scenario (OTBC-1); however, this does not fit well in the OCB-like mode, because new additional random-function calls are needed to process each message block.

We proceed to introduce the notion of *updatable offsets*, and explain why TBCs with updatable offsets are well-suited to build an OCB-like mode. Then we build a simple TBC with updatable offsets (OTBC-2), and give a birthday-attack on it that demonstrates that such a construction is not sufficient to get beyond-birthday security for the OCB. Finally, we introduce the notion of offsets that are not updatable by themselves, but are efficiently computable from updatable *caches*. As the most important technical contribution of the paper, we instantiate a TBC with this property (OTBC-3) and show that it achieves a beyond-birthday TPRP security in the number of nonces queried, as long as the maximum length of each message (i.e., the maximum number of times each block is used) is not very high. Additionally, we also show that OTBC-3 achieves at least security up to the birthday-bound even when nonce is misused and inverse queries are allowed.

Finally, we use OTBC-3 to design an authenticated encryption mode called OCB+, which is beyond-birthday secure in both privacy and authenticity. We argue how the privacy bound follows from our security proof of OTBC-3, while the authenticity can be proved in the exact same way as in [15]. OCB+ uses nine random function calls for processing each nonce, so its rate is approximately $\sigma/(\sigma + 9q)$, where σ is the total number of blocks including messages and associated data, and q is the number of distinct nonces. When the messages are sufficiently long, this rate comes close to 1, making this as efficient as OCB3, but with a BBB security guarantee.

2 Preliminaries

Throughout the paper N will mean 2^n. For any positive integer m, $[m]$ will denote the set $\{1, \ldots, m\}$. Matrices will be denoted with boldface letters, and for a matrix $\mathbf{H}$, $|\mathbf{H}|$ will denote its determinant. We'll use the Pochhammer falling factorial power notation

$$(a)_b := a(a-1)\ldots(a-b+1).$$

For ease of notation we write $+$ to denote field addition (bitwise XOR) when used between two or more field elements. Field multiplication in $\mathbb{GF}(2^n)$ is denoted with a bold dot ($\cdot$).

2.1 Distinguishing Advantage

For two oracles $\mathcal{O}_0$ and $\mathcal{O}_1$, an algorithm $\mathcal{A}$ which tries to distinguish between $\mathcal{O}_0$ and $\mathcal{O}_1$ is called a distinguishing adversary. $\mathcal{A}$ plays an interactive game with $\mathcal{O}_b$ where b is unknown to $\mathcal{A}$, and then outputs a guess for b; $\mathcal{A}$ wins when the guessed bit matches b. The distinguishing advantage of $\mathcal{A}$ is defined as

$$\mathbf{Adv}^{\mathcal{O}_1,\mathcal{O}_0}(\mathcal{A}) := \left| \Pr_{\mathcal{O}_0}[\mathcal{A} \Rightarrow 1] - \Pr_{\mathcal{O}_1}[\mathcal{A} \Rightarrow 1] \right|,$$

where the subscript of Pr denotes the oracle with which $\mathcal{A}$ is playing.

$\mathcal{O}_0$ conventionally represents an ideal primitive, while $\mathcal{O}_1$ represents either an actual construction or a mode of operation built using some other ideal primitives. We use the standard terms real oracle and ideal oracle for $\mathcal{O}_1$ and $\mathcal{O}_0$ respectively. Typically the goal of the function F represented by $\mathcal{O}_1$ is to emulate the ideal primitive F^* represented by $\mathcal{O}_0$. A security game is a distinguishing game with an optional set of additional restrictions, chosen to reflect the desired security goal. When we talk of distinguishing advantage between F and F^* with a specific security game $\mathcal{G}$ in mind, we include $\mathcal{G}$ in the subscript, e.g., $\mathbf{Adv}_{\mathcal{G}}^{F,F^*}(\mathcal{A})$. (We note that this notation is general enough to capture games where each oracle implements multiple functions, e.g., F can handle both encryption and decryption queries by accepting an extra bit to indicate the direction of queries.) Also we sometimes drop the ideal primitive and simply write $\mathbf{Adv}_{\mathcal{G}}^{F}(\mathcal{A})$ when the ideal primitive is clear from the context.

2.2 TPRP, TPRP* and TSPRP Security Notions

Given a tweak-space $\mathcal{W}$, let $\mathsf{Perm}(\mathcal{W}, n)$ be the set of all functions $\tilde{\pi} : \mathcal{W} \times \{0,1\}^n \to \{0,1\}^n$ such that for any tweak $W \in \mathcal{W}$, $\tilde{\pi}(W, \cdot)$ is a permutation over $\{0,1\}^n$. Then a $\tilde{\pi}^*$ distributed uniformly at random over $\mathsf{Perm}(\mathcal{W}, n)$ will be called a *tweakable random permutation* (TRP).

Let $\mathcal{K}$ denote a key-space. Then $\tilde{E} : \mathcal{K} \times \mathcal{W} \times \{0,1\}^n \to \{0,1\}^n$ will be called a *tweakable pseudorandom permutation* (TPRP) if for a key K distributed uniformly at random over $\mathcal{K}$ and for any adversary $\mathcal{A}$ trying to distinguish $\tilde{E}_K := \tilde{E}(K, \cdot, \cdot)$ from $\tilde{\pi}^*$, $\mathbf{Adv}^{\tilde{E}_K, \tilde{\pi}^*}(\mathcal{A})$ is small. We call this game the TPRP game and denote the advantage of $\mathcal{A}$ as $\mathbf{Adv}_{\mathrm{TPRP}}^{\tilde{E}}(\mathcal{A})$ in short.

We will be more interested in a modified version of the TPRP game, where $\mathcal{A}$ is under the added restriction that no two queries can be made with the same tweak. We call this the *tweak respecting pseudorandom permutation* (TPRP*) game, and denote the corresponding advantage of $\mathcal{A}$ as $\mathbf{Adv}_{\mathrm{TPRP}^*}^{\tilde{E}}(\mathcal{A})$.

Finally, the *tweakable strong pseudorandom permutation* (TSPRP) game allows $\mathcal{A}$ to make both encryption and decryption queries to the oracle. The advantage term of $\mathcal{A}$ in a TSPRP game will be denoted $\mathbf{Adv}_{\mathrm{TSPRP}}^{\tilde{E}}(\mathcal{A})$.

2.3 Authenticated Encryption and Its Security Notion

A *nonce-based Authenticated Encryption with associated data* (NAEAD) involves a key-space $\mathcal{K}$, a nonce-space $\mathcal{N}$, an associated-data-space $\mathcal{AD}$, a message space $\mathcal{M}$ and a tag space $\mathcal{T}$ along with two functions $\mathsf{Enc} : \mathcal{K} \times \mathcal{N} \times \mathcal{AD} \times \mathcal{M} \to \mathcal{M} \times \mathcal{T}$ (called the Encryption Function) and $\mathsf{Dec} : \mathcal{K} \times \mathcal{N} \times \mathcal{AD} \times \mathcal{M} \times \mathcal{T} \to \mathcal{M} \cup \{\perp\}$ (called the Decryption Function) with the correctness condition that for any $K \in \mathcal{K}, N \in \mathcal{N}, A \in \mathcal{AD}$ and $M \in \mathcal{M}$, it holds that

$$\mathsf{Dec}(K, N, A, \mathsf{Enc}(K, N, A, M)) = M\,.$$

The NAEAD security game is played between the $(\mathsf{Enc}, \mathsf{Dec})$ scheme described above and an ideal oracle $(\mathsf{Enc}^*, \mathsf{Dec}^*)$ where $\mathsf{Enc}^* : \mathcal{K} \times \mathcal{N} \times \mathcal{AD} \times \mathcal{M} \to$

$\mathcal{M} \times \mathcal{T}$ is an ideal random function and $\mathsf{Dec}^* : \mathcal{K} \times \mathcal{N} \times \mathcal{AD} \times \mathcal{M} \times \mathcal{T} \to \{\perp\}$ is a constant function. The adversary $\mathcal{A}$ can make encryption or decryption queries to the oracle. In addition we assume the following restrictions:

1. $\mathcal{A}$ should be once-respecting, i.e., should not repeat a nonce in more than one encryption queries; and
2. $\mathcal{A}$ should not make *pointless* queries, i.e., should not repeat the same query multiple times or should not make the decryption query (N, A, C, T) if it has already made an encryption query (N, A, M) and received (C, T) in response.

The distinguishing advantage of $\mathcal{A}$ for an NAEAD scheme $\mathcal{E}$ will be denoted by $\mathbf{Adv}^{\mathcal{E}}_{\mathrm{NAEAD}}(\mathcal{A})$. The following two security notions are captured in this advantage.

1. Privacy or Confidentiality, i.e., $\mathcal{A}$ should not be able to distinguish the real oracle from the ideal oracle.
2. Authenticity or Integrity, i.e., $\mathcal{A}$ should not be able to forge the real oracle. In other words, $\mathcal{A}$ should not be able to make a decryption query to the real oracle to which the response isn't $\perp$.

2.4 Coefficients H Technique

The H-coefficient technique is a proof method by Patarin [44] that was modernized by Chen and Steinberger [18,52]. A distinguisher $\mathcal{A}$ interacts with oracles $\mathcal{O}$ (The oracle $\mathcal{O}$ could be a sequence of multiple oracles.) and obtains outputs from a real world $\mathcal{O}_1$ or an ideal world $\mathcal{O}_0$. The results of its interaction are collected in a transcript τ. The oracles can sample random coins before the experiment (often a key or an ideal primitive that is sampled beforehand) and are then deterministic. A transcript τ is attainable if $\mathcal{A}$ can observe τ with non-zero probability in the ideal world.

The Fundamental Theorem of the H-coefficients technique, whose proof can be found, e.g., in [18,44,52], states the following:

Theorem 1 ([44]). Assume, there exist $\epsilon_1, \epsilon_2 \geq 0$ such that

$$\Pr_{\mathcal{O}_0}[\mathsf{bad}] \leq \epsilon_1,$$

and for any attainable transcript τ obtained without encountering bad,

$$\frac{\Pr_{\mathcal{O}_1}[\tau]}{\Pr_{\mathcal{O}_0}[\tau]} \geq 1 - \epsilon_2.$$

Then, for all adversaries $\mathcal{A}$, it holds that $\mathbf{Adv}^{\mathcal{O}_0, \mathcal{O}_1}(\mathcal{A}) \leq \epsilon_1 + \epsilon_2$.

The technique has been generalized by Hoang and Tessaro [27] in their expectation method, which allowed them to derive the Fundamental Theorem as a corollary. Since we only consider bad events in the ideal world, we will write $\Pr_{\mathcal{O}_0}[\mathsf{bad}]$ simply as $\Pr[\mathsf{bad}]$ when there is no scope for confusion; the same notation is used when the event bad is broken down into further sub-events.

2.5 Mirror Theory

Consider a sequence of n-bit variables $W_1, \ldots, W_t$, subject to r bi-variate equations of the form

$$W_i + W_j = \delta_{ij}.$$

Consider the graph with $W_1, \ldots, W_t$ as vertices and the bi-variate equations as weighted edges with δ_{ij} the weight between W_i and W_j. Suppose we can show that the graph is cycle-free, and that each path has a non-zero sum of weights. Let $\xi_{\max}$ be the size of the largest component of this graph. Then Mirror Theory tells us that as long as $\xi_{\max}^2 \leq \sqrt{N}/\log_2 N$ and $t \leq N/12\xi_{\max}^2$, the number of solutions to the system of equations such that W_i's are all distinct is at least $(N)_t/N^r$. [19,21]

3 Finding a Suitable Tweakable Block-cipher

We set out to find an offset-based Tweakable Block-cipher that could give us a beyond-birthday security bound for OCB+. The general structure of this is as follows:

$$C = \pi(M + T) + \widehat{T},$$

where the offsets T and $\widehat{T}$ are functions of the nonce $\mathcal{N}$ and the block-number i.

3.1 Attempt with Same Offset

The first question we asked is whether it is possible to achieve this by having $T = \widehat{T}$, i.e., adding the same offset before and after the blockcipher call, like in OCB. The most powerful version of this is to have

$$T = \widehat{T} = f(\mathcal{N}, i)$$

for some $2n$-bit-to-n-bit random function f. This we call OTBC-0, defined as

$$\mathsf{OTBC\text{-}0}(\mathcal{N}, i, M) := \pi(M + f(\mathcal{N}, i)) + f(\mathcal{N}, i).$$

This construction is shown in Fig. 1.

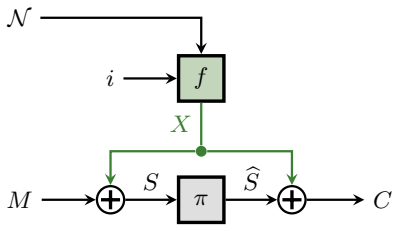

Fig. 1: OTBC-0: Same offset.

Birthday Attack on OTBC-0. Unfortunately, OTBC-0 fails to give us beyond birthday-bound security. This is because for two queries with the same message, there is a collision in the ciphertext whenever there is a collision in the output of f; in addition the ciphertext-collision can also happen if the sum of the outputs of π and f collide. This shows that the collision probability at C is roughly double the collision probability in an ideal tweakable block-cipher, which can be detected in the birthday-bound. A more formal description of the attack is given in Appendix A.

3.2 Independent Offsets

We deduce from the preceding subsection that using the same offset above and below can never give us beyond-birthday TPRP* security for the tweakable block-cipher. We next examine the most powerful version of this possible, where the two offsets on either side of π come from two completely independent $2n$-bit-to-n-bit random functions f_1 and f_2. This we call OTBC-1, defined as

$$\mathsf{OTBC\text{-}1}(\mathcal{N}, i, M) := \pi(M + f_1(\mathcal{N}, i)) + f_2(\mathcal{N}, i).$$

This construction is shown in Fig. 2.

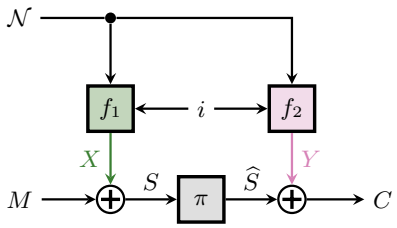

Fig. 2: OTBC-1: Different random offsets.

Security of OTBC-1. As it turns out, OTBC-1 trivially achieves full TPRP* security. This is because in a tweak-respecting game, the offsets are always random and independent of all other offsets in the game, making it impossible to glean any information from the oracle responses. We formally state this as the following theorem, the proof of which is given in Appendix B.

Theorem 2. *For any TPRP* adversary $\mathcal{A}$ making q queries, we have*

$$\boldsymbol{Adv}^{\mathsf{OTBC\text{-}1}}_{\widetilde{\mathrm{TPRP}}^*}(\mathcal{A}) = 0.$$

3.3 Updatable Offsets

While OTBC-1 is a fully secure tweakable blockcipher, it's not very interesting to us in the context of OCB+. This is because when the same nonce is used with different block-numbers (as we need for OCB+), new calls to f_1 and f_2 are needed for each new block-number. Thus we need three primitive calls to process every block of message, which robs us of the main advantage of an OCB-like design.

This points us to the next desirable feature we need in the offsets: they should be *efficiently updatable* when we keep the nonce same and increment the block-number. We call a $2n$-bit-to-n-bit function h efficiently updatable on the second input if there is an efficiently computable function g (called the *update* function) such that for each i we have

$$h(\mathcal{N}, i+1) = g(i, h(\mathcal{N}, i)).$$

In other words, given $h(\mathcal{N}, i)$ has already been computed, $h(\mathcal{N}, i+1)$ can be computed through the update function g while bypassing a fresh call to h. (For this to make sense, of course, h should be computationally heavy and g should be much faster than h.) Note that the update function may or may not use i as an additional argument; while in this work we'll only consider update functions that are *stationary* (i.e., ignore the block-number i, and apply the same function at each block to get the offset for the next block), it is possible to have an update function that varies with i but still satisfies the above-discussed criteria.

The simplest updatable design. The simplest way to design an updatable function is to call a random function f on the nonce $\mathcal{N}$ once, and then use a stationary update function to obtain the offset for each successive block-number. This can be formally defined as follows:

$$\begin{aligned} h(\mathcal{N}, 1) &= g(f(\mathcal{N})), \\ h(\mathcal{N}, i) &= g(h(\mathcal{N}, i-1)) = g^i(f(\mathcal{N})), \qquad i \geq 2. \end{aligned}$$

Using these updatable offsets with two independent random functions f_1 and f_2 for input-masking and output-masking respectively, we can define a tweakable block-cipher OTBC-g as

$$\mathsf{OTBC\text{-}g}(\mathcal{N}, i, M) = \pi\left(M + g^i(f_1(\mathcal{N}))\right) + g^i(f_2(\mathcal{N})).$$

Instantiating OTBC-g. In commonly used finite fields, there generally exist primitive elements that allow very fast multiplication. As an instantiation of g, we use multiplication with one such fixed primitive α. Concretely, we define the update function as

$$g(f(\mathcal{N})) = \alpha \cdot f(\mathcal{N}).$$

Thus, we use as the updatable offsets

$$T = \alpha^i \cdot f_1(\mathcal{N}), \qquad \widehat{T} = \alpha^i \cdot f_2(\mathcal{N}).$$

This gives us the construction OTBC-2, defined as

$$\mathsf{OTBC\text{-}2}(\mathcal{N}, i, M) = \pi\left(M + \alpha^i \cdot f_1(\mathcal{N})\right) + \alpha^i \cdot f_2(\mathcal{N}).$$

This construction is shown in Fig. 3.

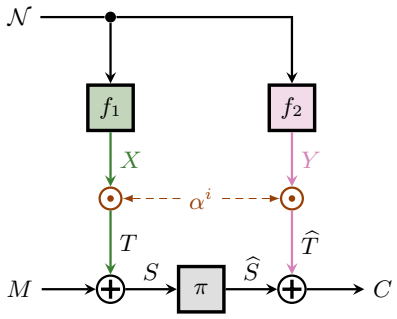

Fig. 3: OTBC-2: Updatable offsets with two independent random-function calls.

Attack on OTBC-2. Unfortunately, this simple updatable function is not sufficient to give us beyond-birthday-bound security. This is because since the update function is linear and publicly known, we can make queries such that successive message blocks under the same nonce follow the update relation, which forces the successive S blocks to also conform to the update relation. Thus, one collision on S between two different nonces ensures that successive blocks also see an S-collision, which can be exploited in a distinguishing attack. This we state as the following theorem, the proof of which is given in Appendix C.

Theorem 3. *There exists a distinguisher $\mathcal{A}$ querying with q nonces and L blocks under each nonce with $L \geq 12$ in a TPRP* game against OTBC-2 such that*

$$\boldsymbol{Adv}^{\mathsf{OTBC\text{-}2}}_{\mathit{TPRP}^*}(\mathcal{A}) \geq \Omega\left(\frac{q^2L^2}{N}\right).$$

3.4 Offsets with Updatable Caches

To get around this problem, we observe that in order to use an offset-based tweakable block-cipher in OCB+, we don't really need it to be updatable; it is enough for it to maintain a small and updatable hidden state or *cache*, such that the offsets are efficiently computable from the cache. Letting ψ denote the *caching function*, g the update function as before, h the offset-generating function, and φ the cache-to-offset function, we have

$$\psi(\mathcal{N}, i+1) = g(i, \psi(\mathcal{N}, i)), \qquad h(\mathcal{N}, i) = \varphi(\psi(\mathcal{N}, i)).$$

Again, for this to make sense, g and ψ should be computationally heavy when computed from scratch, while g and φ should be much faster.

Updatable Caches, Non-updatable Offsets. To avoid the kind of attack that we found on OTBC-2, we want to design a tweakable block-cipher with offsets which are not themselves updatable, but are efficiently computable from updatable caches. This makes the offsets more independent, while still giving us a means of updating them efficiently at a small additional cost.

One simple way to achieve this is to use two independent random functions f_1 and f_2 on the nonce, put the outputs in the cache as two different *branches*, and use two different update functions g and g' on the two branches; the offset can then be generated as the sum of the two branches. This can be formally defined as follows:

$$\begin{aligned}
\psi(\mathcal{N}, 1) &= (g(f_1(\mathcal{N})), g'(f_2(\mathcal{N}))),\\
\psi(\mathcal{N}, i) &= [g, g'](\psi(\mathcal{N}, i-1)) = (g^i(f_1(\mathcal{N})), g'^i(f_2(\mathcal{N}))), \quad i \geq 2,\\
\varphi(x, y) &= x + y,\\
h(\mathcal{N}, i) &= g^i(f_1(\mathcal{N})) + g'^i(f_2(\mathcal{N})) = \varphi(\psi(\mathcal{N}, i)),
\end{aligned}$$

where $[g, g']$ denotes the two-input function that applies g to the first input and g' to the second input. Note that $h(\mathcal{N}, i)$ is not efficiently computable from $h(\mathcal{N}, i-1)$ without accessing the cache $\psi(\mathcal{N}, i-1)$, which makes the offsets themselves non-updatable in the absence of the cache. Using these offsets we can define a tweakable block-cipher OTBC-gg' as

$$\mathsf{OTBC\text{-}gg'}(\mathcal{N}, i, M) = \pi\left(M + g^i(f_1(\mathcal{N})) + g'^i(f_2(\mathcal{N}))\right) + f_3(\mathcal{N}) + g^i(\pi(0^n)).$$

where f_3 is a third independent random-function. Note that we do not bother to use the non-updatable updates for masking the output, because $\mathcal{A}$ can make only encryption queries, and thus cannot exploit the same weakness in the output-masking.

Instantiating OTBC-gg'. As the main contribution of this section, we propose a concrete instantiation of OTBC-gg' and analyse its security. As before we keep the field-multiplication by α as g, and for g' we use field-multiplication by α^2. The resulting tweakable block-cipher, called OTBC-3, is defined as

$$\mathsf{OTBC\text{-}3}(\mathcal{N}, i, M) = \pi\left(M + \alpha^i \cdot f_1(\mathcal{N}) + \alpha^{2i} \cdot f_2(\mathcal{N})\right) + f_3(\mathcal{N}) + \alpha^i \cdot \pi(0^n).$$

This construction is shown in Fig. 4.

3.5 TPRP* Security Analysis of OTBC-3

Consider a distinguisher $\mathcal{A}$ making σ encryption queries to OTBC-3 with q distinct nonces and $\ell^{(j)} \leq L$ block-numbers $1, \ldots, \ell^{(j)}$ for the j-th nonce for each $j \in [q]$. Then we have the following result.

Theorem 4. *As long as $\sigma \leq N/n^2L^2$, we have*

$$\boldsymbol{Adv}^{\mathsf{OTBC\text{-}3}}_{TPRP^*}(\mathcal{A}) \leq \frac{n\sigma L}{N}.$$

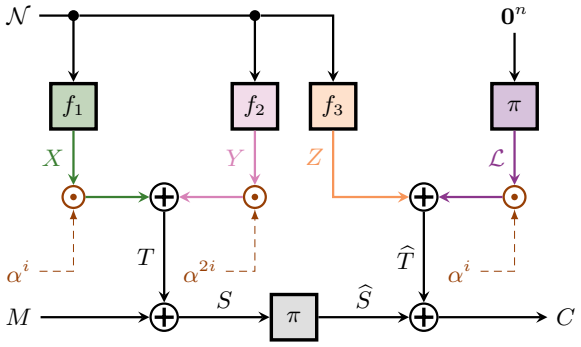

Fig. 4: OTBC-3: Offsets with updatable caches using three independent random-function calls.

Algorithm 1 $\mathsf{OTBC\text{-}3}^{f_1,f_2,f_3,\pi}(\mathcal{N}, i, M)$

1: $T \leftarrow \alpha^i f_1(\mathcal{N}) \oplus \alpha^{2i} f_2(\mathcal{N})$
2: $\widehat{T} \leftarrow f_3(\mathcal{N}) \oplus \alpha^i \pi(0^n)$
3: $S \leftarrow M \oplus T$
4: $\widehat{S} \leftarrow \pi(S)$
5: $C \leftarrow \widehat{S} \oplus \widehat{T}$
6: **return** C

Proof. In this proof, we'll use the following lemma, the proof of which is deferred to Appendix D.

Lemma 1. *For some $r \geq 2$ and $2r$ numbers $i_1, i'_1, \ldots, i_r, i'_r < N$ such that $i_j \neq i'_j$ for each $j \in [r]$, define*

$$\mathbf{B}_r = \begin{bmatrix} \alpha^{i_1} & \alpha^{2i_1} & \alpha^{i'_2} & \alpha^{2i'_2} & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & \alpha^{i_2} & \alpha^{2i_2} & \alpha^{i'_3} & \alpha^{2i'_3} & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \alpha^{i_3} & \alpha^{2i_3} & \alpha^{i'_4} & \alpha^{2i'_4} & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & \alpha^{i_{r-1}} & \alpha^{2i_{r-1}} & \alpha^{i'_r} & \alpha^{2i'_r} \\ \alpha^{i'_1} & \alpha^{2i'_1} & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & \alpha^{i_r} & \alpha^{2i_r} \end{bmatrix}.$$

Then $\mathbf{B}_r$ is at least of rank r.

Label the q nonces $\mathcal{N}^{(1)}, \ldots, \mathcal{N}^{(q)}$. For the j-th nonce, there are $\ell^{(j)}$ queries $(\mathcal{N}^{(j)}, 1, M_1^{(j)}), \ldots, (\mathcal{N}^{(j)}, \ell^{(j)}, M_{\ell^{(j)}}^{(j)})$, with outputs $(C_1^{(j)}, \ldots C_{\ell^{(j)}}^{(j)})$ respectively. For the internal transcript, we have $\mathcal{L}$, the encryption of 0 with π, and for the j-th nonce, we have the three random-function outputs $X^{(j)}, Y^{(j)}, Z^{(j)}$; finally, we have the (input, output) pairs $(S_1^{(j)}, \widehat{S}_1^{(j)}), \ldots, (S_{\ell^{(j)}}^{(j)}, \widehat{S}_{\ell^{(j)}}^{(j)})$ to π, and the (input-offset, output-offset) pairs $(T_1^{(j)}, \widehat{T}_1^{(j)}), \ldots, (T_{\ell^{(j)}}^{(j)}, \widehat{T}_{\ell^{(j)}}^{(j)})$. Then this extended transcript satisfies the following equations for each $j \in [q]$ and each

$i \in [\ell^{(j)}]$:

$$S_i^{(j)} = M_i^{(j)} + T_i^{(j)}, \qquad \widehat{S}_i^{(j)} = C_i^{(j)} + \widehat{T}_i^{(j)},$$
$$T_i^{(j)} = \alpha^i \cdot X^{(j)} + \alpha^{2i} \cdot Y^{(j)}, \qquad \widehat{T}_i^{(j)} = Z^{(j)} + \alpha_i \cdot \mathcal{L}.$$

Internal Sampling. Following the query phase of the game, in the ideal world we sample the internal transcript as follows (subject to certain bad events to be defined subsequently):

- Sample $X^{(j)}, Y^{(j)}$ uniformly at random for each $j \in [q]$;
- Check for bad1, bad2, bad3, bad4;
- Sample $\mathcal{L}$ uniformly at random;
- Check for bad5, bad6;
- Let $S_1, \ldots, S_t$ be a labeling of the unique values in $\{S_i^{(j)} \mid j \in [q], i \in [\ell^{(j)}]\}$;
- Sample $\{\widehat{S}_k \mid k \in [t]\}$ directly from good set, subject to the equations $\widehat{S}_i^{(j)} + \widehat{S}_{i'}^{(j)} = C_i^{(j)} + C_{i'}^{(j)} + (\alpha^i + \alpha^{i'}) \cdot \mathcal{L}$ for each $j \in [q]$ and each $i, i' \in [\ell^{(j)}]$.

Before describing the bad events bad1, ..., bad6, we define two graphs on the extended transcript.

Transcript Graph. For distinct $j_1, j_2 \in [q]$, there is an edge (j_1, j_2) in G if we have some $i_1 \in [\ell^{(j_1)}]$ and some $i_2 \in [\ell^{(j_2)}]$ such that $S_{i_1}^{(j_1)} = S_{i_2}^{(j_2)}$.

We will refer to paths of length 2 in G as *links*. A link (j_1, j_2, j_3) formed with the collisions $S_{i_1}^{(j_1)} = S_{i_2}^{(j_2)}$ and $S_{i_2'}^{(j_2)} = S_{i_3}^{(j_3)}$ for some $i_1 \in [\ell^{(j_1)}]$, $i_2, i_2' \in [\ell^{(j_2)}]$ and $i_3 \in [\ell^{(j_3)}]$ is called *degenerate* if $i_2 = i_2'$ and *non-degenerate* otherwise. We observe that the above link being degenerate implies $S_{i_1}^{(j_1)} = S_{i_3}^{(j_3)}$, so (j_1, j_3) is also an edge in G. By *short-circuiting* a degenerate link (j_1, j_2, j_3) we will refer to the operation of replacing it with the edge (j_1, j_3).

A path of length ≥ 3 is called non-degenerate if at least one of its sublinks is non-degenerate. When a non-degenerate path contains a degenerate sublink, we can short-circuit it to obtain a shorter non-degenerate path. We can repeat this operation as long as the path contains degenerate sublinks to end up with a *minimal* non-degenerate path. When the initial path is a cycle, we end up with either a minimal non-degenerate cycle or a *double-collision* edge, i.e., an edge (j_1, j_2) in G such that for distinct $i_1, i_1' \in [\ell^{(j_1)}]$ and distinct $i_2, i_2' \in [\ell^{(j_2)}]$ we have $S_{i_1}^{(j_1)} = S_{i_2}^{(j_2)}$, and $S_{i_1'}^{(j_1)} = S_{i_2'}^{(j_2)}$.

Dual Graph (for Mirror Theory). We also define a second graph H on the transcript, which is something of a dual of the first. This is the graph we need to check for the conditions necessary to apply mirror theory. First consider the graph H' such that the vertices of H' are the distinct values $S_1, \ldots, S_t$, and there is an edge between S_i and $S_{i'}$ in H if they appear in the same nonce, i.e., if there

is some $j \in [q]$, $i, i' \in [\ell^{(j)}]$ such that $\widehat{S}_i^{(j)} + \widehat{S}_{i'}^{(j)} = C_i^{(j)} + C_{i'}^{(j)} + (\alpha^i + \alpha^{i'}) \cdot \mathcal{L}$; further, the weight of this edge is then $C_i^{(j)} + C_{i'}^{(j)} + (\alpha^i + \alpha^{i'}) \cdot \mathcal{L}$.

From H' we get H by dropping all *redundant edges*—for each $j \in [\ell^{(j)}]$, out of the fully connected subgraph of G with $\binom{\ell^{(j)}}{2}$ edges, we only keep a spanning tree of $\ell^{(j)} - 1$ edges, and drop the rest. For instance, one way of choosing H could be to just keep the edge between $\widehat{S}_i^{(j)}$ and $\widehat{S}_{i+1}^{(j)}$ for each $i \in [\ell^{(j)} - 1]$. (Note that we assume here that all $\widehat{S}_i^{(j)}$ are distinct within any j, because that is the only use-case we'll need; the notions however easily generalise to graphs with intra-nonce collisions.)

We observe that H is cycle-free as long as G is cycle-free, and that the size $\xi_{\max}$ of the largest component of H is at most LM when M is the size of the largest component of G.

Bad Events. Based on the graphs G and H defined above, we can describe our bad events.

bad1: *We have $j \in [q]$ and distinct $i, i' \in [\ell^{(j)}]$ such that $S_i^{(j)} = S_{i'}^{(j)}$.*
bad2: *There is a double-collision edge in G.*
bad3: *There is a minimal non-degenerate cycle in G.*
bad4: *G has a component of size $> n$.*
bad5: *We have $j \in [q]$ and distinct $i, i' \in [\ell^{(j)}]$ such that $C_i^{(j)} + C_{i'}^{(j)} = (\alpha^i + \alpha^{i'}) \cdot \mathcal{L}$.*
bad6: *We have a path in H on which the edge-weights sum to 0.*

Next we give an upper bound on the probability of at least one bad event happening in the ideal world. Define

$$\mathsf{bad} := \bigcup_{p=1}^{6} \mathsf{bad[p]}.$$

Then we have the following lemma.

Lemma 2. *In the ideal world,*

$$\Pr[\mathit{bad}] \leq \frac{n\sigma L}{N}.$$

Proof (of Lemma 1). We bound the probability of each of the six bad events one by one below.

bad1: *We have $j \in [q]$ and distinct $i, i' \in [\ell^{(j)}]$ such that $S_i^{(j)} = S_{i'}^{(j)}$.*
For a fixed choice of indices j, i and i', the probability of the event comes out to be $1/N$ due to the randomness of $T_i^{(j)}$ or $T_{i'}^{(j)}$. From union bound over all possible choices of indices, we obtain

$$\Pr[\mathsf{bad1}] \leq \frac{1}{N} \sum_{j=1}^{q} \ell^{(j)2} \leq \frac{L}{N} \sum_{j=1}^{q} \ell^{(j)} \leq \frac{\sigma L}{N}.$$

bad2: *There is a double-collision edge in G.*
This implies that we have distinct $j_1, j_2 \in [q]$, distinct $i_1, i'_1 \in [\ell^{(j_1)}]$, and distinct $i_2, i'_2 \in [\ell^{(j_2)}]$ such that $S^{(j_1)}_{i_1} = S^{(j_2)}_{i_2}$, and $S^{(j_1)}_{i'_1} = S^{(j_2)}_{i'_2}$. This can be written as $\mathbf{B}_2\mathbf{v} = \mathbf{c}$, where

$$\mathbf{B}_2 = \begin{bmatrix} \alpha^{i_1} & \alpha^{2i_1} & \alpha^{i_2} & \alpha^{2i_2} \\ \alpha^{i'_1} & \alpha^{2i'_1} & \alpha^{i'_2} & \alpha^{2i'_2} \end{bmatrix}, \mathbf{v} = \begin{bmatrix} X^{(j_1)} \\ Y^{(j_1)} \\ X^{(j_2)} \\ Y^{(j_2)} \end{bmatrix}, \mathbf{c} = \begin{bmatrix} M^{(j_1)}_{i_1} + M^{(j_2)}_{i_2} \\ M^{(j_1)}_{i'_1} + M^{(j_2)}_{i'_2} \end{bmatrix}.$$

$\mathbf{B}_2$ is of rank 2 by Lemma 1. Thus, when we fix $j_1, j_2, i_1, i'_1, i_2, i'_2$, we have

$$\Pr[\mathbf{B}_2\mathbf{v} = \mathbf{c}] \le \frac{1}{N^2}.$$

Thus,

$$\Pr[\mathsf{bad2}] \le \frac{1}{N^2} \sum_{j_1=1}^{q} \sum_{j_2=1}^{q} \ell^{(j_1)2}\ell^{(j_2)2} \le \frac{L^2}{N^2} \sum_{j_1=1}^{q} \sum_{j_2=1}^{q} \ell^{(j_1)}\ell^{(j_2)} \le \frac{\sigma^2 L^2}{N^2}.$$

bad3: *There is a minimal non-degenerate cycle in the transcript graph.*
First, suppose there is a minimal non-degenerate cycle of length 3. Thus, we have distinct $j_1, j_2, j_3 \in [q]$, distinct $i_1, i'_1 \in [\ell^{(j_1)}]$, distinct $i_2, i'_2 \in [\ell^{(j_2)}]$, and distinct $i_3, i'_3 \in [\ell^{(j_3)}]$ such that $S^{(j_1)}_{i_1} = S^{(j_2)}_{i'_2}$, $S^{(j_2)}_{i_2} = S^{(j_3)}_{i'_3}$, and $S^{(j_3)}_{i_3} = S^{(j_1)}_{i'_1}$. (We name the indices like this for symmetry.) As before, this can be written as $\mathbf{B}_3\mathbf{v} = \mathbf{c}$, where

$$\mathbf{B}_3 = \begin{bmatrix} \alpha^{i_1} & \alpha^{2i_1} & \alpha^{i'_2} & \alpha^{2i'_2} & 0 & 0 \\ 0 & 0 & \alpha^{i_2} & \alpha^{2i_2} & \alpha^{i'_3} & \alpha^{2i'_3} \\ \alpha^{i'_1} & \alpha^{2i'_1} & 0 & 0 & \alpha^{i_3} & \alpha^{2i_3} \end{bmatrix}, \mathbf{v} = \begin{bmatrix} X^{(j_1)} \\ Y^{(j_1)} \\ X^{(j_2)} \\ Y^{(j_2)} \\ X^{(j_3)} \\ Y^{(j_3)} \end{bmatrix}, \mathbf{c} = \begin{bmatrix} M^{(j_1)}_{i_1} + M^{(j_2)}_{i'_2} \\ M^{(j_2)}_{i_2} + M^{(j_3)}_{i'_3} \\ M^{(j_3)}_{i_3} + M^{(j_1)}_{i'_1} \end{bmatrix}.$$

$\mathbf{B}_3$ is of rank 3 by Lemma 1. Thus, when we fix $j_1, j_2, j_3, i_1, i'_1, i_2, i'_2, i_3, i'_3$, we have

$$\Pr[\mathbf{B}_3\mathbf{v} = \mathbf{c}] \le \frac{1}{N^3}.$$

Next, suppose there is a minimal non-degenerate cycle of length $r \ge 4$. Thus we have distinct $j_1, \ldots, j_r \in [q]$; for $u \in [r-1]$ we have $i_u \in [\ell^{(j_u)}]$ and $i'_{u+1} \in [\ell^{(j_{u+1})}]$ such that $S^{(j_u)}_{i_u} = S^{(j_{u+1})}_{i'_{u+1}}$; and finally, we have $i_r \in [\ell^{(j_r)}]$ and $i'_1 \in [\ell^{(j_1)}]$ such that $S^{(j_r)}_{i_r} = S^{(j_1)}_{i'_1}$; the cycle being minimal non-degenerate

implies that for each $u \in [r]$, $i_u \neq i'_u$. This can be written as $\mathbf{B}_r\mathbf{v} = \mathbf{c}$, where

$$\mathbf{B}_r = \begin{bmatrix} \alpha^{i_1} & \alpha^{2i_1} & \alpha^{i'_2} & \alpha^{2i'_2} & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & \alpha^{i_2} & \alpha^{2i_2} & \alpha^{i'_3} & \alpha^{2i'_3} & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \alpha^{i_3} & \alpha^{2i_3} & \alpha^{i'_4} & \alpha^{2i'_4} & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & \alpha^{i_{r-1}} & \alpha^{2i_{r-1}} & \alpha^{i'_r} & \alpha^{2i'_r} \\ \alpha^{i'_1} & \alpha^{2i'_1} & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & \alpha^{i_r} & \alpha^{2i_r} \end{bmatrix},$$

$$\mathbf{v} = \begin{bmatrix} X^{(j_1)} \\ Y^{(j_1)} \\ X^{(j_2)} \\ Y^{(j_2)} \\ \vdots \\ X^{(j_r)} \\ Y^{(j_r)} \end{bmatrix}, \mathbf{c} = \begin{bmatrix} M_{i_1}^{(j_1)} + M_{i'_2}^{(j_2)} \\ M_{i_2}^{(j_2)} + M_{i'_3}^{(j_3)} \\ M_{i_3}^{(j_3)} + M_{i'_4}^{(j_4)} \\ \vdots \\ M_{i_{r-1}}^{(j_{r-1})} + M_{i'_r}^{(j_r)} \\ M_{i_r}^{(j_r)} + M_{i'_1}^{(j_1)} \end{bmatrix}.$$

$\mathbf{B}_r$ is of rank r by Lemma 1. Thus, for each $r \geq 3$, when we fix $j_1, \ldots, j_r, i_1, i'_1, \ldots, i_r, i'_r$, we have

$$\Pr[\mathbf{B}_r\mathbf{v} = \mathbf{c}] \leq \frac{1}{N^r}.$$

Assuming $2\sigma L \leq N$, we have

$$\begin{aligned} \Pr[\mathsf{bad3}] &\leq \sum_{r=3}^{q} \frac{\prod_{u=1}^{r} \ell^{(j_u)2}}{N^r} \\ &\leq \sum_{r=3}^{q} \left(\left(\frac{L}{N}\right)^r \prod_{u=1}^{r} \ell^{(j_u)} \right) \leq \sum_{r=3}^{q} \left(\frac{\sigma L}{N}\right)^r \leq \frac{2\sigma^3 L^3}{N^3}. \end{aligned}$$

bad4: *G has a component of size $> n$.*

For a component of size M, the minimum number of nonces in that component should be $p+1$ where $p = \lceil M/L \rceil - 1$ with p collisions among themselves. In other words, $\exists$ distinct $j_1, j_2, \cdots, j_{p+1} \in [q]$ and $i_1 \in \ell^{(j_1)}$, $i_2, i'_2 \in \ell^{(j_2)}$, $i_3, i'_3 \in \ell^{(j_3)}$, $\cdots$, $i_p, i'_p \in \ell^{(j_p)}$, $i_{p+1} \in \ell^{(j_{p+1})}$ such that

$$S_{i_1}^{(j_1)} = S_{i_2}^{(j_2)}, S_{i'_2}^{(j_2)} = S_{i_3}^{(j_3)}, \ldots, S_{i'_p}^{(j_p)} = S_{i_{p+1}}^{(j_{p+1})}.$$

For a fixed choice of indices, the probability of the event comes out to be $1/N^p$. The independence assumption comes from the fact that every equation from the system of equations mentioned above introduces a fresh nonce. From

union bound over all the possible choices of indices, we obtain

$$\begin{aligned}\Pr[\mathsf{bad4}] &\le \frac{1}{N^p}\sum_{j_1=1}^{q}\sum_{j_2=1}^{q}\cdots\sum_{j_{p+1}=1}^{q}\ell^{(j_1)2}\ell^{(j_2)2}\ldots\ell^{(j_{p+1})2}\\ &\le \frac{L^{p+1}}{N^p}\sum_{j_1=1}^{q}\sum_{j_2=1}^{q}\cdots\sum_{j_{p+1}=1}^{q}\ell^{(j_1)}\ell^{(j_2)}\ldots\ell^{(j_{p+1})}\\ &\le \frac{\sigma^{p+1}L^{p+1}}{N^p} = \frac{\sigma L}{N}\left(\frac{\sigma^p L^p}{N^{p-1}}\right).\end{aligned}$$

Assuming $\sigma L \le N/2$ and $p = n$, we get

$$\Pr[\mathsf{bad4}] \le \frac{\sigma L}{N}.$$

bad5: *We have $j \in [q]$ and distinct $i, i' \in [\ell_j]$ such that $C_i^{(j)} + C_{i'}^{(j)} = (\alpha^i + \alpha^{i'})\cdot\mathcal{L}$.* For a fixed choice of indices j, i and i', the probability of the event comes out to be $1/N$ due to the randomness of $\mathcal{L}$. From union bound over all possible choices of indices, we obtain

$$\Pr[\mathsf{bad5}] \le \frac{1}{N}\sum_{j=1}^{q}\ell^{(j)2} \le \frac{L}{N}\sum_{j=1}^{q}\ell^{(j)} \le \frac{\sigma L}{N}.$$

bad6: Suppose the first and last vertices on a path inside some component are $\widehat{S}_i^{(j)}$ and $\widehat{S}_{i'}^{(j')}$. Also suppose that the path goes through $x_1, x_2, \cdots, x_y$ vertices of position $i_1, i_2, \cdots, i_y$ respectively. Then this bad event implies

$$C_i^{(j)} + C_{i'}^{(j')} + (\alpha^i + x_1\alpha^{i_1} + \cdots + x_y\alpha^{i_y} + \alpha^{i'})\cdot\mathcal{L} = 0.$$

For a fixed choice of the vertex pair $(\widehat{S}_i^{(j)}, \widehat{S}_{i'}^{(j')})$, the probability of the event comes out to be $1/N$ due to the randomness of $\mathcal{L}$. Applying union bound over all possible vertex pairs, and summing over all components $\mathcal{C}$ of G, we get

$$\begin{aligned}\Pr[\mathsf{bad6}] &\le \sum_{\mathcal{C}}\frac{1}{2N}\cdot\left(\sum_{j\in\mathcal{C}}\ell^{(j)}\right)^2\\ &\le \sum_{\mathcal{C}}\frac{1}{2N}\cdot\xi_{\max}\cdot\sum_{j\in\mathcal{C}}\ell^{(j)} = \frac{\xi_{\max}\sigma}{2N} \le \frac{n\sigma L}{2N}.\end{aligned}$$

Thus, by union-bound, we have

$$\Pr[\mathsf{bad}] \le \frac{4\sigma L}{N} + \frac{\sigma^2 L^2}{N^2} + \frac{2\sigma^3 L^3}{N^3} + \frac{n\sigma L}{2N} \le \frac{n\sigma L}{N},$$

which completes the proof of the lemma. □

Bounding the Ratio of Good Probabilities. Let τ be a good transcript. In the real world, there are q distinct inputs to f_1, q distinct inputs to f_2, and t distinct inputs to π. Thus,

$$\Pr_{\mathcal{O}_1}[\tau] = \frac{1}{N^{2q}(N)_t}.$$

In the ideal world, in the online stage, there are σ outputs that are sampled uniformly at random. In the offline stage, q more values are sampled uniformly, and finally t variables are sampled from the good set subject to r non-redundant equations (we calculate r later). Since $\sigma < N/n^2L^2$, and none of the bad events has happened, the conditions for applying mirror theory are fulfilled. Thus, using mirror theory,

$$\Pr_{\mathcal{O}_0}[\tau] \leq \frac{1}{N^{\sigma+q}} \cdot \frac{N^r}{(N)_t} \leq \frac{1}{N^{\sigma+q-r}(N)_t}.$$

To calculate r, we note that every repeated use of a nonce adds a non-redundant equation to the system. Thus, $r = \sigma - q$, giving us

$$\Pr_{\mathcal{O}_0}[\tau] \leq \frac{1}{N^{2q}(N)_t}.$$

Thus, we have

$$\frac{\Pr_{\mathcal{O}_1}[\tau]}{\Pr_{\mathcal{O}_0}[\tau]} \geq 1,$$

Applying the H-Coefficient Technique with $\epsilon_1 = n\sigma L/N$ and $\epsilon_2 = 0$ completes the proof. □

Appendix E gives a birthday-bound TSPRP proof for OTBC-3.

4 An Application of OTBC-3

Using the tweakable block-cipher OTBC-3, we define an authenticated encryption scheme OCB+ that is about as efficient as OCB3 while providing a higher degree of privacy guarantee without affecting the authenticity guarantee of OCB3. This is shown in Fig. 5.

4.1 Nonce Handling

OCB+ uses a nonce $\mathcal{N}$ of $n-2$ bits, with the final two bits reserved for domain separation. $\mathcal{N}\|00$ is used for processing the message blocks, $\mathcal{N}\|01$ is used for processing the tag, and $\mathcal{N}\|10$ is used for handling the associated data.

4.2 Handling Incomplete Blocks

Incomplete blocks can be handled in the same way as in OCB3, modifying the masking constants for the incomplete blocks. This does not affect the privacy bound significantly, and since the focus of this work is to improve the privacy guarantee of OCB3, we skip giving specific details on how to handle incomplete blocks in OCB+.

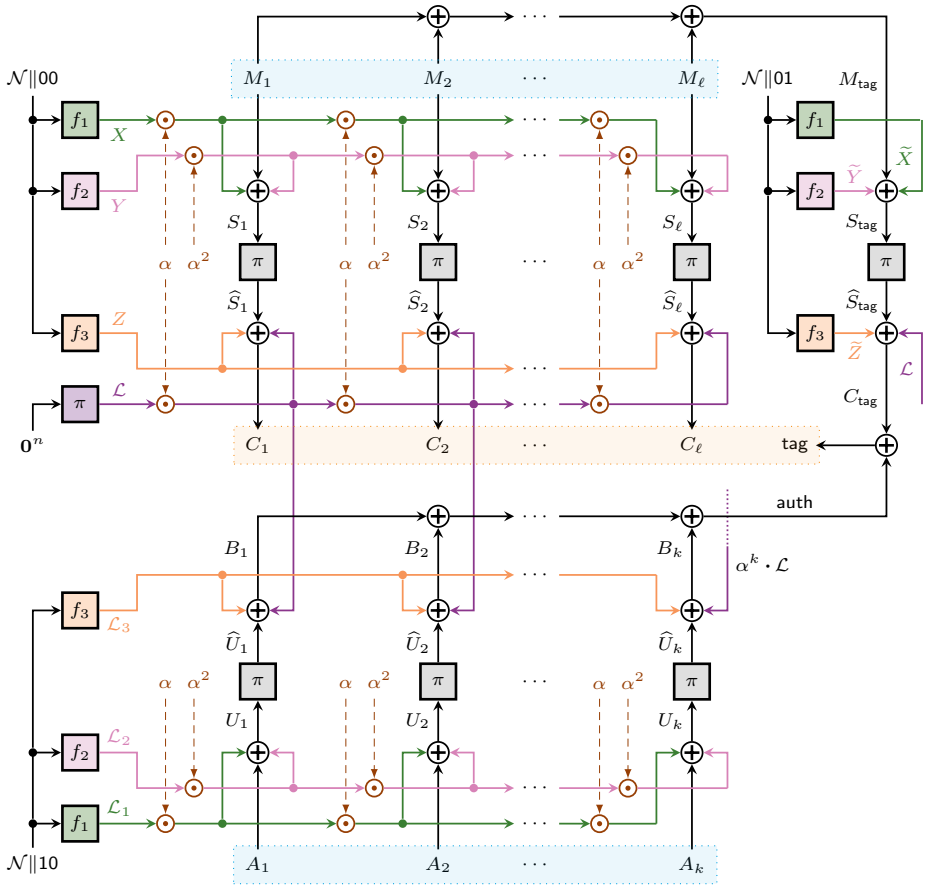

Fig. 5: The OCB+ construction. α is a primitive field-element that allows efficient multiplication.

4.3 Security Claims

We claim that as long as the maximum length L permitted for each message (i.e., the maximum number of blocks encrypted using the same nonce) is small, OCB+ provides both beyond-birthday privacy and beyond-birthday authenticity. Formally we claim the following.

Theorem 5. *Consider a distinguisher $\mathcal{A}$ of OCB+ which can make q encryption queries with distinct nonces with σ blocks and q' decryption queries to its challenger. Suppose the length of the i-th message and the i-th associated data are ℓ_i and k_i respectively, where $\ell_i, k_i \leq L \forall i \in [q_e]$. As long as $\sigma \leq N/n^2L^2$, we have*

$$\boldsymbol{Adv}^{\mathsf{OCB+}}_{\mathsf{NAEAD}}(\mathcal{A}) \leq \frac{n\sigma L}{N} + O\left(\frac{q'L}{N}\right).$$

Algorithm 2 $\mathsf{OCB}+^{f_1,f_2,f_3,\pi}(\mathcal{N}, A, M)$

$M_{\mathsf{tag}} \leftarrow 0^n$
$\mathsf{auth} \leftarrow 0^n$
for $i \leftarrow 1$ to ℓ **do**
    $M_{\mathsf{tag}} \leftarrow M_{\mathsf{tag}} \oplus M_i$
    $C_i \leftarrow \mathsf{OTBC\text{-}3}^{f_1,f_2,f_3,\pi}(\mathcal{N}\|\mathsf{00}, i, M_i)$
end for
$C \leftarrow C_1\| \cdots \|C_\ell$
$C_{\mathsf{tag}} \leftarrow \mathsf{OTBC\text{-}3}^{f_1,f_2,f_3,\pi}(\mathcal{N}\|\mathsf{01}, 0, M_{\mathsf{tag}})$
for $i \leftarrow 1$ to k **do**
    $B_i \leftarrow \mathsf{OTBC\text{-}3}^{f_1,f_2,f_3,\pi}(\mathcal{N}\|\mathsf{10}, i, A_i)$
    $\mathsf{auth} \leftarrow \mathsf{auth} \oplus B_i$
end for
$\mathsf{tag} \leftarrow C_{\mathsf{tag}} \oplus \mathsf{auth}$
$T \leftarrow \mathsf{chop}_\tau(\mathsf{tag})$
return (C, T)

Proof. Suppose there is a distinguisher $\mathcal{B}$ of $\mathsf{OTBC\text{-}3}$ which can make $\sigma + q$ queries to its challenger and which works in the following way. It runs $\mathcal{A}$ to start the game. Whenever $\mathcal{A}$ makes the i-th encryption query $(\mathcal{N}^i, A^i, M^i)$, $\mathcal{B}$ does the following.

- For the j-th message block M^i_j, it makes the encryption query$(\mathcal{N}^i\|\mathsf{00}, j, M^i_j)$ to its challenger. Suppose it receives C^i_j as the response.
- Suppose the length of M^i is ℓ_i blocks. It makes and encryption query $(\mathcal{N}^i\|\mathsf{01}, 0, M^i_1 + \cdots + M^i_{\ell_i})$ to its challenger. Suppose it receives C^i_{tag} as response.
- For the j-th associated data block A^i_j, it makes the encryption query $(\mathcal{N}^i\|\mathsf{10}, j, A^i_j)$ to its challenger. Suppose it receives B^i_j as response.
- Suppose the length of A^i is k_i blocks. It calculates $\mathsf{auth}^i = B^i_1 + \cdots + B^i_{k_i}$.
- Finally it returns $(C^i_1\| \cdots \|C^i_{\ell_i}, \mathsf{chop}_\tau(C^i_{\mathsf{tag}} + \mathsf{auth}^i))$ to $\mathcal{A}$.

Once $\mathcal{A}$ submits its decision bit, $\mathcal{B}$ carries it forward to its challenger as its own decision bit as well. Then we obtain the following privacy advantage of $\mathcal{A}$:

$$\mathbf{Adv}^{\mathsf{OCB}+}_{\mathsf{priv}}(\mathcal{A}) = \mathbf{Adv}^{\mathsf{OTBC\text{-}3}}_{\mathrm{TPRP}^*}(\mathcal{B}).$$

Combining this result with Theorem 4, we obtain

$$\mathbf{Adv}^{\mathsf{OCB}+}_{\mathsf{priv}}(\mathcal{A}) \leq \frac{n\sigma L}{N}. \tag{1}$$

From the security analysis in Section 4 of [15], we obtain the following authenticity advantage of $\mathcal{A}$.

$$\mathbf{Adv}^{\mathsf{OCB}+}_{\mathsf{auth}}(\mathcal{A}) \leq O\left(\frac{q'L}{N}\right). \tag{2}$$

The result of Theorem 5 follows directly from (1) and (2). □

References

1. CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness. https://competitions.cr.yp.to/caesar-submissions.html.
2. Information technology – Security techniques – Authenticated encryption. ISO/IEC 19772:2009, 2009.
3. NIST Lightweight Cryptography. https://csrc.nist.gov/Projects/lightweight-cryptography.
4. Kazumaro Aoki and Kan Yasuda. The security of the OCB mode of operation without the sprp assumption. In Willy Susilo and Reza Reyhanitabar, editors, *Provable Security*, pages 202–220, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.
5. Tomer Ashur, Orr Dunkelman, and Atul Luykx. Boosting authenticated encryption robustness with minimal modifications. In Jonathan Katz and Hovav Shacham, editors, *Advances in Cryptology – CRYPTO 2017*, pages 3–33, Cham, 2017. Springer International Publishing.
6. Roberto Avanzi. The QARMA block cipher family. almost mds matrices over rings with zero divisors, nearly symmetric even-mansour constructions with non-involutory central rounds, and search heuristics for low-latency s-boxes. *IACR Transactions on Symmetric Cryptology*, 2017(1):4–44, Mar. 2017.
7. Zhenzhen Bao, Jian Guo, Tetsu Iwata, and Kazuhiko Minematsu. ZOCB and ZOTR: Tweakable blockcipher modes for authenticated encryption with full absorption. *IACR Transactions on Symmetric Cryptology*, 2019(2):1–54, Jun. 2019.
8. Christof Beierle, Jérémy Jean, Stefan Kölbl, Gregor Leander, Amir Moradi, Thomas Peyrin, Yu Sasaki, Pascal Sasdrich, and Siang Meng Sim. The skinny family of block ciphers and its low-latency variant mantis. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology – CRYPTO 2016*, pages 123–153, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.
9. Christof Beierle, Jérémy Jean, Stefan Kölbl, Gregor Leander, Amir Moradi, Thomas Peyrin, Yu Sasaki, Pascal Sasdrich, and Siang Meng Sim. SKINNY-AEAD and SKINNY-Hash. *IACR Transactions on Symmetric Cryptology*, 2020(S1):88–131, Jun. 2020.
10. Christof Beierle, Gregor Leander, Amir Moradi, and Shahram Rasoolzadeh. Craft: Lightweight tweakable block cipher with efficient protection against dfa attacks. *IACR Transactions on Symmetric Cryptology*, 2019(1):5–45, Mar. 2019.
11. Mihir Bellare and Chanathip Namprempre. Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm. J Cryptol 21, 469–491 (2008). https://doi.org/10.1007/s00145-008-9026-x.
12. Mihir Bellare and Chanathip Namprempre. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In Tatsuaki Okamoto, editor, *Advances in Cryptology — ASIACRYPT 2000*, pages 531–545, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg.
13. Srimanta Bhattacharya and Mridul Nandi. Revisiting variable output length xor pseudorandom function. *IACR Transactions on Symmetric Cryptology*, 2018(1):314–335, Mar. 2018.
14. Ritam Bhaumik, Xavier Bonnetain, André Chailloux, Gaëtan Leurent, María Naya-Plasencia, André Schrottenloher, and Yannick Seurin. QCB: Efficient quantum-secure authenticated encryption. In Mehdi Tibouchi and Huaxiong Wang, editors, *Advances in Cryptology – ASIACRYPT 2021*, pages 668–698, Cham, 2021. Springer International Publishing.

15. Ritam Bhaumik and Mridul Nandi. Improved security for OCB3. In Tsuyoshi Takagi and Thomas Peyrin, editors, *Advances in Cryptology – ASIACRYPT 2017*, pages 638–666, Cham, 2017. Springer International Publishing.
16. Avik Chakraborti, Nilanjan Datta, Ashwin Jha, Cuauhtemoc Mancillas-López, and Mridul Nandi. Light-OCB: Parallel lightweight authenticated cipher with full security. In Lejla Batina, Stjepan Picek, and Mainack Mondal, editors, *Security, Privacy, and Applied Cryptography Engineering*, pages 22–41, Cham, 2022. Springer International Publishing.
17. Avik Chakraborti, Nilanjan Datta, Ashwin Jha, Cuauhtemoc Mancillas-López, Mridul Nandi, and Yu Sasaki. Int-rup secure lightweight parallel ae modes. *IACR Transactions on Symmetric Cryptology*, 2019(4):81–118, Jan. 2020.
18. Shan Chen and John Steinberger. Tight security bounds for key-alternating ciphers. In Phong Q. Nguyen and Elisabeth Oswald, editors, *Advances in Cryptology – EUROCRYPT 2014*, pages 327–350, Berlin, Heidelberg, 2014. Springer Berlin Heidelberg.
19. Benoît Cogliati, Avijit Dutta, Mridul Nandi, Jacques Patarin, and Abishanka Saha. Proof of mirror theory for any $\xi_{\max}$. *IACR Cryptol. ePrint Arch.*, page 686, 2022.
20. Wei Dai, Viet Tung Hoang, and Stefano Tessaro. Information-theoretic indistinguishability via the chi-squared method. In Jonathan Katz and Hovav Shacham, editors, *Advances in Cryptology – CRYPTO 2017*, pages 497–523, Cham, 2017. Springer International Publishing.
21. Avijit Dutta, Mridul Nandi, and Abishanka Saha. Proof of mirror theory for $\xi_{\max} = 2$. *IEEE Trans. Inf. Theory*, 68(9):6218–6232, 2022.
22. Niels Ferguson, Stefan Lucks, Bruce Schneier, Doug Whiting, Mihir Bellare, Tadayoshi Kohno, Jon Callas, and Jesse Walker. The Skein Hash Function Family. SHA3 submission to NIST (Round 3), 2010.
23. Robert Granger, Philipp Jovanovic, Bart Mennink, and Samuel Neves. Improved masking for tweakable blockciphers with applications to authenticated encryption. In Marc Fischlin and Jean-Sébastien Coron, editors, *Advances in Cryptology – EUROCRYPT 2016*, pages 263–293, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.
24. Vincent Grosso, Gaëtan Leurent, François-Xavier Standaert, Kerem Varici, Anthony Journault, François Durvaux, Lubos Gaspar, and Stéphanie Kerckhof. SCREAM v3. Submission to CAESAR competition, 2015.
25. Shay Gueron, Adam Langley, and Yehuda Lindell. AES-GCM-SIV: Nonce Misuse-Resistant Authenticated Encryption. RFC 8452, DOI 10.17487/RFC8452, April 2019, https://www.rfc-editor.org/info/rfc8452.
26. Shay Gueron and Yehuda Lindell. GCM-SIV: Full Nonce Misuse-Resistant Authenticated Encryption at Under One Cycle per Byte. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, CCS '15, page 109–119, New York, NY, USA, 2015. Association for Computing Machinery.
27. Viet Tung Hoang and Stefano Tessaro. Key-alternating ciphers and key-length extension: Exact bounds and multi-user security. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology – CRYPTO 2016*, pages 3–32, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.
28. Akiko Inoue, Tetsu Iwata, Kazuhiko Minematsu, and Bertram Poettering. Cryptanalysis of OCB2: Attacks on Authenticity and Confidentiality. J Cryptol 33, 1871–1913 (2020). https://doi.org/10.1007/s00145-020-09359-8.

29. Akiko Inoue, Tetsu Iwata, Kazuhiko Minematsu, and Bertram Poettering. Cryptanalysis of OCB2: Attacks on Authenticity and Confidentiality. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology – CRYPTO 2019*, pages 3–31, Cham, 2019. Springer International Publishing.
30. Tetsu Iwata. New Blockcipher Modes of Operation with Beyond the Birthday Bound Security. In Matthew Robshaw, editor, *Fast Software Encryption*, pages 310–327, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.
31. Tetsu Iwata. Authenticated Encryption Mode for Beyond the Birthday Bound Security. In Serge Vaudenay, editor, *Progress in Cryptology – AFRICACRYPT 2008*, pages 125–142, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.
32. Jérémy Jean, Ivica Nikolić, and Thomas Peyrin. Tweaks and keys for block ciphers: The tweakey framework. In Palash Sarkar and Tetsu Iwata, editors, *Advances in Cryptology – ASIACRYPT 2014*, pages 274–288, Berlin, Heidelberg, 2014. Springer Berlin Heidelberg.
33. Jérémy Jean, Ivica Nikolic, and Thomas Peyrin. Joltik v1.3. CAESAR Round, 2, 2015.
34. Jérémy Jean, Ivica Nikolic, Thomas Peyrin, and Yannick Seurin. The Deoxys AEAD family. *J. Cryptol.*, 34(3):31, 2021.
35. Charanjit S. Jutla. Encryption Modes with Almost Free Message Integrity. In Birgit Pfitzmann, editor, *Advances in Cryptology — EUROCRYPT 2001*, pages 529–544, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.
36. C.S. Jutla. Encryption Modes with Almost Free Message Integrity. J Cryptol 21, 547–578 (2008). https://doi.org/10.1007/s00145-008-9024-z.
37. Jonathan Katz and Moti Yung. Unforgeable encryption and chosen ciphertext secure modes of operation. In Gerhard Goos, Juris Hartmanis, Jan van Leeuwen, and Bruce Schneier, editors, *Fast Software Encryption*, pages 284–299, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.
38. T. Krovetz and P. Rogaway. The OCB Authenticated-Encryption Algorithm. RFC 7253, DOI 10.17487/RFC7253, May 2014, https://www.rfc-editor.org/info/rfc7253.
39. Ted Krovetz and Phillip Rogaway. The Software Performance of Authenticated-Encryption Modes. In Antoine Joux, editor, *Fast Software Encryption*, pages 306–327, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.
40. Moses Liskov, Ronald L. Rivest, and David Wagner. Tweakable Block Ciphers. J Cryptol 24, 588–613 (2011). https://doi.org/10.1007/s00145-010-9073-y.
41. Moses Liskov, Ronald L. Rivest, and David Wagner. Tweakable Block Ciphers. In Moti Yung, editor, *Advances in Cryptology — CRYPTO 2002*, pages 31–46, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.
42. Kazuhiko Minematsu. Parallelizable Rate-1 Authenticated Encryption from Pseudorandom Functions. In Phong Q. Nguyen and Elisabeth Oswald, editors, *Advances in Cryptology – EUROCRYPT 2014*, pages 275–292, Berlin, Heidelberg, 2014. Springer Berlin Heidelberg.
43. Yusuke Naito. Tweakable blockciphers for efficient authenticated encryptions with beyond the birthday-bound security. *IACR Transactions on Symmetric Cryptology*, 2017(2):1–26, Jun. 2017.
44. Jacques Patarin. The "coefficients h" technique. In Roberto Maria Avanzi, Liam Keliher, and Francesco Sica, editors, *Selected Areas in Cryptography*, pages 328–345, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.
45. Thomas Peyrin and Yannick Seurin. Counter-in-Tweak: Authenticated Encryption Modes for Tweakable Block Ciphers. In Matthew Robshaw and Jonathan Katz,

editors, *Advances in Cryptology – CRYPTO 2016*, pages 33–63, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.
46. Phillip Rogaway. Authenticated-Encryption with Associated-Data. In *Proceedings of the 9th ACM Conference on Computer and Communications Security*, CCS '02, page 98–107, New York, NY, USA, 2002. Association for Computing Machinery.
47. Phillip Rogaway. Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC. In Pil Joong Lee, editor, *Advances in Cryptology - ASIACRYPT 2004*, pages 16–31, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.
48. Phillip Rogaway, Mihir Bellare, and John Black. OCB: A Block-Cipher Mode of Operation for Efficient Authenticated Encryption. *ACM Trans. Inf. Syst. Secur.*, 6(3):365–403, aug 2003.
49. Phillip Rogaway, Mihir Bellare, John Black, and Ted Krovetz. OCB: A Block-Cipher Mode of Operation for Efficient Authenticated Encryption. In *Proceedings of the 8th ACM Conference on Computer and Communications Security*, CCS '01, page 196–205, New York, NY, USA, 2001. Association for Computing Machinery.
50. Phillip Rogaway and Thomas Shrimpton. A provable-security treatment of the key-wrap problem. In Serge Vaudenay, editor, *Advances in Cryptology - EUROCRYPT 2006*, pages 373–390, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.
51. Richard Schroeppel. The Hasty Pudding Cipher. AES submission to NIST, 1998.
52. John Steinberger Shan Chen. Tight security bounds for key-alternating ciphers. Cryptology ePrint Archive, Report 2013/222, 2013. https://ia.cr/2013/222.
53. Ping Zhang, Peng Wang, and Honggang Hu. The int-rup security of ocb with intermediate (parity) checksum. Cryptology ePrint Archive, Report 2016/1059, 2016. https://ia.cr/2016/1059.
54. Ping Zhang, Peng Wang, Honggang Hu, Changsong Cheng, and Wenke Kuai. Int-rup security of checksum-based authenticated encryption. In Tatsuaki Okamoto, Yong Yu, Man Ho Au, and Yannan Li, editors, *Provable Security*, pages 147–166, Cham, 2017. Springer International Publishing.

Appendix

A Attack on OTBC-0

Let the i-th nonce be N_i, and the corresponding offset and ciphertext for the message M_i be X_i and C_i respectively. We get

$$\pi(M_i + X_i) + X_i = C_i.$$

For two distinct queries (say i-th and j-th query with $i \neq j$), we get

$$\pi(M_i + X_i) + \pi(M_j + X_j) + X_i + X_j = C_i + C_j.$$

So, whenever $M_i + X_i = M_j + X_j$, we have $M_i + C_i = M_j + C_j$. We call this event E and consider it as a distinguishing event. We define an event E_1 in which $M_i + X_i = M_j + X_j$ for some $i \neq j$. From the above discussion we see that

$$E_1 \Rightarrow E.$$

Let $\mathsf{cp}(q, N)$ denote the probability of finding a collision-pair in a uniform random sample of size q from a population of size N. Thus,

$$\Pr_{\mathcal{O}_1}[E_1] = \mathsf{cp}(q, N).$$

We also have

$$\Pr_{\mathcal{O}_1}[E] = \mathsf{cp}(q, N) + \Pr_{\mathcal{O}_1}[E \mid \neg E_1] \cdot (1 - \mathsf{cp}(q, N)).$$

Now, when E_1 does not happen, all inputs of π must be distinct. Let us denote the inputs and output of π for the ith query by S_i and $\widehat{S}_i$ respectively. Then the event E is equivalent to finding a collision-pair among the $S_i + \widehat{S}_i$ values. Given that E_1 does not happen both S_i's and $\widehat{S}_i$'s are sampled as uniformly without replacement and $(S_i)_{i\in[q]}$ is independent from $(\widehat{S}_i)_{i\in[q]}$. By using well known result [13,20], we know that the sum of independent without-replacement samples is almost the identically distributed as a uniform random sample and hence

$$\Pr_{\mathcal{O}_1}[E \mid \neg E_1] \approx \mathsf{cp}(q, N).$$

Thus, we have

$$\Pr_{\mathcal{O}_1}[E] \approx \mathsf{cp}(q, N) + (1 - \mathsf{cp}(q, N)) \cdot \mathsf{cp}(q, N).$$

On the other hand in the ideal world,

$$\Pr_{\mathcal{O}_0}(E) = \mathsf{cp}(q, N),$$

as for every distinct nonce the responses should be uniformly and independently distributed. Hence, the distinguishing advantage is around $\mathsf{cp}(q, N)(1 - \mathsf{cp}(q, N))$. Now we know that $\mathsf{cp}(q, N) = 1/2$ is attained for a $q = O(\sqrt{N})$ and hence for that choice of q, the distinguishing advantage is at least $1/4$.

B Proof of Theorem 2

Theorem 3. *Consider an adversary $\mathcal{A}$, playing a q-query distinguishing game between* OTBC-1 *and an ideal tweakable permutation* OTBC-id. *As long as the pair (nonce, block-number) is never repeated, we have*

$$\boldsymbol{Adv}_{TPRP}^{\mathsf{OTBC\text{-}1},\mathsf{OTBC\text{-}id}}(\mathcal{A}) = 0.$$

Proof. Let's call $(\mathcal{N}, i)$ as $\mathcal{T}$. We'll use Coefficients H Technique to bound the advantage of the adversary.

Transcript Notation. The adversary makes q encryption queries $(\mathcal{T}^1, M^1), \cdots, (\mathcal{T}^q, m^q)$ to the oracle, and receives $C^1, \cdots C^q$ as the corresponding responses. So the query-response transcript of the adversary initially looks like $\{(\mathcal{T}^1, M^1, C^1), \cdots, (\mathcal{T}^q, M^q, C^q)\}$.

Sampling in the Ideal World. For each encryption query, the ideal oracle samples the output with replacement from $\{0,1\}^n$ uniformly at random. Once the adversary is done with all its queries, the oracle releases the some additional information to the adversary. The ideal oracle samples them in the following way:

- For all $j \in [q]$, the ideal oracle samples X^j with replacement from $\{0,1\}^n$ uniformly at random.
- For all $j \in [q]$, the ideal oracle samples $\widehat{S}^j$ without replacement from $\{0,1\}^n$ uniformly at random.

The real oracle releases the corresponding true values in this additional release phase. After the additional release, the extended transcript looks like $\{(\mathcal{T}^1, M^1, C^1, X^1, \widehat{S}^1), \cdots, (\mathcal{T}^q, M^q, C^q, X^q, \widehat{S}^q)\}$.

Advantage of the Adversary. For any attainable transcript τ, we get the real interpolation probability as

$$\Pr_{\mathcal{O}_1}[\tau] = \frac{1}{N^q} \cdot \frac{1}{N^q} \cdot \frac{1}{(N)_q}\,.$$

The first, second and third term in the denominator on the right hand side represents the number of choices for X, Y and $\widehat{S}$ respectively. We also get the ideal interpolation probability as

$$\Pr_{\mathcal{O}_0}[\tau] = \frac{1}{N^q} \cdot \frac{1}{N^q} \cdot \frac{1}{(N)_q}\,.$$

The first, second and third term in the denominator on the right hand side represents the number of choices for C, X and $\widehat{S}$ respectively. Thus we finally we get

$$\frac{\Pr_{\mathcal{O}_1}[\tau]}{\Pr_{\mathcal{O}_0}[\tau]} = 1\,.$$

Applying H-Coefficient Technique with $\epsilon_1 = \epsilon_2 = 0$ completes the proof. □

C Proof of Theorem 3

Theorem 4. *There exists a distinguisher $\mathcal{A}$ querying with q nonces and L blocks under each nonce with $L \geq 12$ in a TPRP game against* OTBC-2 *such that*

$$\boldsymbol{Adv}^{\mathsf{OTBC\text{-}2},\mathsf{TBC\text{-}id}}_{TPRP} \geq \Omega\left(\frac{q^2L^2}{N}\right).$$

Proof. $\mathcal{A}$ picks q distinct nonces $\mathcal{N}^{(1)}, \ldots, \mathcal{N}^{(q)}$, and q distinct starting messages $M_1^{(1)}, \ldots, M_1^{(q)}$. For each $j \in [q]$ it makes L queries $(\mathcal{N}^{(j)}, 1, M_1^{(j)}), \ldots, (\mathcal{N}^{(j)}, L, M_L^{(j)})$, such that for each $i \in [L]$ we have

$$M_i^{(j)} := \alpha^{i-1} \cdot M_1^{(j)}.$$

This ensures that we have

$$\begin{aligned} S_i^{(j)} &:= M_i^{(j)} + \alpha^i \cdot X^{(j)} \\ &= \alpha^{i-1} \cdot M_1^{(j)} + \alpha^i \cdot X^{(j)} \\ &= \alpha^{i-1} \cdot \left(M_1^{(j)} + \alpha \cdot X^{(j)} \right) = \alpha^{i-1} \cdot S_1^{(j)}, \end{aligned}$$

where $X^{(j)} := f_1(\mathcal{N}^{(j)})$ and $S_i^{(j)}$ is the input of π on the query $(\mathcal{N}^{(j)}, i, M_i^{(j)})$.

Input Collision. Suppose we have distinct $j, j' \in [q]$ and some $i, i' \in [L-1]$ (not necessarily distinct), such that $S_i^{(j)} = S_{i'}^{(j')}$. Then we have

$$\begin{aligned} S_{i+1}^{(j)} = \alpha^i \cdot S_1^{(j)} &= \alpha \cdot \left(\alpha^{i-1} \cdot S_1^{(j)} \right) \\ &= \alpha \cdot S_i^{(j)} \\ &= \alpha \cdot S_{i'}^{(j')} \\ &= \alpha \cdot \left(\alpha^{i-1} \cdot S_1^{(j')} \right) = \alpha^i \cdot S_1^{(j')} = S_{i'+1}^{(j')}. \end{aligned}$$

In other words, a collision on two input blocks in two different nonces forces a collision on the next block as well (and, in fact, this dominoes into all successive blocks till one of the block-numbers reach L). $\mathcal{A}$ can use this property to mount the distinguishing attack.

Distinguishing Event. $\mathcal{A}$ searches for a pair of distinct $j, j' \in [q]$ and $i, i' \in [L-2]$ (not necessarily distinct) such that

$$C_{i+2}^{(j)} + C_{i'+2}^{(j')} = \alpha \cdot \left(C_{i+1}^{(j)} + C_{i'+1}^{(j')} \right) = \alpha^2 \cdot \left(C_i^{(j)} + C_{i'}^{(j')} \right).$$

If such j, j', i, i' exist, $\mathcal{A}$ outputs 1, else it outputs 0.

We note that in the real world, whenever $S_i^{(j)} = S_{i'}^{(j')}$, we have $\widehat{S}_i^{(j)} = \widehat{S}_{i'}^{(j')}$, which implies that

$$C_i^{(j)} + C_{i'}^{(j')} = \alpha^i \cdot Y^{(j)} + \alpha^{i'} \cdot Y^{(j')}.$$

From the above discussion, we know that $S_i^{(j)} = S_{i'}^{(j')}$ forces the collisions $S_{i+1}^{(j)} = S_{i'+1}^{(j')}$ and $S_{i+2}^{(j)} = S_{i'+2}^{(j')}$. The first of these implies that

$$\begin{aligned} C_{i+1}^{(j)} + C_{i'+1}^{(j')} &= \alpha^{i+1} \cdot Y^{(j)} + \alpha^{i'+1} \cdot Y^{(j')} \\ &= \alpha \cdot \left(\alpha^i \cdot Y^{(j)} + \alpha^{i'} \cdot Y^{(j')} \right) \\ &= \alpha \cdot \left(C_i^{(j)} + C_{i'}^{(j')} \right), \end{aligned}$$

and similarly the second implies that

$$C_{i+2}^{(j)} + C_{i'+2}^{(j')} = \alpha \cdot \left(C_{i+1}^{(j)} + C_{i'+1}^{(j')} \right) = \alpha^2 \cdot \left(C_i^{(j)} + C_{i'}^{(j')} \right).$$

Thus, the collision $S_i^{(j)} = S_{i'}^{(j')}$ for distinct $j, j' \in [q]$ and $i, i' \in [L-2]$ is enough to trigger the distinguishing event.

In the ideal world, this event require two collisions, each with probability $1/N$. Since there are $q(q-1)/2$ choices for j, j' and $(L-2)^2$ choices for i, i', we have

$$\Pr_{\mathcal{O}_0}[\mathcal{A} \text{ outputs } 1] \approx \binom{q}{2} \cdot \frac{(L-2)^2}{N^2}.$$

But in the real world, this only requires one collision, as the other is automatically enforced. Thus,

$$\Pr_{\mathcal{O}_1}[\mathcal{A} \text{ outputs } 1] \approx \binom{q}{2} \cdot \frac{(L-2)^2}{N}.$$

This completes the proof of the claimed lower bound on the advantage of $\mathcal{A}$. □

D The Proof of Lemma 1

Lemma 1. *For some $r \geq 2$ and $2r$ numbers $i_1, i'_1, \ldots, i_r, i'_r < N$ such that $i_j \neq i'_j$ for each $j \in [r]$, define*

$$\mathbf{B}_r = \begin{bmatrix} \alpha^{i_1} & \alpha^{2i_1} & \alpha^{i'_2} & \alpha^{2i'_2} & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & \alpha^{i_2} & \alpha^{2i_2} & \alpha^{i'_3} & \alpha^{2i'_3} & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \alpha^{i_3} & \alpha^{2i_3} & \alpha^{i'_4} & \alpha^{2i'_4} & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & \alpha^{i_{r-1}} & \alpha^{2i_{r-1}} & \alpha^{i'_r} & \alpha^{2i'_r} \\ \alpha^{i'_1} & \alpha^{2i'_1} & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & \alpha^{i_r} & \alpha^{2i_r} \end{bmatrix}.$$

Then $\mathbf{B}_r$ is at least of rank r.

Proof. First we observe that $\mathbf{B}_2$ is of rank 2 since, for the leftmost 2×2 submatrix of $\mathbf{B}_2$, we have

$$\begin{vmatrix} \alpha^{i_1} & \alpha^{2i_1} \\ \alpha^{i'_1} & \alpha^{2i'_1} \end{vmatrix} = \alpha^{i_1+2i'_1} + \alpha^{2i_1+i'_1} = \alpha^{i_1+i'_1}(\alpha^{i_1} + \alpha^{i'_1}) \neq 0.$$

(This also holds for the rightmost 2×2 submatrix.) Next we observe that $\mathbf{B}_3$ is of rank 3 since, for the leftmost 3×3 submatrix of $\mathbf{B}_3$, we have

$$\begin{vmatrix} \alpha^{i_1} & \alpha^{2i_1} & \alpha^{i'_2} \\ 0 & 0 & \alpha^{i_2} \\ \alpha^{i'_1} & \alpha^{2i'_1} & 0 \end{vmatrix} = \alpha^{i_2} \begin{vmatrix} \alpha^{i_1} & \alpha^{2i_1} \\ \alpha^{i'_1} & \alpha^{2i'_1} \end{vmatrix} = \alpha^{i_1+i'_1+i_2}(\alpha^{i_1} + \alpha^{i'_1}) \neq 0.$$

(This also holds for any of the three other contiguous 3×3 submatrices of $\mathbf{B}$.) This leaves the case $r \geq 4$. We consider two cases, based on whether r is even

or odd. First, suppose $r = 2m$. Then we look at the $2m \times 2m$ submatrix $\mathbf{H}$ of $\mathbf{B}_r$ consisting of the columns $4p-1$ and $4p$ for each $p \in [m]$. Thus,

$$\mathbf{H} = \begin{bmatrix} \alpha^{i'_2} & \alpha^{2i'_2} & 0 & 0 & \cdots & 0 & 0 \\ \alpha^{i_2} & \alpha^{2i_2} & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \alpha^{i'_4} & \alpha^{2i'_4} & \cdots & 0 & 0 \\ 0 & 0 & \alpha^{i_4} & \alpha^{2i_4} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \alpha^{i'_{2m}} & \alpha^{2i'_{2m}} \\ 0 & 0 & 0 & 0 & \cdots & \alpha^{i_{2m}} & \alpha^{2i_{2m}} \end{bmatrix}.$$

We observe that $\mathbf{H}$ is a block-diagonal matrix of the form

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 & \mathbf{0}_{2\times 2} & \cdots & \mathbf{0}_{2\times 2} \\ \mathbf{0}_{2\times 2} & \mathbf{H}_2 & \cdots & \mathbf{0}_{2\times 2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{2\times 2} & \mathbf{0}_{2\times 2} & \cdots & \mathbf{H}_m \end{bmatrix},$$

where for each $p \in [m]$,

$$\mathbf{H}_p = \begin{bmatrix} \alpha^{i'_{2p}} & \alpha^{2i'_{2p}} \\ \alpha^{i_{2p}} & \alpha^{2i_{2p}} \end{bmatrix}.$$

Thus, $|\mathbf{H}_p| = \alpha^{i_{2p}+i'_{2p}}(\alpha^{i_{2p}} + \alpha^{i'_{2p}}) \neq 0$ for each $p \in [m]$, and

$$|\mathbf{H}| = |\mathbf{H}_1| \cdot |\mathbf{H}_2| \cdot \ldots \cdot |\mathbf{H}_m| \neq 0,$$

which shows that $\mathbf{H}$ (and thus $\mathbf{B}_r$) is of rank $2m$. Next suppose $r = 2m+1$. We consider the $(m+1) \times (m+1)$ submatrix $\mathbf{H}$ of $\mathbf{B}_r$ consisting of columns $4p-1$ and $4p$ for each $p \in [m]$, as well as column $4m+1$. Thus,

$$\mathbf{H} = \begin{bmatrix} \alpha^{i'_2} & \alpha^{2i'_2} & 0 & 0 & \cdots & 0 & 0 & 0 \\ \alpha^{i_2} & \alpha^{2i_2} & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & \alpha^{i'_4} & \alpha^{2i'_4} & \cdots & 0 & 0 & 0 \\ 0 & 0 & \alpha^{i_4} & \alpha^{2i_4} & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \alpha^{i'_{2m}} & \alpha^{2i'_{2m}} & 0 \\ 0 & 0 & 0 & 0 & \cdots & \alpha^{i_{2m}} & \alpha^{2i_{2m}} & \alpha^{i'_{2m+1}} \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & \alpha^{i_{2m+1}} \end{bmatrix}.$$

Again, we observe that $\mathbf{H}$ is a block-diagonal matrix of the form

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 & \mathbf{0}_{2\times 2} & \cdots & \mathbf{0}_{2\times 3} \\ \mathbf{0}_{2\times 2} & \mathbf{H}_2 & \cdots & \mathbf{0}_{2\times 3} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{3\times 2} & \mathbf{0}_{3\times 2} & \cdots & \mathbf{H}_m \end{bmatrix},$$

where for each $p \in [m-1]$,

$$\mathbf{H}_p = \begin{bmatrix} \alpha^{i'_{2p}} & \alpha^{2i'_{2p}} \\ \alpha^{i_{2p}} & \alpha^{2i_{2p}} \end{bmatrix},$$

and

$$\mathbf{H}_m = \begin{bmatrix} \alpha^{i'_{2m}} & \alpha^{2i'_{2m}} & 0 \\ \alpha^{i_{2m}} & \alpha^{2i_{2m}} & \alpha^{i'_{2m+1}} \\ 0 & 0 & \alpha^{i_{2m+1}} \end{bmatrix}.$$

We've already seen that $|\mathbf{H}_p| \neq 0$ for each $p \in [m-1]$. Further, we see that $|\mathbf{H}_m| = \alpha^{i_{2m}+i'_{2m}+i_{2m+1}}(\alpha^{i_{2m}} + \alpha^{i'_{2m}}) \neq 0$. Thus,

$$|\mathbf{H}| = |\mathbf{H}_1| \cdot |\mathbf{H}_2| \cdot \ldots \cdot |\mathbf{H}_m| \neq 0,$$

which shows that $\mathbf{H}$ (and thus $\mathbf{B}_r$) is of rank $2m+1$.

E TSPRP Security Analysis of OTBC-3

Let's call $(\mathcal{N}, i)$ as $\mathcal{T}$. We'll use Coefficients H Technique to bound the advantage of the adversary.

Transcript Notation. The adversary makes encryption queries $(\mathcal{T}^{(j)}, M^{(j)})$ to the oracle to receive $C^{(j)}$ and decryption queries $(\mathcal{T}^{(j')}, C^{(j')})$ to the oracle to receive $K^{(j')}$ with $j, j' \in [\sigma]$ and $j \neq j'$. So the query-response transcript of the adversary initially looks like $\{(\mathcal{T}^{(1)}, M^{(1)}, C^{(1)}), \cdots, (\mathcal{T}^{(\sigma)}, M^{(\sigma)}, C^{(\sigma)})\}$.

Sampling in the Ideal World. For each encryption query $(\mathcal{T}^{(j)}, M^{(j)})$, the ideal oracle samples $C^{(j)}$ with replacement from $\{0,1\}^n$ uniformly at random. Similarly, for each decryption query $(\mathcal{T}^{(j')}, C^{(j')})$, the ideal oracle samples $M^{(j')}$ with replacement from $\{0,1\}^n$ uniformly at random. Once the adversary is done with all its queries, the oracle releases the some additional information to the adversary. The ideal oracle samples them in the following way:

- The ideal oracle samples $\mathcal{L}$ from $\{0,1\}^n$ uniformly at random.
- For all $j \in [\sigma]$, the ideal oracle samples $X^{(j)}, Y^{(j)}$ and $Z^{(j)}$ with replacement from $\{0,1\}^n$ uniformly at random.

The real oracle releases the corresponding true values in this additional release phase. After the additional release, the extended transcript looks like $\{\mathcal{L}, (\mathcal{T}^{(1)}, M^{(1)}, C^{(1)}, X^{(1)}, Y^{(1)}, Z^{(1)}), \cdots, (\mathcal{T}^{(q)}, M^{(q)}, C^{(q)}, X^{(q)}, Y^{(q)}, Z^{(q)})\}$.

Bad Events and Their Probabilities. We identify the following events as bad.

bad1: $\exists j, j' \in [\sigma]$ with $j \neq j'$ such that $S^{(j)} = S^{(j')}$. The probability of this event can be bounded by (σ^2/N) due to the randomness of X or Y.

bad2: $\exists j, j' \in [\sigma]$ with $j \neq j'$ such that $\widehat{S}^{(j)} = \widehat{S}^{(j')}$. The probability of this event can also be bounded by (σ^2/N) due to the randomness of X or Y.

bad3: $\exists j \in [\sigma]$ such that $S^{(j)} = 0^n$. The probability of this event can be bounded by (σ/N) due to the randomness of X or Y.

bad4: $\exists j \in [\sigma]$ such that $\widehat{S}^{(j)} = \mathcal{L}$. The probability of this event can also be bounded by (σ/N) due to the randomness of X or Y.

Good Interpolation Probabilities and Their Ratio. For any good transcript τ, we get the real interpolation probability as

$$\Pr_{\mathcal{O}_1}[\tau] = \frac{1}{N^\sigma} \cdot \frac{1}{N^\sigma} \cdot \frac{1}{N^\sigma} \cdot \frac{1}{(N)_{\sigma+1}} .$$

The first, second and third term in the denominator on the right hand side represents the number of choices for X, Y and Z respectively, and the fourth term represents the number of choices for distinct permutation calls. We also get the ideal interpolation probability as

$$\Pr_{\mathcal{O}_0}[\tau] = \frac{1}{N^\sigma} \cdot \frac{1}{N^\sigma} \cdot \frac{1}{N^\sigma} \cdot \frac{1}{N^{\sigma+1}} .$$

The first, second and third term in the denominator on the right hand side represents the number of choices for X, Y and Z respectively, and the fourth term represents the number of choices for distinct permutation calls. Thus we finally we get

$$\frac{\Pr_{\mathcal{O}_1}[\tau]}{\Pr_{\mathcal{O}_0}[\tau]} \geq 1 .$$

Advantage of the Adversary. Applying H-Coefficient Technique, we get that the TSPRP advantage of the adversary is bounded above by

$$\epsilon_1 = \frac{2\sigma^2}{N} + \frac{2\sigma}{N} .$$

□