Quantum Attacks on Beyond-Birthday-Bound MACs

Hong-Wei Sun[1], Bin-Bin Cai[1], Su-Juan Qin[1], Qiao-Yan Wen[1], and Fei Gao[1⋆]

[1] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, 100876, China
gaof@bupt.edu.cn

Abstract. In this paper, we investigate the security of several recent MAC constructions with provable security beyond the birthday bound (called BBB MACs) in the quantum setting. On the one hand, we give periodic functions corresponding to targeted MACs (including PMACX, PMAC with parity, HPxHP, and HPxNP), and we can recover secret states using Simon algorithm, leading to forgery attacks with complexity $O(n)$. This implies our results realize an exponential speedup compared with the classical algorithm. Note that our attacks can even break some optimally secure MACs, such as mPMAC+-f, mPMAC+-p1, mPMAC+-p2, mLightMAC+-f, etc. On the other hand, we construct new hidden periodic functions based on SUM-ECBC-like MACs: SUM-ECBC, PolyMAC, GCM-SIV2, and 2K-ECBC_Plus, where periods reveal the information of the secret key. Then, by applying Grover-meets-Simon algorithm to specially constructed functions, we can recover full keys with $O(2^{n/2}n)$ or $O(2^{m/2}n)$ quantum queries, where n is the message block size and m is the length of the key. Considering the previous best quantum attack, our key-recovery attacks achieve a quadratic speedup.

Keywords: Beyond-Birthday-Bound · MAC · Quantum cryptanalysis · Quantum algorithm

1 Introduction

In recent years, a variety of fast quantum algorithms have been proposed for solving equations [1–3], dimensionality reduction [4–8], linear regression [9–13], anomaly detection [14, 15], classification [16–18], and so on [19–21]. The potential applications of quantum computation are expanding and deepening in various fields. Cryptography would undoubtedly be seriously impacted. For example, asymmetric primitives (e.g. RSA, ECC) would suffer from devastating attacks due to Shor's algorithm [22]. In symmetric cryptography, it has long been thought that the only threat was the quantum acceleration on exhaustive search [23], which leads to the fact that the best security a key of length n can offer is $2^{n/2}$ [24]. This was changed with the appearance of a Simon-based attack proposed by Kuwakado and Morii [25, 26], that is, they proved Even-Mansour

⋆ Corresponding author: Fei Gao.

and 3-round Feistel constructions would be broken in polynomial time. Several years later, more generic constructions were broken using different quantum algorithms, including the Simon-based attacks [27–31], the Grover-meets-Simon-based attacks [32–34], and the Bernstein-Vazirani (BV)-based attacks [35–38], etc.

Message Authentication Code (MAC) is a fundamental symmetric-key primitive to ensure the authenticity of data. Most popular MACs such as CBC-MAC [39], CMAC [40], OMAC [41], and GMAC [42], only achieve security up the birthday bound, i.e., the number of queries by the adversary is bounded by $2^{n/2}$, where n is the state size. However, the birthday-bound security might not be enough in practice, especially when a MAC is instantiated with a lightweight block cipher such as PRESENT [43], PRINCE [44], and GIFT [45] whose block size is small. In such a case, the birthday bound becomes 2^{32} as $n = 64$ and is vulnerable in certain practical applications. To go beyond the birthday bound, a series of block cipher-based MACs, which are secure for above $2^{n/2}$ queries (called BBB MACs), have been proposed, including SUM-ECBC [46], PolyMAC [47], GCM-SIV2 [48], 2K-ECBC_Plus [49], and some optimally secure MACs (such as mPMAC+-f, mLightMAC+-f, which are secure up to 2^n queries) [50], etc.

Previous attacks. At CRYPTO 2016, Kaplan et al. [29] showed that several widely used modes of operation for authentication and authenticated encryption, such as CBC-MAC [39], GMAC [42], PMAC [51], and some CAESAR candidates, could be broken by Simon algorithm. Recently, Bonnetain et al. [37] further introduced quantum forgery attacks on PolyMAC [47], GCM-SIV2 [48], LightMAC+ [52], PMAC-TBC3k [53], etc., with polynomial quantum queries by applying Simon algorithm. The crucial point is to construct a periodic function corresponding to the targeted block cipher, and then use Simon algorithm to recover the period. Recovering the period, which is a secret state, then allows to break the confidentiality or authenticity of these cryptographic primitives by recovering a key or distinguishing them from a random function. This kind of Simon-based attack provides an exponential speedup in the number of queries compared to classical attacks.

In addition to the Simon algorithm, the Grover-meets-Simon algorithm is also used to attack MACs. At ASIANCRYPT 2017, Leander and May used combinations of Simon algorithm and Grover algorithm to design the key recovery attack on FX-construction. The main idea is to construct a special period function with two inputs based on the targeted construction, say $f(u, x)$. That is, when the first input u equals a special value k, the function has a hidden period s such that $f(k, x) = f(k, x \oplus s)$ for all x. Here we call this kind of function a hidden periodic function. Leander and May proposed to use a Grover search for $u \in \{0, 1\}^m$. In order to test if a guess of u is the good one, they ran Simon algorithm with the function $f(u, x)$, which is periodic of period s if and only if $u = k$, and random otherwise. Thus, Grover acts as an outer loop with running time roughly $2^{m/2}$, and Simon acts as an inner loop with polynomial complexity. With this technique, Guo et al. [54] proposed for the first time quantum secret state recovery and key recovery attacks for a series of BBB MACs that were not vulnerable to the Simon algorithm, leading to forgery attacks. Unlike the exponential speedup of the Simon-based attack, these attacks only provide a polynomial speedup compared with classical attacks, i.e., the complexity reduces from $O(2^{3n/4})$ to $O(2^{n/2})$.

Our contributions. Till now, for most BBB MACs, there are no successful Simon attacks (which generally achieve exponential speedup). Only Grover-meets-Simon attacks (which achieve polynomial speedup) were given for some BBB MACs. In this work, we further study the BBB MACs' security in quantum circumstances and answer the following two questions. Table 1 summarizes our main results and comparison with previous works.

1. Can we give Simon attacks for BBB MACs?
   For some MACs such as PMACX, PMAC with parity, HPxHP, HPxNP, etc., we give periodic functions corresponding to targeted MACs and then utilize Simon algorithm to recover the secret state, which leads to a successful forgery attack. The proposed attacks need only $O(n)$ quantum queries and realize an exponential speedup compared with their classical versions. Besides, our attacks are more efficient than some related results, i.e., it exponentially improves previous quantum attacks (with complexity $O(2^{n/2})$) [54] on mPMAC+-f, mPMAC+-p1, and mPMAC+-p2 from the viewpoint of quantum query complexity.

2. Are there any better Grover-meets-Simon attacks for those BBB MACs which were attacked in Ref. [54]?
   For SUM-ECBC-like MACs such as SUM-ECBC, PolyMAC, GCM-SIV2, and 2K-ECBC-Plus, we construct new condition-period functions based on targeted MACs, where periods reveal the information of the secret key. Therefore, we can apply Grover-meets-Simon algorithm to recover the secret key with a complexity of $O(2^{n/2}n)$ or $O(2^{m/2}n)$, where n is the message block size and m is the length of the key of the underlying block cipher. For the case of usual block ciphers (i.e., $m = O(n)$), our result achieves a quadratic speedup compared with the previous quantum attacks.

Organization. The paper is organized as follows. In Sect. 2, we introduce some basic notations, the quantum algorithms (Grover, Simon, and Grover-meets-Simon algorithms) used in this paper, and the quantum security of MACs. In Sect. 3, we propose secret state recovery attacks on several BBB MACs by applying Simon algorithm. In Sect. 4, we give some new quantum key-recovery attacks by applying the Grover-meets-Simon algorithm. Finally, we conclude in Sect. 5.

Table 1: Summary of the main results, where n is the block size, and m is the length of the key of the underlying block cipher.

Goal	Construction	# Keys[3]	Provable classical security query bound	[54]			ours		
				queries	qubits	algorithm	queries	qubits	algorithm
SR[1]	mPMAC+-f [50]	5	$\Omega(2^n)$ [50]	$O(2^{n/2}n)$	$O(n^2)$	Grover-meets-Simon	$O(n)$	$O(n)$	Simon
	mPMAC+-p1 [50]	5	$\Omega(2^n)$ [50]	$O(2^{n/2}n)$	$O(n^2)$	Grover-meets-Simon	$O(n)$	$O(n)$	Simon
	mPMAC+-p2 [50]	5	$\Omega(2^n)$ [50]	$O(2^{n/2}n)$	$O(n^2)$	Grover-meets-Simon	$O(n)$	$O(n)$	Simon
	mLightMAC+-f [50]	5	$\Omega(2^n)$ [50]	-	-	-	$O(n)$	$O(n)$	Simon
	mLightMAC+-p1 [50]	5	$\Omega(2^n)$ [50]	-	-	-	$O(n)$	$O(n)$	Simon
	mLightMAC+-p2 [50]	5	$\Omega(2^n)$ [50]	-	-	-	$O(n)$	$O(n)$	Simon
	PMACX [55]	2	$\Omega(2^{n/2})$ [55]	-	-	-	$O(n)$	$O(n)$	Simon
	HPxHP [56]	2	$\Omega(2^{2n/3})$ [56]	-	-	-	$O(n)$	$O(n)$	Simon
	HPxNP [56]	2	$\Omega(2^{2n/3})$ [56]	-	-	-	$O(n)$	$O(n)$	Simon
	PMAC with parity [57]	4	$\Omega(2^{n/2})$ [57]	-	-	-	$O(n)$	$O(n)$	Simon
KR[2]	SUM-ECBC [46]	4	$\Omega(2^{3n/4})$[47]	$O(2^m n)$	$O(m+n^2)$	Grover-meets-Simon	$O(2^{m/2}n)$	$O(m+n^2)$	Grover-meets-Simon
	PolyMAC [47]	4	$\Omega(2^{3n/4})$[47]	$O(2^{(m+n)/2}n)$	$O(m+n^2)$	Grover-meets-Simon	$O(2^{n/2}n)$	$O(n^2)$	Grover-meets-Simon
	GCM-SIV2 [48]	6	$\Omega(2^{2n/3})$[48]	$O(2^{(m+n)/2}n)$	$O(m+n^2)$	Grover-meets-Simon	$O(2^{n/2}n)$	$O(n^2)$	Grover-meets-Simon
	2K-ECBC_Plus [49]	3	$\Omega(2^{2n/3})$ [49]	$O(2^m n)$	$O(m+n^2)$	Grover-meets-Simon	$O(2^{m/2}n)$	$O(m+n^2)$	Grover-meets-Simon

[1] secret state recovery
[2] key recovery
[3] the number of block cipher keys used in the construction

2 Preliminaries

Let F_2 denote the prime field with two elements 0 and 1. And the n-dimensional vector space of F_2 is denoted by F_2^n. We let " $\oplus$ " denote the XOR (addition in F_2^n), " $\odot$ " denote multiplication in F_2^n, and " $\cdot$ " denote the scalar product of bit-strings seen as n-bit vectors. Let $|X|$ be the number of the elements in set X.

2.1 Quantum algorithm

In the following, we review Grover, Simon, and Grover-meets-Simon algorithms used in this paper. We refer to [34, 58] for a broader presentation.

1) Grover algorithm. Grover algorithm [23] is a well-known quantum algorithm that achieves quadratic speedups on database searching tasks compared with classical algorithms. Precisely, it solves the following problem.

Grover problem. Let $f : X \to \{0,1\}$ be a test function. Given oracle to f, find $x \in X$ such that $f(x) = 1$.

Classically, one preimage is expected to be found in time (and oracle access to f) $O(\frac{|X|}{e})$ if there are e preimages of 1 ($|\{x : f(x) = 1\}| = e$). Quantumly, Grover algorithm finds one preimage in time (and oracle access to O_f) $O(\sqrt{\frac{|X|}{e}})$. Grover algorithm works first by producing a uniform superposition $|\psi\rangle = \frac{1}{\sqrt{|X|}} \Sigma_{x \in X} |x\rangle$. Next, it repeatedly applies the unitary operator $(2|\psi\rangle\langle\psi| - I)O_f$ on the state $|\psi\rangle$. The process increases the amplitude of success roughly by a constant on each iteration. Then a final measurement will produce a good state with an overwhelming probability. Generally, the checking procedure can be done only with some errors. The test function always returns 1 for elements in the target set, but for elements not in the target set, it also returns 1 with a negligible probability. The following theorem tackles this case.

Theorem 1 [59, 54]. Let $X \in \{0,1\}^m$, $p_0 := \frac{e}{2^m}$ and $f : \{0,1\}^m \to \{0,1\}$ be a test function such that

$$\begin{cases} Pr[f(x) = 1] = 1 & \text{if } x \in X, \\ Pr[f(x) = 1] \leq p_1 & \text{if } x \notin X. \end{cases} \tag{1}$$

Assume the quantum implementation of $f(x)$ costs $O(n)$ qubits. Then Grover algorithm with $t = \lceil \frac{\pi}{4arcsin\sqrt{p_0}} \rceil$ quantum queries to $f(x)$ and $O(m+n)$ qubits will output an $x \in X$ with probability at least $\frac{p_0}{p_0+p_1}[1 - (\frac{p_1}{p_0} + \sqrt{p_0+p_1} + 2\sqrt[3]{1+\frac{p_1}{p_0}p_0})^2]$.

In particular, if $e \leq 2$ and $p_1 \leq \frac{1}{2^{2m}}$, the error decreases exponentially with m.

2) Simon algorithm. Simon algorithm [60] gives the first example of an exponential quantum time speedup relative to an oracle. That is, it can find the period of a periodic function in polynomial time.

Simon problem. Given a function $f : \{0,1\}^n \to \{0,1\}^d$ and promise that there exists $s \in \{0,1\}^n$ such that for any $(x,y) \in \{0,1\}^n$, $[f(x) = f(y)] \Leftrightarrow [x \oplus y \in \{0,s\}]$, the goal is to find s.

This problem can be solved classically by searching collisions with $O(2^{n/2})$ queries. As the quantum superposition of queries of form $\Sigma_{x,y}\lambda_{x,y}|x\rangle|y\rangle \mapsto \Sigma_{x,y}\lambda_{x,y}|x\rangle|f(x) \oplus y\rangle$ is introduced into Simon algorithm, its query complexity is only $O(n)$. After repeating the following subroutine (Algorithm 1) cn times, we can obtain s by solving a system of linear equations. The algorithm can be applied to the problem of which condition "$f(x) = f(y)$ if and only if $x \oplus y \in \{0,s\}$" is replaced with the weaker condition "$f(x \oplus s) = f(x)$ for any x", under the assumption that f satisfies some good properties. Concretely, Kaplan et al. [29] have proved the following theorem.

Theorem 2 [29]. Let $\varepsilon(f,s) := \max\limits_{t \in \{0,1\}^n \setminus \{0,s\}} Pr_x[f(x) = f(x \oplus t)]$. If $\varepsilon(f,s) \leq p_0 < 1$, then Simon algorithm returns s with cn queries and $O(n+d)$ qubits, with probability at least $1 - (2(\frac{1+p_0}{2})^c)^n$.

3) Grover-meets-Simon algorithm. In Ref. [34], Leander and May proposed to combine Simon algorithm with Grover algorithm (i.e., Grover-meets-Simon

[1] When there is no ambiguity, we write 0 for the vector $(0,0,\cdots,0)$ of appropriate length.

Algorithm 1 Quantum subroutine of Simon algorithm.

Input: n, $O_f : |x\rangle|0\rangle \mapsto |x\rangle|f(x)\rangle$
Output: y orthogonal to s

1: Applying a Hadamard transform $H^{\otimes n}$ to the initial state $|\psi_0\rangle = |0\rangle|0\rangle$ [1] (a $(n+d)$-qubit state) to obtain the quantum superposition

$$|\psi_1\rangle = \frac{1}{\sqrt{2^n}}\Sigma_{x\in F_2^n}|x\rangle|0\rangle.$$

2: A quantum query to the function f maps to the state

$$|\psi_2\rangle = \frac{1}{\sqrt{2^n}}\Sigma_{x\in F_2^n}|x\rangle|f(x)\rangle.$$

3: Measuring the second register gives a value $f(z)$ and the first register is collapsed to

$$|\psi_3\rangle = \frac{1}{\sqrt{2}}(|z\rangle + |z\oplus s\rangle).$$

4: Applying again the Hadamard transform $H^{\otimes n}$ to the first register yields

$$\frac{1}{\sqrt{2}}\frac{1}{\sqrt{2^n}}\Sigma_{y\in F_2^n}(-1)^{y\cdot z}(1+(-1)^{y\cdot s})|y\rangle.$$

5: Measuring the state yields a value of y, which meets that $y \cdot s = 0$.

algorithm) to attack the construction with whitening keys. This algorithm solves the following problem.

Grover-meets-Simon problem. Let $f : \{0,1\}^m \times \{0,1\}^n \rightarrow \{0,1\}^d$ be a function such that there exist some $u \in \{0,1\}^m$ such that $f(u,\cdot)$ hide a non-trivial period s_u. Find any tuple $(u, s_u) \in U_s$, where $U_s := \{(u, s_u) : u \in \{0,1\}^m, s_u$ is the period of $f(u,\cdot)\}$.

Leander and May define a Grover search over $u \in \{0,1\}^m$, where they test for the periodicity of every $f(u,\cdot)$ via Simon algorithm. Thus, they have Grover algorithm as an outer loop with a running time of roughly $2^{m/2}$, and Simon algorithm as an inner loop with polynomial complexity. The following theorem shows the effect of the parameter

$$\varepsilon(f) := \max_{(u,t)\in\{0,1\}^m\times\{0,1\}^n\setminus\{0,U_s\}} Pr_x[f(u,x) = f(u,x\oplus t)] \qquad (2)$$

on the success probability of the Grover-meets-Simon algorithm.

Theorem 3 [59, 54]. Let c be a positive integer, $p_0 := \frac{e}{2^m}$ and $p_1 := [2 \cdot (\frac{1+\varepsilon(f)}{2})^c]^n$. Then Grover-meets-Simon algorithm with $\lceil\frac{\pi}{4arcsin\sqrt{p_0}}\rceil \cdot cn$ quantum queries to f and $O(m + cn^2 + cdn)$ qubits outputs a tuple $(u, s_u) \in U_s$ with probability at least $\frac{(1-p_1)p_0}{p_0+p_1}[1 - (\frac{p_1}{p_0} + \sqrt{p_0 + p_1} + 2\sqrt[3]{1 + \frac{p_1}{p_0}}\, p_0)^2]$.

In particular, if $\varepsilon(f) \leq 1/2$ and $e \leq 2$, the error decreases exponentially with n. In the case $d = m = n$, the Grover-meets-Simon algorithm solves this problem with $O(2^{n/2}n)$ quantum queries and $O(n^2)$ qubits.

2.2 Quantum security of MACs

Message Authentication Code (MAC) is a fundamental symmetric-key primitive to ensure the authenticity of data. A MAC system contains two algorithms: a MAC signing algorithm $S(k, m)$ and a MAC verification algorithm $V(k, m, T)$. Here k denotes the secret key, m denotes a message and T denotes the MAC tag. Classically, a MAC system is considered to be secure if an efficient attacker capable of mounting a chosen message attack cannot produce an existential MAC forgery. To translate this security notion to the quantum setting, Boneh and Zhandry [61] assumed that the adversary can make quantum queries to the signing oracle, and defined the existential unforgeability against quantum chosen message attack (EUF-qCMA). That is, a MAC is EUF-qCMA security if the adversary cannot generate $q + 1$ valid classical message-tag pairs after making q quantum chosen message queries.

3 Quantum secret state recovery attack for BBB MACs

In this section, we focus on quantum secret state recovery attacks against BBB MACs. In particular, we give polynomial-time attacks on PMACX, PMAC with parity, HPxHP, and HPxNP, and show that they can also be extended to some optimally secure MACs. Recovering the secret state leads to a forgery attack. We improve some previous superposition attacks by reducing the query complexity from exponential [54] to polynomial. See Table 1 for a comparison of attack complexity.

3.1 Attack strategy

In the following, we give a strategy for attacking BBB MACs by using Simon algorithm. Our attack is described as the following procedure:

1. Construct a periodic function f corresponding to the targeted MAC, where the period s satisfies $f(x) = f(x \oplus s)$ for all x;

2. Run Simon algorithm for the above f to find s.

Recovering the period s allows one to recover a key, distinguish, carry out forgery attacks, etc. Note that our strategy requires that the attacker has quantum oracle access to f.

Quantum linearization attacks. In fact, the core step of this attack strategy is to construct a periodic function. The common method (like those of [29, 30]) is invalid for BBB MACs. Here, we introduce a new technique. At Asiacrypt 2021 [37], Bonnetain et al. showed a quantum linearization attack against the EUF-qCMA security of MACs. Specifically, they use inputs of multiple blocks as an interface of a function to hide a linear structure. The main idea is to linearize the function by limiting the block inputs to obtain an affine function. Consider a function of l blocks $x_1, x_2, \cdots, x_l$ with the form of $G(x_1, x_2, \cdots, x_l) = g_1(x_1) \oplus g_2(x_2) \oplus \cdots \oplus g_l(x_l) \oplus C$, where C is an independent constant, and the attacker has no access to the g_i $(1 \leq i \leq l)$, which are independent random functions. Then they make each block x_i $(1 \leq i \leq l)$ takes only a one-bit value, and define the following function

$$F(x) = F(b_1 \| \cdots \| b_l) = G(0^{n-1} \| b_1, \cdots, 0^{n-1} \| b_l). \tag{3}$$

Now, F is an affine function of $b_1, \cdots, b_l$, and BV algorithm[1] can distinguish it from a random function in polynomial time. In particular, the linearization attack can serve as an efficient method to construct a periodic function. For example, by linearizing the function $G'(x_1, x_2, \cdots, x_l) = g(G(x)) = g(g_1(x_1) \oplus g_2(x_2) \oplus \cdots \oplus g_l(x_l) \oplus C)$, we can obtain a periodic function $G'(x) = g(F(x))$, where $F(x)$ is an affine function, g is a random function and is unknown to the attacker.

3.2 Quantum secret state recovery attack for BBB MACs

From the above claim, we need to construct a periodic function based on the targeted MAC, and then the Simon algorithm to recover the period is used. In what follows, taking PMAC with parity as an example, we give the detailed attack process and the complexity analysis for quantum adversaries.

1) Secret state recovery attack for PMAC with parity. PMAC with parity [57] is a variant of PMAC (Parallelizable Message Authentication Code). It uses four permutations P_1, P_2, P_3 and P_4, which are in practice realized via a block cipher using four keys. PMAC with parity with a $2i$-block message is shown in Fig. 1, which can be written as

[1] The BV algorithm [38] offers a polynomial speedup for finding the slope of an affine function over F_2^n.

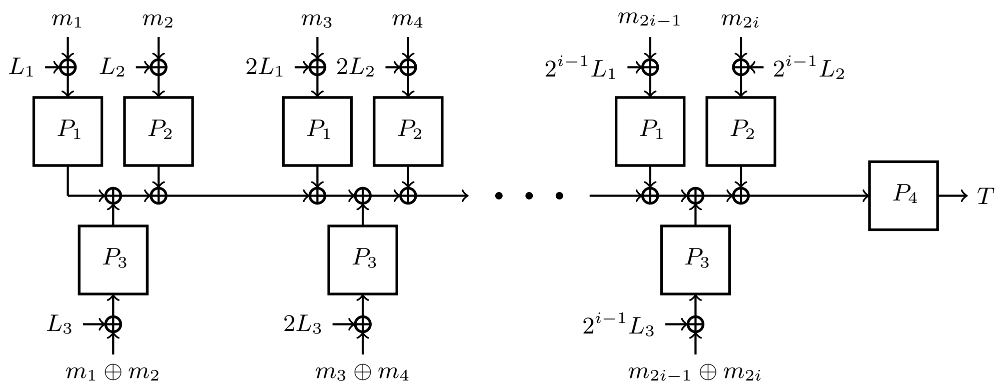

Fig. 1: PMAC with parity [57].

$$\begin{aligned}\mathrm{MAC}(m_1, m_2, \cdots, m_{2i}) = P_4(&\bigoplus_{j=1}^{i} P_1(m_{2j-1} \oplus 2^{j-1}L_1) \oplus \bigoplus_{j=1}^{i} P_2(m_{2j} \oplus 2^{j-1}L_2) \\ &\oplus \bigoplus_{j=1}^{i} P_3(m_{2j-1} \oplus m_{2j} \oplus 2^{j-1}L_3)),\end{aligned} \tag{4}$$

where $L_1 = P_1(0)$, $L_2 = P_2(0)$, $L_3 = P_3(0)$, and the size of the block is n.

We now show that the PMAC with parity is not secure in a quantum setting. We consider the case that each even-block message is an arbitrary constant, and define the following function for the odd-block messages, with some arbitrary constants m^0_{2j-1} and m^1_{2j-1} such that $m^0_{2j-1} \neq m^1_{2j-1}$ $(1 \leq j \leq i)$:

$$\begin{aligned}F(b) &\equiv \mathrm{MAC}(m_1^{b_1}, m_2, m_3^{b_2}, m_4, \cdots, m_{2i-1}^{b_i}, m_{2i}) \\ &= P_4(\bigoplus_{j=1}^{i} P_1(m^{b_j}_{2j-1} \oplus 2^{j-1}L_1) \oplus \bigoplus_{j=1}^{i} P_2(m_{2j} \oplus 2^{j-1}L_2) \oplus \bigoplus_{j=1}^{i} P_3(m^{b_j}_{2j-1} \oplus m_{2j} \oplus 2^{j-1}L_3)) \\ &= P_4(f(b)),\end{aligned} \tag{5}$$

where $b = b_1 b_2 \cdots b_i$ forms an i-bit input. It is easy to see that f is an affine function of b:

$$\begin{aligned}f(b) &= \bigoplus_{j=1}^{i} P_1(m^{b_j}_{2j-1} \oplus 2^{j-1}L_1) \oplus \bigoplus_{j=1}^{i} P_2(m_{2j} \oplus 2^{j-1}L_2) \oplus \bigoplus_{j=1}^{i} P_3(m^{b_j}_{2j-1} \oplus m_{2j} \oplus 2^{j-1}L_3) \\ &= \bigoplus_{j=1}^{i}((P_1(m^0_{2j-1} \oplus 2^{j-1}L_1) \oplus P_1(m^1_{2j-1} \oplus 2^{j-1}L_1)) \odot b_j \oplus P_1(m^0_{2j-1} \oplus 2^{j-1}L_1)) \\ &\quad \oplus \bigoplus_{j=1}^{i}((P_3(m^0_{2j-1} \oplus m_{2j} \oplus 2^{j-1}L_3) \oplus P_3(m^1_{2j-1} \oplus m_{2j} \oplus 2^{j-1}L_3)) \odot b_j \oplus P_3(m^0_{2j-1} \oplus m_{2j} \oplus 2^{j-1}L_3)) \\ &\quad \oplus \bigoplus_{j=1}^{i} P_2(m_{2j} \oplus 2^{j-1}L_2)\end{aligned}$$

$$
\begin{aligned}
&= (P_1(m_1^0 \oplus L_1) \oplus P_1(m_1^1 \oplus L_1), \cdots, P_1(m_{2i-1}^0 \oplus 2^{i-1}L_1) \oplus P_1(m_{2i-1}^1 \oplus 2^{i-1}L_1)) \times \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_i \end{pmatrix} \\
&\quad \oplus (P_3(m_1^0 \oplus m_2 \oplus L_1) \oplus P_3(m_1^1 \oplus m_{2i} \oplus L_1), \cdots, P_3(m_{2i-1}^0 \oplus m_{2i} \oplus 2^{i-1}L_1) \oplus P_3(m_{2i-1}^1 \oplus m_{2i} \oplus 2^{i-1}L_1)) \times \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_i \end{pmatrix} \\
&\quad \oplus \bigoplus_{j=1}^{i} P_1(m_{2j-1}^0 \oplus 2^{j-1}L_1) \oplus \bigoplus_{j=1}^{i} P_3(m_{2j-1}^0 \oplus m_{2j} \oplus 2^{j-1}L_3) \oplus \bigoplus_{j=1}^{i} P_2(m_{2j} \oplus 2^{j-1}L_2) \\
&= (A_m \oplus A'_m) \times \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_i \end{pmatrix} \oplus C,
\end{aligned} \tag{6}
$$

where the columns of A_m correspond to $P_1(m_{2j-1}^0 \oplus 2^{j-1}L_1) \oplus P_1(m_{2j-1}^1 \oplus 2^{j-1}L_1)$, the columns of A'_m correspond to $P_3(m_{2j-1}^0 \oplus m_{2j} \oplus 2^{j-1}L_3) \oplus P_3(m_{2j-1}^1 \oplus m_{2j} \oplus 2^{j-1}L_3)$ and $C = \bigoplus_{j=1}^{i} P_1(m_{2j-1}^0 \oplus 2^{j-1}L_1) \oplus \bigoplus_{j=1}^{i} P_3(m_{2j-1}^0 \oplus m_{2j} \oplus 2^{j-1}L_3) \oplus \bigoplus_{j=1}^{i} P_2(m_{2j} \oplus 2^{j-1}L_2)$. Then,

$$
F(b) = P_4\left((A_m \oplus A'_m) \times \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_i \end{pmatrix} \oplus C\right), \tag{7}
$$

where the matrix $A_m \oplus A'_m$ has n rows and i columns, and its kernel is nontrivial if and only if $i \geq n+1$. That is, there exists a non-zero vector s such that

$$
(A_m \oplus A'_m) \times s = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}. \tag{8}
$$

Obviously, F satisfies $F(b \oplus s) = F(b)$:

$$
F(b \oplus s) = P_4((A_m \oplus A'_m) \times (b \oplus s) \oplus C) = F(b). \tag{9}
$$

That is the function F is a periodic function of b. Since the “inner" function f is an affine function and P_4 is a permutation function, they do not contain any unwanted collisions. Therefore, according to Theorem 2, we can apply Simon algorithm to recover the secret state with a high probability.

Note that the subspace of periods will become larger as the i increases. In particular, if $i = n$, there will be a non-trivial period with a probability around $1 - 1/e$. Recovering the secret state, i.e. the period s, allows to forge messages easily:

1. Query the tag of $(m_1^{b_1}, m_2, m_3^{b_2}, m_4, \cdots, m_{2i-1}^{b_i}, m_{2i})$ for an arbitrary b;
2. The same tag is valid for $(m_1^{b_1 \oplus s_1}, m_2, m_3^{b_2 \oplus s_2}, m_4, \cdots, m_{2i-1}^{b_i \oplus s_i}, m_{2i})$.

As for SUM-ECBC, these two steps can be repeated $q'+1$ times, where q' is the number of quantum queries issued. The adversary then produces $2(q'+1)$ messages after only $2q'+1$ queries to the cryptographic oracle.

2) Secret state recovery attack for PMACX. In 2015, Zhang [55] combined the construction of PMAC with parity and MDS-coding to design PMACX. It can be viewed as a generalization of PMAC with parity, whose "parity processing" part is replaced with a general MDS generator matrix multiplication. The message blocks $M_1, M_2, \cdots, M_s$ are processed as follows: $X_i = X_i[1] \parallel X_i[2] \parallel \cdots \parallel X_i[m] = G \cdot M_i$ and then

$$\mathrm{PMACX}(M_1, M_2, \cdots, M_s) = P_2(\bigoplus_{i=1}^{s} \bigoplus_{j=1}^{m} P_1(X_i[j] \oplus 2^{i-1} L_j)), \tag{10}$$

where the message blocks M_i are of size ln, G is an $m \times l$ matrix over $GF(2^n)$, and $L_j = P_1(j-1)$.

In a similar way, we define the following function using the given random constants m_i^0 and m_i^1:

$$\mathrm{PMACX}(M_1^{b_1}, M_2^{b_2}, \cdots, M_s^{b_s}) = P_2 \left(A_m \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_s \end{pmatrix} \oplus C \right) \equiv f(b), \tag{11}$$

where the columns of A_m correspond to $\bigoplus_{j=1}^{m} \left(P_1(X_i^0 \oplus 2^{i-1} L_j) \oplus P_1(X_i^1 \oplus 2^{i-1} L_j)\right)$, $C = \bigoplus_{i=1}^{s} \bigoplus_{j=1}^{m} P_1(X_i^0[j] \oplus 2^{i-1} L_j)$. When $s \geq n+1$, we can obtain a periodic function, and break PMACX.

3) Secret state recovery attack for HPxHP and HPxNP. In 2019, Alexander and Eik proposed [56] two constructions based on permutations and universal hashing, providing a security proof up to $2^{2n/3}$ queries. The first structure (HPxHP) is a stateless deterministic scheme that uses two hash functions, whereas the second structure (HPxNP) is a nonce-based scheme with one hash-function call and a nonce. As shown in Fig. 2 and Fig. 3 respectively, these two MACs can be written as

$$\begin{aligned} \mathrm{HPxHP}(m_1, m_2, \cdots, m_l) &= P_1(k_1^l m_1 \oplus k_1^{l-1} m_2 \oplus \cdots \oplus k_1^1 m_l) \oplus P_2(k_2^l m_1 \oplus k_2^{l-1} m_2 \oplus \cdots \oplus k_1^2 m_l) \\ \mathrm{HPxNP}(m_1, m_2, \cdots, m_l) &= P_1(k_1^l m_1 \oplus k_1^{l-1} m_2 \oplus \cdots \oplus k_1^1 m_l \oplus N) \oplus P_2(N), \end{aligned} \tag{12}$$

where P_1 and P_2 represent two permutations over $\{0,1\}^n$, h_1 and h_2 are two universal hash functions and N is a nonce. By fixing each block message as m_i^0

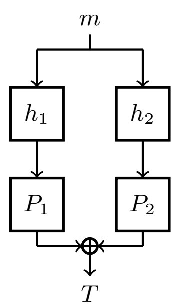

Fig. 2: HPxHP.

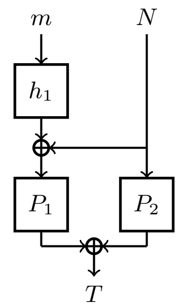

Fig. 3: HPxNP.

and m_i^1, we can recover a period with Simon algorithm such that

$$\bigoplus_i s_i k_1^{l+1-i} = 0 \ \text{ and } \ \bigoplus_i s_i k_2^{l+1-i} = 0. \tag{13}$$

This provides a forgery attack, then we can recover multiple such periods and solve the corresponding equations to obtain K_1 and K_2.

Role of the nonce. In this paper, we focus on two different kinds of quantum access constructions: those that use a nonce (e.g. HPxNP) and those that do not (e.g. HPxHP). In the nonce case, we use a weaker security notion (following the IND-qCPA definition of [62, 29]) where the nonce is chosen randomly by the oracle, and not repeated. The oracle O_{f_N} is then $M \mapsto (N, MAC(N, M))$. If we can break the MAC construction in this model, the attack will also be valid with any reasonable CPA security definition. In this setting, applying the subroutine of Simon algorithm to the function f_N always gives a vector orthogonal to s, for any random choice of N. Therefore, we can still recover s after $O(n)$ steps, even if each step uses a different value of N.

4) Secret state recovery attack for some optimally secure MACs. In Ref. [50], Cogliati et al. introduced several constructions to build optimally secure variable-input-length (VIL) PRFs from secret random permutations, such as mPMAC+-f, mPMAC+-p1, mPMAC+-p2, mLightMAC+-f, mLightMAC+-p1 and mLightMAC+-p2. They are secure up to 2^n queries, where n denotes the block size. Here only take mPMAC+-p2 as an example:

$$\begin{aligned}\text{mPMAC+-p2}(m_1, m_2, \cdots, m_l) = {} & P_3(P_1(\bigoplus_i^{l-1} P_0(m_i \oplus \triangle_i) \oplus m_l) \oplus \bigoplus_{i=1}^{l-1} 2^{l-i} P_0(m_i \oplus \triangle_i) \oplus m_l) \\ & \oplus P_4(P_2(\bigoplus_i^{l-1} 2^{l-i} P_0(m_i \oplus \triangle_i) \oplus m_l) \oplus \bigoplus_{i=1}^{l-1} P_0(m_i \oplus \triangle_i) \oplus m_l),\end{aligned} \tag{14}$$

where $\triangle_i = 2^i P_0(0^n) \oplus 2^{2i} P_0(10^{n-1})$ and P is a random permutation. Let m_l be a fixed value, and define the following function for the $l-1$ block messages with some arbitrary constants m_i^0 and m_i^1 such that $m_i^0 \neq m_i^1$ $(1 \leq i \leq l-1)$:

$$F(b) \equiv \text{mPMAC+-p2}(m_1^{b_1}, m_2^{b_2}, \cdots, m_{l-1}^{b_{l-1}}, m_l). \tag{15}$$

In that case, we can remark that there exists a period s such that $A_m s = A'_m s = 0$, where matrices A_m and A'_m have n rows and $l-1$ columns. The columns of A_m correspond to $P_0(m_i^0 \oplus \triangle_i) \oplus P_0(m_i^1 \oplus \triangle_i)$ and the columns of A'_m correspond to $2^{l-i}(P_0(m_i^0 \oplus \triangle_i) \oplus P_0(m_i^1 \oplus \triangle_i))$. Therefore, the period s can be recovered with $O(n)$ quantum queries and $O(n)$ qubits by Theorem 2.

4 Quantum key-recovery attack for BBB MACs

This section gives new quantum key-recovery attacks on SUM-ECBC-like MACs, such as SUM-ECBC, PolyMAC, GCM-SIV2, and 2K-ECBC_Plus, with $O(2^{m/2}n)$ or $O(2^{n/2}n)$ superposition queries. They achieve a quadratic acceleration of the query complexity of some previous attacks [54]. See Table 2 for a comparison of attack complexity.

Table 2: Summary of previous and new quantum key-recovery attacks, where n is the block size, and m is the length of the key of the underlying block cipher.

Construction	# Keys	Provable classical security query bound	Query complexity of classical attack	[54]		ours	
				queries	qubits	queries	qubits
SUM-ECBC [46]	4	$\Omega(2^{3n/4})$ [47]	$O(2^{3n/4})$ [63]	$O(2^m n)$	$O(m+n^2)$	$O(2^{m/2}n)$	$O(m+n^2)$
PolyMAC [47]	4	$\Omega(2^{3n/4})$ [47]	$O(2^{3n/4})$	$O(2^{(m+n)/2}n)$	$O(m+n^2)$	$O(2^{n/2}n)$	$O(n^2)$
GCM-SIV2 [48]	6	$\Omega(2^{2n/3})$ [48]	$O(2^{3n/4})$ [63]	$O(2^{(m+n)/2}n)$	$O(m+n^2)$	$O(2^{n/2}n)$	$O(n^2)$
2K-ECBC_Plus [49]	3	$\Omega(2^{2n/3})$ [49]	$O(2^{3n/4})$	$O(2^m n)$	$O(m+n^2)$	$O(2^{m/2}n)$	$O(m+n^2)$

4.1 Attack strategy

The SUM-ECBC-like MACs follow a generic design paradigm called Double-block Hash-then-Sum (in short DbHtS) [49]. In this paradigm, it computes a double block hash on the message and then sums the encrypted output of these two hash blocks:

$$DbHtS(M) = G(M) \oplus H(M). \tag{16}$$

Note that MACs of single-chain, such as ECBC-MAC, can be broken by using the quantum period finding algorithm [54, 29]. Considering the single-chain G (resp. H), we can use the same method C [29] to construct a period function $g(b,x) = C^G(b,x)$ (resp. $h(b,x) = C^H(b,x)$). The period of g (resp. h) is denoted as $1\|s_1$ (resp. $1\|s_2$). In particular, the period value s can be determined by the underlying block cipher with key k and a fixed pair of messages (α_0, α_1), i.e., $s_1 = E_{k_1}(\alpha_0) \oplus E_{k_1}(\alpha_1)$ and $s_2 = E_{k_3}(\alpha_0) \oplus E_{k_3}(\alpha_1)$. Then, applying the method C to $DbHtS = G \oplus H$ will give

$$C^{DbHtS}(b,x) = C^G(b,x) \oplus C^H(b,x) = g(b,x) \oplus h(b,x). \tag{17}$$

More precisely, we define the following function, with two arbitrary constants α_0 and α_1 such that $\alpha_0 \neq \alpha_1$:

$$\begin{aligned} f(u,x) &= C^{DbHtS}(0,x) \oplus C^{DbHtS}(1, x \oplus E_u(\alpha_0) \oplus E_u(\alpha_1)) \\ &= g(0,x) \oplus h(0,x) \oplus g(1, x \oplus E_u(\alpha_0) \oplus E_u(\alpha_1)) \\ &\quad \oplus h(1, x \oplus E_u(\alpha_0) \oplus E_u(\alpha_1)). \end{aligned} \tag{18}$$

In particular, this function is periodic if and only if $u = k_1/k_3$. Then, we can apply the Grover-meets-Simon algorithm to recover k_1/k_3.

4.2 Key recovery attack for SUM-ECBC-like MACs

1) Key recovery attack for SUM-ECBC. SUM-ECBC [46] was presented by Yasuda in 2010, inspired by MAC constructions summing two encrypted CBC-MACs. It uses a block cipher keyed with four independent keys in $\{0,1\}^m$, denoted as E_1, E_2, E_3, and E_4. For a message $M = m_1 \| m_2$, SUM-ECBC is defined as (see Fig. 4):

$$\text{SUM-ECBC}(m_1, m_2) = E_2(E_1(E_1(m_1) \oplus m_2)) \oplus E_4(E_3(E_3(m_1) \oplus m_2)). \tag{19}$$

Here, we only describe the modes with full-block messages for simplicity, the

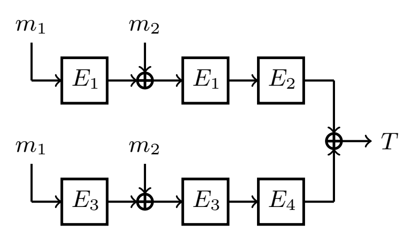

Fig. 4: SUM-ECBC with a two-block message.

attacks can trivially be extended to the more general modes with arbitrary inputs.

In what follows, based on the Grover-meets-Simon algorithm, we present a new quantum key recovery attack on SUM-ECBC and the complexity analysis for quantum adversaries. We first focus on the partial key recovery.

Partial key recovery. We fix two arbitrary message blocks α_0, α_1 with $\alpha_0 \neq \alpha_1$, and define the following function

$$\begin{aligned} \phi : \{0,1\} \times \{0,1\}^n &\to \{0,1\}^n \\ b, x &\mapsto \text{SUM-ECBC}(\alpha_b \| x) = g(b,x) \oplus h(b,x), \end{aligned} \tag{20}$$

where $g(b,x) = E_2(E_1(E_1(\alpha_b) \oplus x))$, $h(b,x) = E_4(E_3(E_3(\alpha_b) \oplus x))$. It is easy to see that the function g (resp. h) satisfies $g(0,x) = g(1, x \oplus s_1)$ (resp. $h(0,x) = h(1, x \oplus s_2)$), where $s_1 = E_1(\alpha_0) \oplus E_1(\alpha_1)$, $s_2 = E_3(\alpha_0) \oplus E_3(\alpha_1)$. By the randomness of k_1 and k_3, the probability of $s_1 = s_2$ is negligible. To realize partial key recovery with the Grover-meets-Simon algorithm, we define the following function (see Fig. 5)

$$\begin{aligned} f : \{0,1\}^m \times \{0,1\}^n &\to \{0,1\}^n \\ u, x &\mapsto \text{SUM-ECBC}(\alpha_0, x) \\ &\quad \oplus \text{SUM-ECBC}(\alpha_1, x \oplus E_u(\alpha_0) \oplus E_u(\alpha_1)). \end{aligned} \tag{21}$$

In particular, this function is periodic if and only if $u = k_1/k_3$, and we take $u = k_1$ as an example:

$$\begin{aligned} f(k_1, x) &= \text{SUM-ECBC}(\alpha_0, x) \oplus \text{SUM-ECBC}(\alpha_1, x \oplus E_1(\alpha_0) \oplus E_1(\alpha_1)) \\ &= g(0,x) \oplus h(0,x) \oplus g(1, x \oplus E_1(\alpha_0) \oplus E_1(\alpha_1)) \oplus h(1, x \oplus E_1(\alpha_0) \oplus E_1(\alpha_1)) \\ &= h(0,x) \oplus h(1, x \oplus E_1(\alpha_0) \oplus E_1(\alpha_1)). \end{aligned} \tag{22}$$

The third equation follows from the fact that g has a period $1\|s_1$. Moreover,

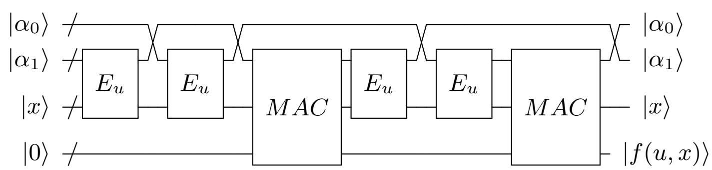

Fig. 5: Grover-meets-Simon's function f for ECBC-MAC.

$$\begin{aligned} f(k_1, x') = f(k_1, x) &\Leftrightarrow h(0,x') \oplus h(1, x' \oplus E_1(\alpha_0) \oplus E_1(\alpha_1)) = h(0,x) \oplus h(1, x \oplus E_1(\alpha_0) \oplus E_1(\alpha_1)) \\ &\Leftrightarrow E_4(E_3(E_3(\alpha_0) \oplus x')) \oplus E_4(E_3(E_3(\alpha_1) \oplus x' \oplus E_1(\alpha_0) \oplus E_1(\alpha_1))) \\ &\quad = E_4(E_3(E_3(\alpha_0) \oplus x)) \oplus E_4(E_3(E_3(\alpha_1) \oplus x \oplus E_1(\alpha_0) \oplus E_1(\alpha_1))) \\ &\Leftrightarrow \begin{cases} x' = x \\ x' = x \oplus E_1(\alpha_0) \oplus E_1(\alpha_1) \oplus E_3(\alpha_0) \oplus E_3(\alpha_1). \end{cases} \end{aligned} \tag{23}$$

Therefore, the function $f(k_1, x)$ has a period $s = s_1 \oplus s_2$, where $s_1 = E_1(\alpha_0) \oplus E_1(\alpha_1)$, $s_2 = E_3(\alpha_0) \oplus E_3(\alpha_1)$. Furthermore, the parameter $\varepsilon(f) := \max_{(u,t) \in \{0,1\}^m \times \{0,1\}^n \setminus \{0, U_s\}} Pr_x[f(u,x) = f(u, x \oplus t)]$ is bounded with overwhelming probability, assuming that E behaves as a random permutation. We will prove $\varepsilon(f) < 1/2$ with overwhelming probability. Indeed, if $\varepsilon(f) > 1/2$, there exists $(u,t) \notin \{0, U_s\}$ such that $Pr_x[f(u,x) = f(u, x \oplus t)] > 1/2$, i.e.,

$$Pr_x \begin{bmatrix} E_2(E_1(E_1(\alpha_0) \oplus x)) \oplus E_4(E_3(E_3(\alpha_0) \oplus x)) \\ \oplus E_2(E_1(E_1(\alpha_1) \oplus x \oplus E_u(\alpha_0) \oplus E_u(\alpha_1))) \oplus E_4(E_3(E_3(\alpha_1) \oplus x \oplus E_u(\alpha_0) \oplus E_u(\alpha_1))) \\ \oplus E_2(E_1(E_1(\alpha_0) \oplus x \oplus t)) \oplus E_4(E_3(E_3(\alpha_0) \oplus x \oplus t)) \\ \oplus E_2(E_1(E_1(\alpha_1) \oplus x \oplus E_u(\alpha_0) \oplus E_u(\alpha_1) \oplus t)) \oplus E_4(E_3(E_3(\alpha_1) \oplus x \oplus E_u(\alpha_0) \oplus E_u(\alpha_1) \oplus t)) = 0 \end{bmatrix} > 1/2. \tag{24}$$

This corresponds to a higher order differential for $f(u,x)$ with probability $1/2$, which only happens with negligible probability for a random choice of E [64]. Then Grover-meets-Simon algorithm can recover k_1 and k_3 with $O(2^{m/2}n)$ quantum queries and $O(m+n^2)$ qubits, using Theorem 3.

We now turn to the full key recovery.

Full key recovery. For the full key recovery, we fix two arbitrary message blocks α_0, α_1 with $\alpha_0 \neq \alpha_1$, and we define the following function (see Fig. 6)

$$\begin{aligned}\varphi : \{0,1\}^m \times \{0,1\} \times \{0,1\}^n &\to \{0,1\}^n \\ u, b, x &\mapsto \text{SUM-ECBC}(\alpha_b \| x) \oplus E_u(E_1(E_1(\alpha_b) \oplus x)). \quad (25)\end{aligned}$$

In particular, we have

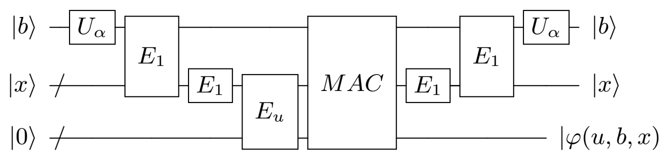

Fig. 6: Grover-meets-Simon's function φ for ECBC-MAC.

$$\begin{aligned}\varphi(k_2,b',x') = \varphi(k_2,b,x) &\Leftrightarrow E_2(E_1(E_1(\alpha_{b'}) \oplus x')) \oplus E_4(E_3(E_3(\alpha_{b'}) \oplus x')) \oplus E_2(E_1(E_1(\alpha_{b'}) \oplus x')) \\ &\qquad = E_2(E_1(E_1(\alpha_b) \oplus x)) \oplus E_4(E_3(E_3(\alpha_b) \oplus x)) \oplus E_2(E_1(E_1(\alpha_b) \oplus x)) \\ &\Leftrightarrow E_4(E_3(E_3(\alpha_{b'}) \oplus x')) = E_4(E_3(E_3(\alpha_b) \oplus x)) \\ &\Leftrightarrow \begin{cases} x' \oplus x = 0, & \text{if } b' = b; \\ x' \oplus x = E_3(\alpha_0) \oplus E_3(\alpha_1), & \text{if } b' \neq b. \end{cases}\end{aligned} \tag{26}$$

Therefore, this function is periodic if and only if $u = k_2$. From the above analysis, we can show that $\varepsilon(\varphi) \leq 1/2$ with overwhelming probability, and running the Grover-meets-Simon algorithm on the function f will return k_2. Then, we can obtain k_4 in the same way.

Forgery attack. Finally, we conclude that $\varepsilon(f) \leq 1/2$ and $\varepsilon(\varphi) \leq 1/2$, unless the SUM-ECBC has higher order differentials with probability $1/2$. If E_k is a random permutation, these differentials are only found with negligible probability. Therefore, we can apply the Grover-meets-Simon algorithm to recover k_1, k_2, k_3, and k_4 following Theorem 3. This allows one to create forgeries as follows:

1. Query the tag T_1 of $\alpha_0 \| m_1$ for an arbitrary block m_1;
2. Query the tag T_2 of $\alpha_1 \| m_1 \oplus E_1(\alpha_0) \oplus E_1(\alpha_1)$;
3. Query the tag T_3 of $\alpha_0 \| m_1 \oplus E_1(\alpha_0) \oplus E_1(\alpha_1) \oplus E_3(\alpha_0) \oplus E_3(\alpha_1)$;

4. The new tag $T_1 \oplus T_2 \oplus T_3$ is valid for $\alpha_1 \| m_1 \oplus E_3(\alpha_0) \oplus E_3(\alpha_1)$.

To break the formal notion of EUF-qCMA security, we need to produce $q+1$ valid classical message-tag pairs with only q queries to the oracle of SUM-ECBC. Let $q' = O(2^{m/2}n)$ denote the number of quantum queries made to recover k_1, k_2, k_3, and k_4. The attacker will repeat the forgery step $q'+1$ times to produce $4q'+4$ message-tag pairs, after a total of $4q'+3$ classical and quantum queries to the MAC oracle. Therefore, SUM-ECBC is broken by a quantum existential forgery attack.

2) Key recovery attack for PolyMAC. PolyMAC [47] is a Double-block Hash-then-Sum construction based on the polynomial evaluation. It uses two hashing keys $k_1, k_3 \in \{0,1\}^n$ and two encryption keys $k_2, k_4 \in \{0,1\}^m$. More precisely, the PolyMAC algorithm with two-block messages is defined as (see Fig. 7):

$$\mathrm{PolyMAC}(m_1, m_2) = E_2(k_1^2 m_1 \oplus k_1 m_2) \oplus E_4(k_3^2 m_1 \oplus k_3 m_2). \tag{27}$$

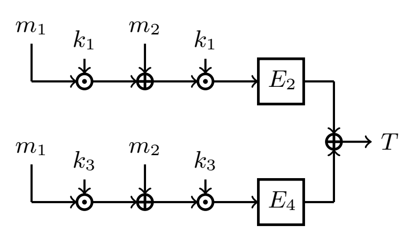

Fig. 7: PolyMAC with a two-block message.

We now give the quantum attacks to realize the partial key recovery and full key recovery, respectively.

Partial key recovery. For a two-block message, we use the same f as in the SUM-ECBC attack, with fixed blocks α_0 and α_1:

$$\begin{aligned} f : \{0,1\}^n \times \{0,1\}^n &\to \{0,1\}^n \\ u, x &\mapsto \mathrm{PolyMAC}(\alpha_0, x) \oplus \mathrm{PolyMAC}(\alpha_1, x \oplus u(\alpha_0 \oplus \alpha_1)), \end{aligned} \tag{28}$$

where $\mathrm{PolyMAC}(\alpha_b, x) = E_2(k_1^2 \alpha_b \oplus k_1 x) \oplus E_4(k_3^2 \alpha_b \oplus k_3 x)$. It satisfies $f(k_1/k_3, x) = f(k_1/k_3, x \oplus k_1(\alpha_0 \oplus \alpha_1) \oplus k_3(\alpha_0 \oplus \alpha_1))$, for any x, and $\varepsilon(f) \leq 1/2$ with overwhelming probability if E is a random function. Moreover (take $u = k_1$ as an example):

$$f(k_1, x) = \mathrm{PolyMAC}(\alpha_0, x) \oplus \mathrm{PolyMAC}(\alpha_1, x \oplus k_1(\alpha_0 \oplus \alpha_1))$$

$$
\begin{aligned}
&= g(0,x) \oplus h(0,x) \oplus g(1, x \oplus k_1(\alpha_0 \oplus \alpha_1)) \oplus h(1, x \oplus k_1(\alpha_0 \oplus \alpha_1)) \\
&= h(0,x) \oplus h(1, x \oplus k_1(\alpha_0 \oplus \alpha_1)) \\
&= h(1, x \oplus k_3(\alpha_0 \oplus \alpha_1)) \oplus h(0, x \oplus k_1(\alpha_0 \oplus \alpha_1) \oplus k_3(\alpha_0 \oplus \alpha_1)) \\
&= f(k_1, x \oplus k_1(\alpha_0 \oplus \alpha_1) \oplus k_3(\alpha_0 \oplus \alpha_1)), \qquad (29)
\end{aligned}
$$

where $\text{PolyMAC}(\alpha_b, x) = g(b,x) \oplus h(b,x)$, $g(b,x) = E_2(k_1^2\alpha_b \oplus k_1 x)$ and $h(b,x) = E_4(k_3^2\alpha_b \oplus k_3 x)$. Here the third and fourth equations follow from the fact that g has a period $1\|k_1(\alpha_0 \oplus \alpha_1)$ and h has a period $1\|k_3(\alpha_0 \oplus \alpha_1)$. It is easy to see that the function f is periodic if and only if $u = k_1/k_3$. Therefore, an application of the Grover-meets-Simon algorithm returns k_1 and k_3, with complexity $O(2^{n/2}n)$.

Full key recovery. The above attack of recovering partial keys can be generalized to be the following attack. For the full key recovery, we fix two arbitrary message blocks α_0, α_1 with $\alpha_0 \neq \alpha_1$, and we define the following function

$$
\begin{aligned}
\varphi : \{0,1\}^m \times \{0,1\} \times \{0,1\}^n &\to \{0,1\}^n \\
u, b, x &\mapsto \text{PolyMAC}(\alpha_b \| x) \oplus E_u(k_1^2\alpha_b \oplus k_1 x). \qquad (30)
\end{aligned}
$$

In particular, we have

$$
\begin{aligned}
\varphi(k_2, b', x') = \varphi(k_2, b, x) &\Leftrightarrow E_2(k_1^2\alpha_{b'} \oplus k_1 x') \oplus E_4(k_3^2\alpha_{b'} \oplus k_3 x') \oplus E_2(k_1^2\alpha_{b'} \oplus k_1 x') \\
&\quad = E_2(k_1^2\alpha_b \oplus k_1 x) \oplus E_4(k_3^2\alpha_b \oplus k_3 x) \oplus E_2(k_1^2\alpha_b \oplus k_1 x) \\
&\Leftrightarrow E_4(k_3^2\alpha_{b'} \oplus k_3 x') = E_4(k_3^2\alpha_b \oplus k_3 x) \\
&\Leftrightarrow \begin{cases} x' \oplus x = 0, & \text{if } b' = b; \\ x' \oplus x = k_3(\alpha_0 \oplus \alpha_1), & \text{if } b' \neq b. \end{cases} \qquad (31)
\end{aligned}
$$

Note that this function satisfies $\varphi(k_2, 0, x) = \varphi(k_2, 1, x \oplus k_3(\alpha_0 \oplus \alpha_1))$ and $\varepsilon(\varphi) \leq 1/2$, with the same arguments as previously. Therefore, we can apply the Grover-meets-Simon algorithm to recover k_1, k_2, k_3, and k_4. Again, this leads to a forgery attack.

3) Key recovery attack for GCM-SIV2. GCM-SIV2 is a provably secure authenticated encryption mode designed by Iwata and Minematsu [48] as a double-block-hash version of GCM-SIV. For simplicity, we focus on the authentication part of GCM-SIV2, and the tag with an l-block message and a nonce N is defined as follows

$$
\text{GCM-SIV2}(N, M) = E_1(\Sigma(M)) \oplus E_2(\Theta(M)) \| E_3(\Sigma(M)) \oplus E_4(\Theta(M)), \qquad (32)
$$

where

$$
\begin{aligned}
\Sigma(M) &= N \oplus l \odot H_1 \oplus \bigoplus_{i=1}^{l} m_i \odot H_1^{l+2-i} \\
\Theta(M) &= N \oplus l \odot H_2 \oplus \bigoplus_{i=1}^{l} m_i \odot H_2^{l+2-i}. \qquad (33)
\end{aligned}
$$

The structure of the authentication part of GCM-SIV2 is similar to the structure of SUM-ECBC, where the block cipher calls E_1 and E_3 are replaced by multiplication by hash keys H_1 and H_2. Thus, we can essentially repeat the above attack to recover the full key, with $O(2^{n/2}n)$ quantum queries and $O(m+n^2)$ qubits.

4) Key recovery attack for 2K-ECBC_Plus. 2K-ECBC_Plus [49] is the sequential mode of block cipher-based instantiation of two-keyed DbHtS. In full generality, there are three keys k_1, k_2, and k_3. The two-block message m_1, m_2 is processed as

$$\text{2K-ECBC\_Plus}(m_1, m_2) = E_3(fix0(E_1(E_1(m_1) \oplus m_2))) \oplus E_3(fix1(E_2(E_2(m_1 \oplus m_2)))), \tag{34}$$

where the functions $fix0$ and $fix1$ take an n-bit binary string x and return x with its least significant bit set to 0 and 1 respectively. This falls into our framework, and then we can recover k_1, k_2, and k_3 by applying the Grover-meets-Simon algorithm.

5 Conclusion

In this paper, we give secret state recovery attacks and key recovery attacks for some BBB MACs in a quantum setting, leading to forgery attacks. The first kind of attack costs $O(n)$ quantum queries by using Simon algorithm, where n is the size of the block. Notice that our secret recovery attack for HPxHP and HPxNP can also recover the full key $K = (k_1, k_2)$. It gives an exponential speedup compared with the classical attack. The second kind of attack costs $O(2^{n/2}n)$ quantum queries by applying Grover-meets-Simon algorithm. This leads to a better analysis of BBB MACs, that is, the complexity of some previous key-recovery attacks reduces from $O(2^n n)$ to $O(2^{n/2}n)$. Our results show that these MAC constructions cannot achieve security beyond the birthday bound of $O(2^{n/2})$ in the quantum model.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (Grant Nos. 62272056, 61972048, 61976024).

References

1. Liu H L, Wu Y S, Wan L C, et al. Variational quantum algorithm for the Poisson equation. Physical Review A, 104(2): 022418 (2021).

2. Wan L C, Yu C H, Pan S J, et al. Asymptotic quantum algorithm for the Toeplitz systems. Physical Review A, 97(6): 062322 (2018).
3. Wan L C, Yu C H, Pan S J, et al. Block-encoding-based quantum algorithm for linear systems with displacement structures. Physical Review A, 104(6): 062414 (2021).
4. S. Lloyd, M. Mohseni, and P. Rebentrost, Quantum principal component analysis, Nature Physics, 10: 631-633 (2014).
5. I. Cong and L. Duan, Quantum discriminant analysis for dimensionality reduction and classification, New Journal of Physics, 18: 073011 (2016).
6. Pan S J, Wan L C, Liu H L, et al. Improved quantum algorithm for A-optimal projection. Physical Review A, 102(5): 052402 (2020).
7. Pan S J, Wan L C, Liu H L, et al. Quantum algorithm for Neighborhood Preserving Embedding. Chinese Physics B, 31(6): 060304 (2022).
8. Yu C H, Gao F, Lin S, et al. Quantum data compression by principal component analysis. Quantum Information Processing, 18(8): 1-20 (2019).
9. N. Wiebe, D. Braun, and S. Lloyd, Quantum algorithm for data fitting, Physical Review Letters, 109: 050505 (2012).
10. M. Schuld, I. Sinayskiy, and F. Petruccione, Prediction by linear regression on a quantum computer, Physical Review A, 94: 022342 (2016).
11. G. Wang, Quantum algorithm for linear regression, Physical review A, 96: 012335 (2017).
12. Yu C H, Gao F, Wen Q. An improved quantum algorithm for ridge regression. IEEE Transactions on Knowledge and Data Engineering, (2019).
13. Yu C H, Gao F, Liu C, et al. Quantum algorithm for visual tracking. Physical Review A, 99(2): 022301 (2019).
14. M C. Guo, H L. Liu, Y M. Li, W M. Li, F. Gao, S J. Qin, Q Y. Wen, Quantum algorithms for anomaly detection using amplitude estimation, Physica A: Statistical Mechanics and its Applications 604: 127936 (2022).
15. Wang, H., Xue, Y., Qu, Y. et al. Multidimensional Bose quantum error correction based on neural network decoder. npj Quantum Inf 8, 134 (2022).
16. P. Rebentrost, M. Mohseni, and S. Lloyd, Quantum support vector machine for big data classification, Physical Review Letters, 113: 130503 (2014).
17. M. Schuld, I. Sinayskiy, and F. Petruccione. Quantum computing for pattern classification. in Pacific Rim International Conference on Artificial Intelligence (Springer, 2014), pp. 208-220.
18. Huang R, Tan X, Xu Q. Variational quantum tensor networks classifiers. Neurocomputing, 452: 89-98 (2021).
19. Huang R, Tan X, Xu Q. Learning to Learn Variational Quantum Algorithm. IEEE Transactions on Neural Networks and Learning Systems, (2022).
20. Wang, H., Xue, Y., Qu, Y. et al. Multidimensional Bose quantum error correction based on neural network decoder. npj Quantum Inf 8, 134 (2022).
21. Yu C H, Gao F, Wang Q L, et al. Quantum algorithm for association rules mining. Physical Review A, 94(4): 042311 (2016).
22. Shor, P.W.: Algorithms for quantum computation: Discrete logarithms and factoring. In: 35th Annual Symposium on Foundations of Computer Science. pp. 124-134. IEEE Computer Society (1994).
23. Grover, L.K.: A Fast Quantum Mechanical Algorithm for Database Search. In: Miller, G.L. (ed.) Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996. pp. 212-219. ACM (1996).

24. Li, Z., Cai, B., Sun, H. et al. Novel quantum circuit implementation of Advanced Encryption Standard with low costs. Sci. China Phys. Mech. Astron. 65, 290311 (2022).
25. Kuwakado, H., Morii, M.: Quantum distinguisher between the 3-round Feistel cipher and the random permutation. In: 2010 IEEE International Symposium on Information Theory Proceedings (ISIT), June 2010, pp. 2682-2685 (2010).
26. Kuwakado, H., Morii, M.: Security on the quantum-type even-mansour cipher. In: ISITA. pp. 312-316. IEEE (2012).
27. Bin-Bin Cai, Yusen Wu, Jing Dong, Su-Juan Qin, Fei Gao, Qiao-Yan Wen, Quantum Attacks on 1K-AES and PRINCE, The Computer Journal, 2022;, bxab216, https://doi.org/10.1093/comjnl/bxab216
28. Dong, X., Dong, B. and Wang, X. Quantum attacks on some feistel block ciphers. Des. Codes Cryptogr. 88, 1179-1203 (2020).
29. Kaplan M., Leurent G., Leverrier A., et al.: Breaking symmetric cryptosystems using quantum period finding. In: CRYPTO 2016, Part II, pp. 207-237 (2016).
30. Santoli T., Schaffner C.: Using simon's algorithm to attack symmetric-key cryptographic primitives. Quantum Inf. Comput. 17, 65-78 (2017).
31. Sun H W, Cai B B, Qin S J, et al. Quantum Attacks on Type-1 Generalized Feistel Schemes. Cryptology ePrint Archive, (2023).
32. Bonnetain X, Schrottenloher A, Sibleyras F. Beyond quadratic speedups in quantum attacks on symmetric schemes. arXiv preprint arXiv:2110.02836, (2021).
33. Dong, X., Wang, X. Quantum key-recovery attack on Feistel structures. Sci. China Inf. Sci. 61, 102501 (2018).
34. G. Leander, A. May. Grover Meets Simon - Quantumly Attacking the FX-construction, Advances in Cryptology - ASIACRYPT, pp. 161-178 (2017).
35. Sun, HW., Wei, CY., Cai, BB. et al. Improved BV-based quantum attack on block ciphers. Quantum Inf Process 22, 9 (2023).
36. Xie, H., Yang, L. Using Bernstein-Vazirani algorithm to attack block ciphers. Des. Codes Cryptogr. 87, 1161-1182 (2019).
37. X. Bonnetain, G. Leurent,M. N.-Plasencia, A. Schrottenloher. Quantum linearization attacks. Advances in Cryptology - ASIACRYPT 2021, LNCS vol, 13090, pp. 422-452, (2021).
38. Bernstein, E., Vazirani, U.V.: Quantum complexity theory. SIAM J. Comput. 26(5), 1411-1473 (1997).
39. Bellare, M., Kilian, J., Rogaway, P.: The security of the cipher block chaining message authentication code. J. Comput. Syst. Sci. 61(3), 362-399 (2000).
40. Dworkin, M.: Recommendation for block cipher modes of operation: the CMAC mode for authentication. NIST Special Publication 800-38B, National Institute for Standards and Technology, May (2005).
41. Iwata, T., Kurosawa, K.: OMAC: one-key CBC MAC. In: Johansson, T. (ed.) FSE 2003. LNCS, vol. 2887, pp. 129-153. Springer, Heidelberg (2003).
42. McGrew, D.A., Viega, J.: The security and performance of the Galois/Counter Mode (GCM) of operation. In: Canteaut, A., Viswanathan, K. (eds.) INDOCRYPT 2004. LNCS, vol. 3348, pp. 343-355. Springer, Heidelberg (2004).
43. Bogdanov, A., et al.: PRESENT: an ultra-lightweight block cipher. In: Paillier, P., Verbauwhede, I. (eds.) CHES 2007. LNCS, vol. 4727, pp. 450-466. Springer, Heidelberg (2007).
44. Guo, J., Peyrin, T., Poschmann, A., Robshaw, M.: The LED block cipher. In: Preneel, B., Takagi, T. (eds.) CHES 2011. LNCS, vol. 6917, pp. 326-341. Springer, Heidelberg (2011).

45. Banik, S., Pandey, S.K., Peyrin, T., Sasaki, Y., Sim, S.M., Todo, Y.: GIFT: a small present. In: Fischer, W., Homma, N. (eds.) CHES 2017. LNCS, vol. 10529, pp. 321-345. Springer, Cham (2017).
46. Yasuda, K.: The sum of CBC macs is a secure PRF. In: Topics in Cryptology - CT - RSA 2010. pp. 366-381 (2010).
47. Kim, S., Lee, B., Lee, J.: Tight security bounds for Double-Block Hash-then-Sum MACs. In: Advances in Cryptology - EUROCRYPT 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12105, pp. 435-465. Springer (2020).
48. Iwata, T., Minematsu, K.: Stronger security variants of GCM-SIV. IACR Trans. Symmetric Cryptol. 2016(1), 134-157 (2016).
49. Datta, N., Dutta, A., Nandi, M., Paul, G.: Double-block hash-then-sum: A paradigm for constructing BBB secure PRF. IACR Trans. Symmetric Cryptol. 2018(3), 36-92 (2018).
50. Cogliati, B., Jha, A., Nandi, M.: How to build optimally secure prfs using block ciphers. In: Moriai, S., Wang, H. (eds.) Advances in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Part I. Lecture Notes in Computer Science, vol. 12491, pp. 754-784. Springer (2020).
51. Black, J., Rogaway, P.: A block-cipher mode of operation for parallelizable message authentication. In: Knudsen, L.R. (ed.) EUROCRYPT. LNCS, vol. 2332, pp. 384-397. Springer (2002).
52. Naito, Y.: Blockcipher-based MACs: Beyond the birthday bound without message length. In: Advances in Cryptology - ASIACRYPT 2017, Proceedings, Part III. pp. 446-470 (2017).
53. Naito, Y.: Full PRF-secure message authentication code based on tweakable block cipher. In: Provable Security - 9th International Conference, ProvSec 2015. pp. 167-182 (2015).
54. Guo, T., Wang, P., Hu, L., Ye, D.: Attacks on beyond-birthday-bound macs in the quantum setting. In: PQCrypto. Lecture Notes in Computer Science, vol. 12841, pp. 421-441. Springer (2021).
55. Zhang, Y.: Using an error-correction code for fast, beyond-birthday-bound authentication. In: Nyberg, K. (ed.) CT-RSA 2015. LNCS, vol. 9048, pp. 291-307. Springer, Heidelberg (2015).
56. 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).
57. Kan Yasuda. PMAC with Parity: Minimizing the Query-Length Influence. In CT-RSA 2012, volume 7178 of LNCS, pages 203-214. Springer, (2012).
58. Nielsen, M.A., Chuang, I.: Quantum computation and quantum information. AAPT (2002).
59. Bonnetain, X.: Tight bounds for simon's algorithm. In: LATINCRYPT 2021. vol. 12912, pp. 3-23. Springer (2021).
60. Simon, D.R.: On the power of quantum computation. SIAM J. Comput. 26(5), 1474-1483 (1997).
61. Boneh, D., Zhandry, M.: Quantum-secure message authentication codes. In: Johansson, T., Nguyen, P.Q. (eds.) Advances in Cryptology - EUROCRYPT 2013. Lecture Notes in Computer Science, vol. 7881, pp. 592-608. Springer (2013).
62. Boneh, D., Zhandry, M.: Secure signatures and chosen ciphertext security in a quantum computing world. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 361-379. Springer, Heidelberg (2013).

63. Leurent, G., Nandi, M., Sibleyras, F.: Generic attacks against beyond-birthday-bound MACs. In: Advances in Cryptology - CRYPTO 2018, Proceedings, Part I. pp. 306-336 (2018).
64. Daemen, J., Rijmen, V.: Probability distributions of correlation and differentials in block ciphers. J. Math. Crypt. 1(3), 221-242 (2007).