Li et al. *EPJ Quantum Technology* (2025) 12:67
https://doi.org/10.1140/epjqt/s40507-025-00374-x

EPJ Quantum Technology
a SpringerOpen Journal

EPJ.org

RESEARCH Open Access

Quantum attacks on Sum of Even-Mansour construction utilizing online classical queries

Zhenqiang Li[1], Shuqin Fan[1*], Fei Gao[2], Yonglin Hao[1], Hongwei Sun[3], Xichao Hu[1] and Dandan Li[4]

*Correspondence: fansq@sklc.org
[1]State Key Laboratory of Cryptology, Beijing, 100878, China
Full list of author information is available at the end of the article

Abstract

The Sum of Even-Mansour (SoEM) construction, proposed by Chen et al. at Crypto 2019, has become the basis for designing some symmetric schemes, such as the nonce-based MAC scheme nEHtM$_p$ and the nonce-based encryption scheme CENCPP*. In this paper, we make the first attempt to study the quantum security of SoEM under the Q1 model where the targeted encryption oracle can only respond to classical queries rather than quantum ones. Firstly, we propose a quantum key recovery attack on SoEM21 with a time complexity of $\tilde{O}(2^{n/3})$ along with $O(2^{n/3})$ online classical queries. Compared with the current best classical result which requires $O(2^{2n/3})$ time, our method offers a quadratic time speedup while maintaining the same number of queries. The time complexity of our attack is less than that observed for quantum exhaustive search by a factor of $2^{n/6}$. We further propose classical and quantum key recovery attacks on the generalized SoEMs1 construction (consisting of $s \geq 2$ independent public permutations), revealing that the application of quantum algorithms can provide a quadratic acceleration over the pure classical methods. Our results also imply that the quantum security of SoEM21 cannot be strengthened merely by increasing the number of permutations.

Keywords: Offline Simon's algorithm; SoEM construction; Query complexity; Birthday-bound

1 Introduction

With the rapid development of quantum computation, well designed quantum algorithms have shown significant speedup over classical ones in handling certain problems, such as linear systems [1–3], dimensionality reduction [4–6], and so on [7–10]. Some quantum algorithms have become serious threats to the security of classical cryptographic schemes. In the field of asymmetric cryptography, Shor's algorithm [11] can solve factorization and discrete logarithms in polynomial time, which will completely break the currently used public-key systems, such as RSA (Rivest-Shamir-Adleman) and ECC (Elliptic Curve Cryptography). For symmetric cryptography, Grover's algorithm [12] offers a quadratic speedup on an exhaustive search attack. Using Simon's algorithm [13], many symmetric ciphers, such as the 3-round Feistel construction [14], Even-Mansour construction [15], CBC-MAC [16], PMACX [17], PMAC with parity [17], can be broken in polynomial time

Springer

if the adversary is allowed to make quantum queries to the encryption oracle. In these attacks, the target issue (distinguishing or key recovery) is reduced to the period-finding problem, which can be solved efficiently using Simon's algorithm.

Based on the capabilities of the adversary, Zhangdry et al. [18] grouped quantum attacks into two models named as Q1 and Q2. In the Q1 model, the adversary can employ quantum computers for any offline computation, but only online classical queries are allowed. In the Q2 model, the adversary can not only do quantum computations offline but make online superposition queries as well. Quantum cryptanalysis results of the Q2 model often enjoy lower complexities, but the ability to make superposition queries is impractical. On the contrary, quantum attacks for the Q1 model are usually considered as more practical: once a large-scale fault-tolerant quantum computer becomes available, the Q1 attacks can pose real threats to the primitives immediately.

The quantum algorithms available for constructing attacks in the Q1 model are limited. The quantum exhaustive search with Grover's algorithm is the most used algorithm. Some quantum collision finding algorithms or clawing finding algorithms [19, 20] can also be applied to perform the attack on symmetric ciphers in the Q1 model with $O(2^{n/3})$ complexity. However, they often have massive quantum hardware requirements (e.g., exponential qubits and classical memory). To address these limitations, inspired by Grover-meets-Simon algorithm in the Q2 model [21], Bonnetain et al. [22] proposed offline Simon's algorithm to perform Q1 attack on FX construction with $\tilde{O}(2^{(n+k)/3})$ time, $O(n)$ qubits and classical memory, instead of $O(2^{2(n+k)/3})$ time and $O(2^{(n+k)/3})$ classical memory from a classical attack ($n + k$ is the length of key). Note that $\tilde{O}$ represents the complexity notation which hides lower-order terms but focuses on high-order terms that dominate the complexity (e.g., $O(n^3 2^{(n+k)/3})$ can be represented as $\tilde{O}(2^{(n+k)/3})$). However, the practical application of the offline Simon's algorithm also faces some non-trivial challenges. On the one hand, when it comes to reducing the number of classical queries, the input is required to be of the form $(x\|c)$ (c is a fixed constant). On the other hand, a conditional periodic function, which is a periodic function with part of the secret keys as the period when the guess of another key value is correct, needs to be constructed. Thus, while the offline Simon's algorithm is a powerful tool, it is only applied to conduct Q1 attacks on the Even-Mansour construction [22], the 2-round iterated Even-Mansour construction (2-IEM) [23], the 4-round iterated Even-Mansour construction with two keys (4-IEM) [24], the 2-round iterated Even-Mansour construction [23], and the 2XOR-Cascade construction [25]. Whether the offline Simon's algorithm can be extended for attacking other symmetric ciphers is an open question that deserves attention.

Related works and motivations In 2019, Chen et al. [26] introduced the Sum of Even-Mansour (SoEM) construction, which is built by the XOR of two instances of the Even-Mansour cipher [27]. SoEM21 is a specific SoEM variant consisting of two n-bit permutations controlled by one n-bit key. For the classical setting, Chen et al. [26] proved that the SoEM21 cipher can ensure its security when the number of queries is bounded as $O(2^{n/2})$. In 2022, Shinagawa and Iwata [28] presented the SoEMs1 cipher, a natural extension of SoEM21 cipher, and demonstrated that a Q2 adversary can recover the key of the SoEM21 and SoEMs1 ciphers in polynomial time by using Simon's algorithm. Such a key-recovery method was later employed by Zhang [29] and applied to attacks under the related key settings. It is noticeable that the existing quantum key recovery attacks on SoEM21 and

SoEM*s*1 [28, 29] rely on Simon's algorithm and can only be applicable for the Q2 model. There is a lack of knowledge whether a Q1 adversary can carry out quantum key recovery attacks on SoEM21 and SoEM*s*1.

As the SoEM construction serves as a foundational component in certain lightweight symmetric constructions (e.g., nEHtM$_p$ and CENCPP*), its vulnerability to quantum attacks under the Q1 model could undermine entire classes of real-world systems. The Q1 model's significance lies in its allowance for classical online queries, which mirror adaptive attack strategies where adversaries interact with encryption oracles in real time. The Q1 model captures the hybrid quantum-classical threat landscape that modern systems actually face. Thus, analyzing the quantum security of SoEM in the Q1 model provides critical insights for developing quantum-resistant symmetric schemes.

Contributions In this paper, we study the quantum security of SoEM construction in the Q1 model for the first time. We successfully construct a conditional periodic function by exploiting the construction of SoEM21, and then apply the offline Simon's algorithm to this constructed function to recover the key. For the n-bit key, our Q1 key recovery attack requires a time complexity of $O(n^3 2^{n/3})$ along with $O(2^{n/3})$ classical queries. Compared with the best classical attack which requires $O(2^{2n/3})$ time in [26], our attack offers a quadratic speedup, with the same query complexity. Compared with the quantum exhaustive key search with Grover's algorithm, our attack reduces the time complexity by a factor of $2^{n/6}$. Besides, our attack requires $O(n^2)$ qubits and $O(n)$ classical memory, while the classical attack requires $O(2^{n/3})$ classical memory. For the generalized SoEM*s*1, where no efficient classical attack is currently known, we construct a classical attack model to evaluate security and demonstrate the acceleration effect of quantum attacks. This model serves as a baseline to quantify the advantage of quantum algorithms, highlighting the vulnerability of SoEM*s*1 under the Q1 model. The Q1 attack on SoEM*s*1 employs the idea of that on SoEM21 and provides a quadratic time speedup over the classical attack. A complexity comparison of quantum attacks for the SoEM21, SoEM*s*1, 2-IEM and 4-IEM constructions is shown in Table 1.

Outline The remainder of this paper is structured as follows. In Sect. 2, we briefly review some preliminaries. Section 3 is dedicated to deducing the key recovery attacks on SoEM21 ciphers in the Q1 model. In Sect. 4, we propose the classical attack on SoEM*s*1 ciphers, and give a quantum key recovery attack in the Q1 model. Finally, in Sect. 5, a concise conclusion is drawn and discussed.

Table 1 Comparisons of quantum attacks for SoEM21, SoEMs1, 2-IEM and 4-IEM constructions

Structures	Setting	Queries	Time	Qubits	Classical memory
SoEM21	Classical [26]	$O(2^{n/3})$	$O(2^{2n/3})$	–	$O(2^{n/3})$
	Q2 [28]	$O(n)$	$O(n^3)$	$O(n^2)$	O(1)
	Q1 (Sect. 3)	$O(2^{n/3})$	$O(n^3 2^{n/3})$	$O(n^2)$	$O(n)$
SoEMs1	Classical (Sect. 4)	$O(2^{n/3})$	$O(2^{2n/3})$	–	$O(2^{n/3})$
	Q2 [28]	$O(n)$	$O(n^3)$	$O(n^2)$	O(1)
	Q1 (Sect. 4)	$O(2^{n/3})$	$O(n^3 2^{n/3})$	$O(n^2)$	$O(n)$
2-IEM	Q1 [23]	$O(2^{2n/3})$	$O(n^3 2^{2n/3})$	$O(n^2)$	$O(n)$
4-IEM	Q1 [24]	$O(2^n)$	$O(n2^{n/2})$	$O(n^2)$	$O(n)$

2 Preliminaries

2.1 Notation

Let $\{0,1\}^n$ denote the set of all n-bit strings, where n is a positive integer. Let $Perm(n)$ represent the set of all permutations on $\{0,1\}^n$. Meanwhile, let $Func(n)$ stand for the set of all functions from $\{0,1\}^n$ to $\{0,1\}^n$. Let $|X|$ be the number of elements contained in set X. For two bit strings a and b, $a\|b$ indicates their concatenation. If a and b have the same dimension, then their bit-wise XOR is denoted as $a \oplus b$ and their inner product is marked as $a \cdot b$.

2.2 Basics of quantum computation

Since this paper applies quantum algorithms as black boxes to analyze the symmetric cryptosystems, we introduce some necessary notations about quantum computation. For a more extensive presentation, please refer to [30].

As the fundamental building block in quantum computation, a single qubit can be regarded as a vector in a two-dimensional complex vector space, where $|0\rangle$ and $|1\rangle$ together form an orthonormal basis. Classical bits can exist only in the states of 0 or 1, while qubits can exist in a superposition state $|\varphi\rangle = \alpha|0\rangle + \beta|1\rangle$ with the condition that $\|\alpha\|^2 + \|\beta\|^2 = 1$. More generally, an n-qubit corresponds to a vector in a 2^n-dimensional complex vector space, which can be uniquely written as $|\varphi\rangle = \sum_{x\in\{0,1\}^n} \alpha_x|x\rangle$ with the condition that $\sum_{x\in\{0,1\}^n} \|\alpha_x\|^2 = 1$.

A quantum gate, represented by a unitary matrix U, transforms one quantum state $|\varphi_1\rangle$ to another state $|\varphi_2\rangle = U|\varphi_1\rangle$. For example, the single qubit gate H, called Hadamard gate, can turn $|0\rangle$ and $|1\rangle$ to $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and $|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ respectively. For an n-qubit system, $H^{\otimes n}$ represents performing the Hadamard gate H to each qubit. If the n-qubit is $|x\rangle$ ($x \in \{0,1\}^n$), then

$$H^{\otimes n}|x\rangle = \frac{1}{\sqrt{2^n}} \sum_{y\in\{0,1\}^n} (-1)^{x\cdot y}|y\rangle.$$

The single qubit gate X realizes the function $|a\rangle \to |a \oplus 1\rangle$. The two-qubit gate CNOT realizes the function $|a\rangle|b\rangle \to |a\rangle|b \oplus a\rangle$.

Various quantum algorithms can be performed using a quantum circuit, which represents a sequence of quantum gates applied to a set of qubits. The reversibility of quantum gates ensures that any quantum circuit is reversible. In a quantum circuit, certain quantum computations are performed on the input state $|\varphi\rangle$. The computation result is copied to an output register by utilizing CNOT gates. Subsequently, an uncomputation process, which consists of reversing the same operations, is performed to restore the initial state $|0\rangle$ of the ancilla qubits. The uncomputation of a unitary operation U corresponds to employing its adjoint (i.e., conjugate transpose) operator $U^\dagger$.

Many quantum algorithms rely on efficient access to quantum oracles. A quantum oracle for a function f is represented as a unitary operator

$$O_f|x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle,$$

and this operator can be queried in superposition, that is, $O_f \sum_x |x\rangle|y\rangle = \sum_x |x\rangle|y \oplus f(x)\rangle$. If we have access to O_f, we claim that we can make quantum superposition queries to f.

Algorithm 1: Grover's algorithm [12]

Input: Oracle $O_f : |x\rangle \to (-1)^{f(x)}|x\rangle$ with $f : \{0,1\}^n \to \{0,1\}$.

Output: x such that $f(x) = 1$.

1 Initialize an n-qubit quantum state $|0\rangle^{\otimes n}$.

2 Apply $H^{\otimes n}$:

$$\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle = |\varphi\rangle.$$

3 Perform Grover iteration $R \approx \frac{\pi}{4}\sqrt{2^n/t}$ times:

$$[(2|\varphi\rangle\langle\varphi| - I)O_f]^R|\varphi\rangle.$$

4 Measure to return a x.

2.3 Some typical quantum algorithms

When solving specific problems, quantum algorithms may have significant advantages over their classical counterparts. In this part, we make brief introductions to the typical quantum algorithms used in the cryptanalysis of symmetric-key primitives.

Grover's Algorithm [12], also known as Grover search, is regarded as the main threat to the symmetric-key primitives bringing quadratic speedup to the exhaustive search of the secret key. It can solve the following Grover's problem in Problem 1.

Problem 1 *(Grover's problem)* Given a set X ($|X| = t$), let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function such that $f(x) = 1$ if and only if $x \in X$. Find a x in X.

In the classical setting, the time complexity for solving Problem 1 is $O(2^n/t)$. But Grover's algorithm in Algorithm 1 can give a solution with a probability close to 1 in time $O(\sqrt{2^n/t})$, using $O(n)$ qubits.

Furthermore, Grover's algorithm has been generalized as quantum amplitude amplification (QAA) technique, as described in Theorem 1 [31].

Theorem 1 [31] *Let $\mathcal{A}$ be any quantum algorithm on q qubits that does not employ measurement. Let $\mathcal{B} : \{0,1\}^q \to \{0,1\}$ be a function that categorizes the outcomes of $\mathcal{A}$ as either good state or bad state. Let $p > 0$ be the initial success probability that the measurement of $\mathcal{A}|0\rangle$ is good. Set $t = \lceil \frac{\pi}{4\theta} \rceil$, where θ is defined as $sin^2\theta = p$ and $0 < \theta < \frac{\pi}{2}$. Besides, the unitary operation $Q = \mathcal{A}\mathcal{S}_0\mathcal{A}^{-1}\mathcal{S}_\mathcal{B}$ is defined, where $\mathcal{S}_\mathcal{B}$ changes the sign of the good state, i.e.,*

$$\mathcal{S}_\mathcal{B} : |x\rangle \to \begin{cases} -|x\rangle, \text{if } \mathcal{B}(x) = 1, \\ |x\rangle, \text{if } \mathcal{B}(x) = 0, \end{cases}$$

while $\mathcal{S}_0 = 2|0\rangle\langle 0| - I$ changes the sign of the amplitude exclusively when it isn't the zero state $|0\rangle$. Eventually, after performing the computation of $Q^t\mathcal{A}|0\rangle$, the measurement generates a good state with probability at least $Max\{1-p, p\}$.

Simon's Algorithm [13] is a polynomial time algorithm for finding the period of a periodic function. We demonstrate the period finding problem as the Simon's problem in Problem 2:

Problem 2 *(Simon's problem)* Given a Boolean function $f : \{0,1\}^n \to \{0,1\}^n$. Find $s \in \{0,1\}^n \setminus \{0^n\}$, such that for any $x, y \in \{0,1\}^n$, $f(x) = f(y) \Leftrightarrow [x \oplus y = s \text{ or } x = y]$.

In the classical setting, Problem 2 can be solved with $O(2^{n/2})$ queries. In the Q2 model where quantum superposition queries to f are allowed, Simon's algorithm in Algorithm 2 can reduce the complexity to polynomial level: it requires $O(n^3)$ time, $O(n)$ quantum queries and $O(n)$ qubits. Note that a boolean linear equation of size n can be solved in time $O(n^3)$ using a classical solver.

For the Q1 model where only classical queries to f are allowed, Bonnetain et al. [22] proposed a simplified version of Simon's algorithm to determine the period s (see Algo-

Algorithm 2: Simon's algorithm [13]

Input: Oracle $O_f : |x\rangle|y\rangle \to |x\rangle|y \oplus f(x)\rangle$ with a periodic function $f : \{0,1\}^n \to \{0,1\}^n$.

Output: s.

1 Initialize two registers $|0\rangle^{\otimes n}|0\rangle^{\otimes n}$.

2 Apply quantum gate $H^{\otimes n}$ on the first register:

$$\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle|0\rangle.$$

3 Apply oracle O_f:

$$\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle|f(x)\rangle.$$

4 Measure the second register to get:

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

5 Reapply quantum gate $H^{\otimes n}$:

$$\frac{1}{\sqrt{2^{n+1}}} \sum_{y\in\{0,1\}^n} (-1)^{y\cdot z}(1 + (-1)^{y\cdot s})|y\rangle.$$

6 Measure to yield a random vector y such that $y \cdot s = 0$.

7 Run step 1 to step 6 $O(n)$ times to generate $n - 1$ independent vectors y.

8 Perform the Gaussian elimination classically for solving a boolean linear equation to recover s.

rithm 5 in the Appendix). Notably, this algorithm requires $O(2^n)$ classical queries, $O(n^3)$ time and $O(n^2)$ qubits.

Grover-meets-Simon algorithm In 2017, Leander and May [21] combined Simon's algorithm with Grover's algorithm, called Grover-meets-Simon algorithm, to devise an attack on the FX construction [32] defined as follow:

$$FX(x) = E_{k_0}(x \oplus k_1) \oplus k_2 \tag{1}$$

where $k_0 \in \{0,1\}^k, k_1, k_2 \in \{0,1\}^n$ and E is a block cipher. They construct the function $f(i,x) = FX(x) \oplus E_i(x)$. For $i = k_0$, the function $f(i,x)$ is a periodic function with the period k_1. Otherwise, for $i \neq k_0$, the function $f(i,x)$ is a random permutation. Based on this, they apply Grover's algorithm as an outer loop to search the correct i, and Simon's algorithm as an inner loop to judge whether the function $f(i,x)$ has a period. That is, in every Grover iteration, c ($c = O(n)$) parallel Simon's algorithms are applied to judge whether $f(i,x)$ is periodic, which requires $O(n)$ quantum queries to $f(i,x)$ and $O(n^3)$ time for solving boolean linear equations. Since the number of Grover iterations is $O(2^{k/2})$, the Grover-meets-Simon algorithm can recover the key k_0 in time $O(n^3 2^{k/2})$, with $O(n2^{k/2})$ quantum queries and $O(n^2)$ qubits. After that, k_1 can be determined by performing Simon's algorithm to the function $f(k_0,x)$. The computation of k_2 is trivial.

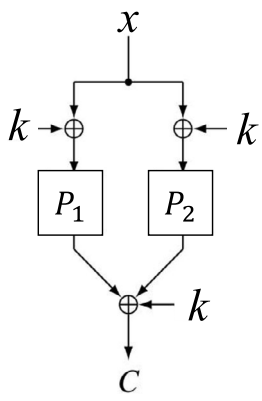
Offline Simon's algorithm In 2019, Bonnetain et al. [22] introduced the offline Simon's algorithm, enhancing the quantum attack proposed by Leander and May [21] on FX constructions. Its primary idea is to separate the online quantum queries from offline quantum computations, and iteratively reusing $|\psi_g\rangle = \bigotimes_{j=1}^{c} \sum_{x\in\{0,1\}^n} |x\rangle|FX(x)\rangle$ ($c = O(n)$) prepared by the online queries. $|\psi_g\rangle$ can be prepared with c quantum queries to $FX(x)$. Since each Grover iteration requires $O(n)$ quantum queries to $E_i(x)$ and $O(n^3)$ time for solving boolean linear equations, in the Q2 model, the offline Simon's algorithm reduces the number of quantum queries to $FX(x)$ from $O(n2^{k/2})$ to $O(n)$ while remaining $O(n^3 2^{k/2})$ time complexity and $O(n^2)$ qubits.

In the Q1 model, Bonnetain et al. [22] replaced the quantum queries to $FX(x)$ with classical queries, and partitioned the n-bit key k_1 into a u-bit k_1^l and a $(n-u)$-bit k_1^r. They prepared the quantum state $|\psi_g\rangle = \bigotimes_{j=1}^{c} \sum_{x\in\{0,1\}^u} |x\rangle|FX(x\|0^{n-u})\rangle$ ($c = O(n)$) by making $O(2^u)$ classical queries, and then performed the offline Simon's algorithm to recover k_0 and k_1^r. Subsequently, Simon's algorithm in the Q1 model (i.e., Algorithm 5 in the Appendix) is applied to recover k_1^l. As a result, in the Q1 model, the key k_0, k_1 and k_2 can be recovered with $D = O(2^u)$ classical queries and $T = O(n^3 2^{(k+n-u)/2})$ time. The tradeoff between the number of classical queries D and the time complexity T is $DT^2 = \tilde{O}(2^{n+k})$, which balances at $D = T = \tilde{O}(2^{(n+k)/3})$. For a given D, we get the time $T = \tilde{O}(2^{(n+k)/2}/\sqrt{D})$, providing a quadratic speedup over the best classical attack. The offline Simon's algorithm developed by Bonnetain et al. [22] offers a powerful tool for investigating the quantum security of symmetric schemes.

2.4 SoEM21, SoEMs1 and the Q2 cryptanalysis results

In Crypto 2019, Chen et al. [26] introduced the sum of Even-Mansour (SoEM) construction, which combines two instances of the Even-Mansour cipher using an XOR operation.

Figure 1 The SoEM21 cipher



As a variant of SoEM, the SoEM21 cipher (see Fig. 1) is defined as

$$SoEM21(x) = P_1(x \oplus k) \oplus P_2(x \oplus k) \oplus k, \tag{2}$$

where P_1 and P_2 are two independent public permutations in $Perm(n)$, and the key $k \in \{0,1\}^n$. In the classical setting, it is proven secure up to $O(2^{n/2})$ online classical queries and time. In fact, the tradeoff between the number of classical queries D and the time complexity T is $DT = O(2^n)$, which balances at $T = D = O(2^{n/2})$.

In 2021, Shinagawa and Iwata [28] applied Simon's algorithm to perform a key recovery attack on the SoEM21 cipher. Since Simon's algorithm identifies the period of a periodic function, constructing a function satisfying $f(x) = f(x \oplus s)$ for all x is essential. Notably, the period s is associated with keys. The periodic function built in [28] is

$$f(x) = SoEM21(x) \oplus P_1(x) \oplus P_2(x), \tag{3}$$

and it has a period of k, since

$$\begin{aligned} f(x \oplus k) &= SoEM21(x \oplus k) \oplus P_1(x \oplus k) \oplus P_2(x \oplus k) \\ &= P_1(x \oplus k \oplus k) \oplus P_2(x \oplus k \oplus k) \oplus k \oplus P_1(x \oplus k) \oplus P_2(x \oplus k) \\ &= P_1(x) \oplus P_2(x) \oplus k \oplus P_1(x \oplus k) \oplus P_2(x \oplus k) \\ &= SoEM(x) \oplus P_1(x) \oplus P_2(x) \\ &= f(x). \end{aligned} \tag{4}$$

Thus, applying Simon's algorithm, the key can be determined with $O(n)$ quantum queries and $O(n^3)$ time.

Besides, Shinagawa and Iwata [28] defined SoEMs1 as a natural generalization of SoEM21. Given s (≥ 2) independent public permutations $P_1, P_2, \ldots, P_s \in Perm(n)$ and a n-bit key string k, the SoEMs1 cipher is written as

$$SoEMs1(x) = P_1(x \oplus k) \oplus P_2(x \oplus k) \oplus \cdots \oplus P_s(x \oplus k) \oplus k. \tag{5}$$

According to [28], Simon's algorithm can also be applicable to the key recovery attack on SoEM*s*1 by constructing the following periodic function

$$f(x) = SoEMs1(x) \oplus P_1(x) \oplus P_2(x) \oplus \cdots \oplus P_s(x). \tag{6}$$

As can be seen, the function f in Eq. (6) has period k. Thus, by running Simon's algorithm to $f(x)$, the key k can be recovered with $O(n)$ quantum queries and $O(n^3)$ time.

3 Key recovery attacks on SoEM21 under the Q1 model

This section proposes a quantum key recovery attack on SoEM21 ciphers under the Q1 model where the encryption oracle can only respond to classical queries. We assume that evaluating each primitive (e.g., a block cipher) once requires $O(1)$ time.

3.1 Quantum attacks

It is noticeable that there is a polynomial-time key recovery attack on SoEM21 under the Q2 model utilizing Simon's algorithm as shown in Sect. 2.4. A common method of transforming the Q2 attack into the Q1 attack is replacing quantum queries with classical ones. Since the domain of the function $f(x)$ in Eq. (3) is $\{0,1\}^n$, we need $O(2^n)$ classical queries, i.e., querying the whole classical codebook, to prepare $\sum_{x\in\{0,1\}^n} |x\rangle|f(x)\rangle$. This is comparable to the complexity required for brute-force attacks. In the following, we will demonstrate our method for reducing the number of classical queries by utilizing the offline Simon's algorithm. The tradeoff determined in this section is $DT^2 = \tilde{O}(2^n)$. For a given D, the required time T is $\tilde{O}(\sqrt{2^n/D})$, which is the square-root of the classical $O(2^n/D)$.

Let u be an integer in the range $0 \le u \le n$. Let k express as $k^l \| k^r$, where k^l consists of u bits and k^r consists of $n-u$ bits. To reduce the number of classical queries, we reduce the domain size of $f(x)$ from $\{0,1\}^n$ to $\{0,1\}^u$, and then perform Grover's algorithm on $i \in \{0,1\}^{n-u}$ to search k^r, while we apply Simon's algorithm to recover k^l. Figure 2 depicts the flowchart of our Q1 attack on SoEM21.

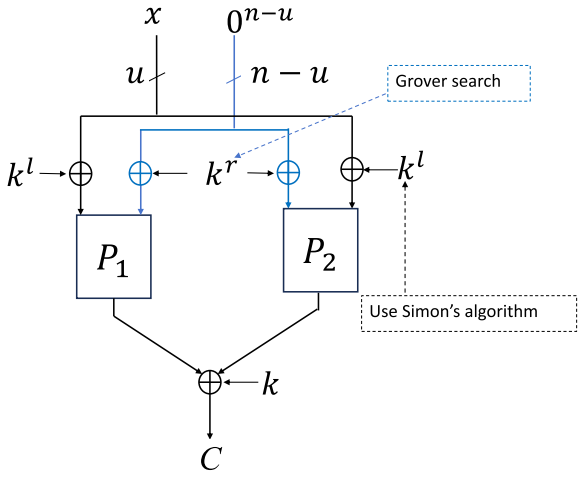


Figure 2 Flowchart of our Q1 attack on SoEM21

We consider two functions, F and G, which are constructed based on the SoEM21 encryption and permutations as follows:

$$\begin{aligned} &F : \{0,1\}^{n-u} \times \{0,1\}^u \to \{0,1\}^n, \quad F(i,x) = P_1(x\|i) \oplus P_2(x\|i), \\ &G : \{0,1\}^u \to \{0,1\}^n, \qquad\qquad G(x) = SoEM21(x\|0^{n-u}). \end{aligned} \tag{7}$$

$F(i,x) \oplus G(x)$ has the period k^l when $i = k^r$, since

$$\begin{aligned} &G(x) \oplus F(k^r, x) \\ &= P_1(x \oplus k^l \| k^r) \oplus P_2(x \oplus k^l \| k^r) \oplus k \oplus P_1(x\|k^r) \oplus P_2(x\|k^r) \\ &= P_1(x\|k^r) \oplus P_2(x\|k^r) \oplus k \oplus P_1(x \oplus k^l \| k^r) \oplus P_2(x \oplus k^l \| k^r) \\ &= G(x \oplus k^l) \oplus F(k^r, x \oplus k^l). \end{aligned} \tag{8}$$

Otherwise, if $i \neq k^r$, $F(i,x) \oplus G(x)$ is a random function. Thus, one part of k will be handled by the Grover's algorithm, while another part can be recovered by using Simon's algorithm in the Q1 model (i.e., Algorithm 5 in the Appendix).

First, we prepare the quantum state $|\psi_G\rangle = \bigotimes_{j=1}^{c} \sum_{x_j \in \{0,1\}^u} |x_j\rangle |G(x_j)\rangle$. Under known $G(p)$ via classical queries to the SoEM21 cipher for any $p \in \{0, 1, \ldots, 2^u - 1\}$, let U_p be a quantum oracle

$$|p\rangle \bigotimes_{j=1}^{c} |x_j\rangle |y\rangle \to \begin{cases} |p\rangle \bigotimes_{j=1}^{c} |x_j\rangle |y \oplus G(x_j)\rangle, \textit{if } x_1 = \cdots = x_c = p, \\ |p\rangle \bigotimes_{j=1}^{c} |x_j\rangle |y\rangle, \qquad \textit{otherwise.} \end{cases} \tag{9}$$

Thus, we can construct the following Algorithm 3 to generate $|\psi_G\rangle$.

Algorithm 3: Prepared $|\psi_G\rangle$ using classical queries

Input: Classical query access to the SoEM21 cipher, integer c.

Output: $|\psi_G\rangle$.

Initialize three registers $|0\rangle^{\otimes u} |0\rangle^{\otimes cu} |0\rangle^{\otimes cn}$.

Perform quantum gate $H^{\otimes cu}$ on the second register:

$$|\phi\rangle = \bigotimes_{j=1}^{c} \sum_{x_j \in \{0,1\}^u} |x_j\rangle |0\rangle.$$

For $p = 0, 1, \ldots, 2^u - 1$ do

  Prepare the u-qubit state $|p\rangle$ on the first register

  Compute $G(p)$ by making classical queries to the SoEM21 cipher

  Apply the quantum oracle U_p to $|p\rangle \otimes |\phi\rangle$

  Uncompute $|p\rangle$

end for

Return $|\psi_G\rangle$

Algorithm 4: Recover k^r using classical queries

Input: Oracle $O_F : |i,x,y\rangle \mapsto |i,x,y \oplus F(i,x)\rangle$ with $F : \{0,1\}^{n-u} \times \{0,1\}^u \to \{0,1\}^n$, classical query access to the SoEM21 cipher, integer c.

Output: k^r.

1 Initialize $|0\rangle^{\otimes n-u}|0\rangle^{\otimes cu}|0\rangle^{\otimes cn}|1\rangle$.

2 Apply Algorithm 3: $|\psi_G\rangle = \bigotimes_{j=1}^{c} \sum_{x_j \in \{0,1\}^u} |x_j\rangle|G(x_j)\rangle$.

3 Perform $H^{\otimes n-u} \otimes I_u^{\otimes c} \otimes I_n^{\otimes c} \otimes H$:

$$\sum_{i \in \{0,1\}^{n-u}} |i\rangle \otimes |\psi_G\rangle|b\rangle,$$

where $|b\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ and I_n represents identity operator acting on n qubits.

4 Make c oracle queries O_F:

$$\sum_{i \in \{0,1\}^{n-u}} |i\rangle \bigotimes_{j=1}^{c} \sum_{x_j \in \{0,1\}^u} |x_j\rangle|G(x_j) \oplus F(i,x_j)\rangle = \sum_{i \in \{0,1\}^{n-u}} |i\rangle \otimes |\psi_{G \oplus F}\rangle|b\rangle.$$

5 Apply quantum gate $I_{n-u} \otimes H^{\otimes cu} \otimes I_n^{\otimes c} \otimes I_1$:

$$\sum_{i \in \{0,1\}^{n-u}} |i\rangle \bigotimes_{j=1}^{c} \sum_{x_j, y_j \in \{0,1\}^u} (-1)^{x_j \cdot y_j} |y_j\rangle|G(x_j) \oplus F(i,x_j)\rangle|b\rangle.$$

6 Compute the dimension d of the vector space spanned by $y_1, y_2, \cdots, y_c$. If $d = u$, set $r = 0$; If $d < u$, set $r = 1$. Add r to b, and then uncompute d and r to obtain

$$\sum_{i \in \{0,1\}^{n-u}} |i\rangle \bigotimes_{j=1}^{c} \sum_{x_j, y_j \in \{0,1\}^u} (-1)^{x_j \cdot y_j} |y_j\rangle|G(x_j) \oplus F(i,x_j)\rangle|b \oplus r\rangle.$$

7 Reapply Hadmard gates $I_{n-u} \otimes H^{\otimes cu} \otimes I_n^{\otimes c} \otimes I_1$.

8 Make c oracle queries to O_F:

$$\sum_{i \in \{0,1\}^{n-u}} |i\rangle \otimes |\psi_G\rangle|b \oplus r\rangle.$$

9 After performing $O(2^{(n-u)/2})$ Grover iterations, measure to return k^r.

Then, using the quantum oracle $O_F : |i,x,y\rangle \mapsto |i,x,y \oplus F(i,x)\rangle$, we give the procedure finding k^r in Algorithm 4. Next, k^l can be recovered by performing Simon's algorithm in the Q1 model to $F(k^r,x) \oplus G(x)$.

3.2 Complexity analysis

In Algorithm 3, $|p\rangle$ ($p \in \{0,1,\ldots,2^u-1\}$) can be prepared using some quantum gates only, such as the X gate. The implementation of each U_p requires one classical query to $G(x)$.

Since one evaluation of $G(x)$ can be done by $O(1)$ evaluations of the SoEM21 cipher, Algorithm 3 generates $|\psi_G\rangle$ with $O(2^u)$ classical queries and $O(u + c(n + u))$ qubits (u is an integer in the range $0 \leq u \leq n$).

Algorithm 4 is divided into two phases, online and offline computations. The online phase, i.e., step 2, prepares the quantum state $|\psi_G\rangle$ by performing Algorithm 3. And $|\psi_G\rangle$ will be reused in each Grover iteration. The offline computations correspond to step 3 to step 9. In particular, steps 4-8 corresponds to the test oracle:

$$|i\rangle|\psi_G\rangle|b\rangle \rightarrow \begin{cases} |i\rangle|\psi_G\rangle|b \oplus 1\rangle, F(i,x) \oplus G(x) \textit{ is a periodic function}, \\ |i\rangle|\psi_G\rangle|b \oplus 0\rangle, \qquad \textit{otherwise}. \end{cases} \tag{10}$$

In Eq. (10), $|b\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. It is achieved without any new query to $G(x)$. Instead, $2c$ quantum queries to $F(i,x)$ and $O(n^3)$ time for implementing step 6 are required. In step 6, we choose the quantum circuit proposed by Bonnetain and Jaques [33] to compute the dimension d of the vector span, whose costs is shown in *Remark* 1. As a result, Algorithm 4 recovers k^r with $O(2^u)$ classical queries to $G(x)$ and $O((n^3 + 2c)2^{(n-u)/2})$ time. Note that one evaluation of $F(i,x)$ can be done by $O(1)$ evaluations of P_1 and P_2.

Remark 1 [33] Given c u-bit vectors as input, the dimension d of their span is computed with $(c + u)\log_2 u$ depth, at least $c + 3u^2/2 - u/2$ auxiliary qubits (excluding cn qubits used to store the input) and $cu^2 + cu$ Toffoli gates.

According to Theorem 14 of [34], when $c = n + \tau + 1$ (τ is a positive integer), the success probability of Algorithm 4 is lower bounded by $1 - 2^{-\tau} - (2^{-\tau/2-1} + 2^{-\tau} + 2^{-(n-u)/2+1})^2$. Setting $n - u \geq 32$ and $\tau = 7$ ensures the success probability exceeds 99%. By performing Simon's algorithm in the Q1 model (i.e., Algorithm 5 in the Appendix) to the function $F(k^r,x) \oplus G(x)$, we can recover k^l with $O(2^u)$ classical queries, $O(u^3 + u)$ time (requiring $O(u)$ quantum queries to $F(k^r,x)$) and $O(u^2)$ qubits. Thus, we obtain the following Theorem 2.

Theorem 2 *There exists a quantum attack that can recover the key k of* SoEM21 *with a probability at least* 99% *by performing $D = O(2^u)$ classical queries to* SoEM21 *and making $T = O(n^3 2^{(n-u)/2})$ time. The tradeoff is $DT^2 = \tilde{O}(2^n)$, which balances at $T = D = \tilde{O}(2^{n/3})$. Besides, our attack requires $O(n)$ classical memory and $O(n^2)$ qubits.*

In Theorem 2, since $T = O(n^3 2^{(n-u)/2})$ can be represented as $\tilde{O}(2^{(n-u)/2})$, the tradeoff curve between time complexity T and the number of classical queries D is given by $DT^2 = \tilde{O}(2^n)$. From this curve, we can derive the time complexity as $T = \tilde{O}\left(\sqrt{2^n/D}\right)$, which decreases as D increases. The balanced case occurs when both D and T are minimized simultaneously. Substituting $D = O(2^{n/3})$ into the tradeoff curve yields $T = \tilde{O}\left(\sqrt{\frac{2^n}{2^{n/3}}}\right) = \tilde{O}(2^{n/3})$, achieving a symmetric balance between query complexity and time complexity. Practically, this implies that a quantum adversary can break the SoEM21 cipher in $\tilde{O}(2^{n/3})$ time, with $O(2^{n/3})$ classical queries and $O(n^2)$ qubits. While current quantum devices are far from this capability, our analysis establishes a baseline for future hardware requirements.

We introduce Lemma 1 to characterize the quantum resources required for attacking SoEM21.

Lemma 1 *Given $u = n/3$ and $c = n + \tau + 1$, the quantum resources required for recovering the key k are characterized as follows*:

- Depth D:

$$D = D_{|\psi_G\rangle} + 2^{n/3}\left(2D_P + D_{test}[c, n/3]\right);$$

- Number of qubits W:

$$\begin{aligned} W &= W_{|\psi_G\rangle} + (n-u) + cu + cn + c \cdot W_P + W_{test}[c,u] + 1 \\ &= W_{|\psi_G\rangle} + \frac{2n}{3} + \frac{4nc}{3} + c \cdot W_P + W_{test}[c, n/3] + 1; \end{aligned}$$

- Number of quantum gates G:

$$G = G_{|\psi_G\rangle} + 2^{n/3}\left(2c \cdot G_P + G_{test}[c, n/3]\right).$$

Here, P denotes the permutation $P_1 \oplus P_2$ from SoEM21. *$|\psi_G\rangle$ is prepared via Algorithm 3, and test is the process computing the dimension d of the c u-bit vectors. D_f, W_f and G_f ($f = |\psi_G\rangle, test, P$) denote the depth, the number of auxiliary qubits and the number of quantum gates applied to implement f respectively. Only the costs required for Algorithm 4 are estimated. The costs of some operations, including computing k^l using Algorithm 5 and applying H gates in steps 3, 5, 7 of Algorithm 4, are ignored, because they have a minimal impact on the overall resource consumption.*

4 Quantum attacks on SoEMs1

In this section, we apply the offline Simon's algorithm to develop the quantum attack on the SoEM*s*1 cipher, using classical queries only. First, we show its classical security for the first time. We again assume that evaluating each primitive (e.g., a block cipher) once requires a time complexity of $O(1)$.

4.1 The classical attack on SoEMs1

In the following, we give a classical attack on the SoEM*s*1 cipher and obtain the following Theorem 3.

Theorem 3 *There exists a classical attack on* SoEM*s*1 *ciphers recovering an n-bit key with a probability at least $1 - 2^{-n}$, requiring $D = O(2^u)$ online queries, $T = O(2^{n-u})$ time and $O(2^u)$ classical memory. Besides, complexities are balanced at $u = n/2$.*

Proof We write $x = x^u \| x^{n-u}$, where x^u is of u bits and x^{n-u} is of $n-u$ bits. Let $\mathcal{L} = \{(x, x^*, x^{**}) | x = x^u \| 0^{n-u}, x^* = x^u \| 0^{n-u-1}1, x^{**} = x^u \| 0^{n-u-2}10, x^u \in \{0,1\}^u\}$ and $\mathcal{X} = \{(y, y^*, y^{**}) | y = 0^u \| y^{n-u}, y^* = 0^u \| y^{n-u} \oplus 0^{n-1}1, y^{**} = 0^u \| y^{n-u} \oplus 0^{n-2}10, y^{n-u} \in \{0,1\}^{n-u}\}$ be subsets of $\{0,1\}^{3n}$ respectively.

First, by making online classical queries to SoEM*s*1 ciphers, we obtain all ciphers corresponding to the input (x, x^*, x^{**}) in $\mathcal{L}$, i.e.,

$$C = \{(SoEMs1(x), SoEMs1(x^*), SoEMs1(x^{**}) | (x, x^*, x^{**}) \in \mathcal{L}\}, \tag{11}$$

and compute

$$\begin{aligned} &SoEMs1(x) \oplus SoEMs1(x^*), \\ &SoEMs1(x) \oplus SoEMs1(x^{**}). \end{aligned} \tag{12}$$

Second, by making offline queries to $P_1, \ldots, P_s$ respectively, we obtain

$$C^* = \{P_i(y), P_i(y^*), P_i(y^{**}) | (y, y^*, y^{**}) \in \mathcal{X}, i = 1, 2, \ldots, s\}. \tag{13}$$

and compute

$$\begin{aligned} &P_1(y) \oplus P_1(y^*) \oplus \cdots \oplus P_s(y) \oplus P_s(y^*), \\ &P_1(y) \oplus P_1(y^{**}) \oplus \cdots \oplus P_s(y) \oplus P_s(y^{**}). \end{aligned} \tag{14}$$

Third, we find (x, x^*, x^{**}) and (y, y^*, y^{**}) such that

$$\begin{aligned} &SoEMs1(x) \oplus SoEMs1(x^*) = P_1(y) \oplus P_1(y^*) \oplus \cdots \oplus P_s(y^*) \oplus P_s(y), \\ &SoEMs1(x) \oplus SoEMs1(x^{**}) = P_1(y) \oplus P_1(y^{**}) \oplus \cdots \oplus P_s(y^{**}) \oplus P_s(y). \end{aligned} \tag{15}$$

At last, we compute k as $x \oplus y$.

There is exactly one (x, x^*, x^{**}) and (y, y^*, y^{**}) such that $x \oplus y = k$, $x^* \oplus y^* = k$ and $x^{**} \oplus y^{**} = k$, leading to

$$\begin{aligned} &SoEMs1(x) = P_1(y) \oplus P_2(y) \oplus \cdots \oplus P_s(y) \oplus k, \\ &SoEMs1(x^*) = P_1(y^*) \oplus P_2(y^*) \oplus \cdots \oplus P_s(y^*) \oplus k, \\ &SoEMs1(x^{**}) = P_1(y^{**}) \oplus P_2(y^{**}) \oplus \cdots \oplus P_s(y^{**}) \oplus k. \end{aligned} \tag{16}$$

If (x, x^*, x^{**}) and (y, y^*, y^{**}) are correct, all identities in the Eq. (15) are satisfied with probability 1. If (x, x^*, x^{**}) and (y, y^*, y^{**}) are incorrect, these two identities in the Eq. (15) are fulfilled with probability at most 2^{-2n}. Then, the probability that there are incorrect (x, x^*, x^{**}) and (y, y^*, y^{**}) that satisfies the identities in the Eq. (15) is at most $(2^u - 1)(2^{n-u} - 1)2^{-2n} < 2^{-n}$. Thus, the classical attack above can recover the correct k with probability at least $1 - 2^{-n}$.

As a result, our classical attack on the SoEM*s*1 cipher can recover n-bit key k with a high probability by performing $D = O(2^u)$ online queries and making $T = O(2^{n-u})$ offline queries to $P_1, P_2, \ldots, P_s$. The tradeoff between D and T is $TD = O(2^n)$, which balances at $D = T = O(2^{n/2})$. □

4.2 The key recovery attack on SoEMs1 under the Q1 model

Recall that when performing the Q1 attack on SoEM21 ciphers in Sect. 3, the key k was divided into $k^l \| k^r$. The offline Simon's algorithm was employed to determine k^r, while the Simon's algorithm in the Q1 model was applied to recover k^l. For the SoEM*s*1 cipher, a similar approach is applied to recover k. The method achieves a tradeoff $DT^2 = \tilde{O}(2^n)$. For a given D, the required time T is $\tilde{O}(\sqrt{2^n/D})$, offering a quadratic speedup over the classical attack.

Let u be an integer such that $0 \leq u \leq n$. We take into account the following two functions, namely F and G, as follows:

$$\begin{aligned} &F : \{0,1\}^{n-u} \times \{0,1\}^u \to \{0,1\}^n, \quad F(i,x) = P_1(x\|i) \oplus \cdots \oplus P_s(x\|i), \\ &G : \{0,1\}^u \to \{0,1\}^n, \qquad\qquad G(x) = SoEMs1(x\|0^{n-u}). \end{aligned} \tag{17}$$

It can be observed that when $i = k^r$, $F(k^r, x) \oplus G(x)$ exhibits a period of k^l, since

$$\begin{aligned} F(k^r,x) \oplus G(x) = P_1(x \oplus k^l\|k^r) \oplus \cdots \oplus P_s(x \oplus k^l\|k^r) \oplus k \oplus P_1(x\|k^r) \oplus \\ \cdots \oplus P_s(x\|k^r) = F(k^r, x \oplus k^l) \oplus G(x \oplus k^l). \end{aligned} \tag{18}$$

Similar to the analysis of attacking the SoEM21 cipher in Sect. 3, the recovery of k^r is divided into online phase and offline computations phase. In the online phase, the quantum state $|\psi_G\rangle = \bigotimes_{j=1}^{c} \sum_{x_j \in \{0,1\}^u} |x_j\rangle|G(x_j)\rangle$ is prepared. The offline computations compute the function $F(i,x) \oplus G(x)$. For each fixed i, a test oracle is used to determine whether $F(i,x) \oplus G(x)$ has a period, and uncomputations is performed to revert $|\psi_G\rangle$. After performing $O(2^{(n-u)/2})$ Grover iterations, k^r can be obtained with a probability at least 99% (setting $c = n + \tau + 1$, $n - u \geq 32$ and $\tau = 7$). At last, k^l can be recovered by apply Simon's algorithm in the Q1 model to $F(k^r, x) \oplus G(x)$.

As a result, the following Theorem 4 can be gotten.

Theorem 4 *There exists a quantum attack that can recover the key k of* SoEM*s*1 *ciphers with a probability at least* 99% *by performing $D = O(2^u)$ classical queries and requiring $T = O(n^3 2^{(n-u)/2})$ time. The tradeoff is $DT^2 = \tilde{O}(2^n)$, which balances at $T = D = \tilde{O}(2^{n/3})$. Besides, our attack requires $O(n)$ classical memory and $O(n^2)$ qubits.*

Similarly, for a given D, our Q1 attack on SoEM*s*1 requires $\tilde{O}(\sqrt{2^n/D})$ time, whereas the classical attack requires $O(2^n/D)$ time. Thus, our Q1 attack on SoEM*s*1 also provides a quadratic time speedup over the classical attack.

We extend the computational resource analysis of Lemma 1 to SoEM*s*1. The quantum state $|\psi_G\rangle = \bigotimes_{j=1}^{c} \sum_{x_j \in \{0,1\}^u} |x_j\rangle|G(x_j)\rangle$ can be prepared via Algorithm 3 with classical queries to SoEM*s*1. For $P = P_1 \oplus P_2 \oplus \cdots \oplus P_s$, we denote its quantum implementation costs: having depth D_P, using W_P qubits and G_P quantum gates. By substituting D_P, W_P and G_P into Lemma 1, and taking $u = n/3$, we can derive the computational resources for attacking SoEM*s*1. While the scale of current quantum computers cannot realize the quantum attack, our analysis establishes a baseline for future hardware requirements.

5 Conclusion

This paper studied the quantum security of SoEM constructions in the Q1 model for the first time. We observed that the offline Simon's algorithm can improve the best classical attack on SoEM21 ciphers by reducing the time complexity from $O(2^{2n/3})$ to $O(2^{n/3})$, with the same query complexity. For SoEM*s*1, we introduced the first classical attack. We further showed that the offline Simon's algorithm can be extended to the key recovery at-

tack on SoEM*s*1 cipher within a time complexity of $O(2^{n/3})$, providing a quadratic speedup over the classical attack. Our results reveal that with a single key, the SoEM*s*1 cipher cannot strengthen the security of block ciphers even if multiple independent permutations are employed.

However, the proposed quantum attacks require $O(n^2)$ qubits for state preparation, which far exceeds the capacity of even the largest quantum devices. The exponential time complexity implies a depth of quantum operations that far surpasses the coherence time of qubits. Thus, exploring novel techniques, such as hybrid quantum-classical cryptographic methods, to reduce qubit count and time complexity required for attacking SoEM21 and SoEM*s*1 could mitigate hardware dependencies, representing an exciting future direction. Since increasing the number of independent keys may resist these quantum attacks, investigating the security of SoEM with multiple independent keys and permutations in the Q1 model is another interesting research direction. Furthermore, exploring other potential defenses against these quantum attacks is necessary.

Appendix: Simon's algorithm in the Q1 model

Under the condition where $l = O(n)$, Algorithm 5 returns the period of $f(x)$ with a probability at least $1 - 3^l/2^{2l-n}$ (setting $l \geq 3n$ ensures the success probability is at least 99%) by performing $O(2^n)$ online classical queries and doing $O(n^3)$ offline computations. Besides, the number of qubits required for Algorithm 5 is $O(n^2)$.

Algorithm 5: Simon's algorithm using classical queries [22]

Input: Classical query access to the periodic function $f(x) : \{0,1\}^n \to \{0,1\}^n$.

Output: s.

1 Perform 2^n classical query to $f(x)$ and prepare the quantum state

$$|\psi_f\rangle = \bigotimes_{j=1}^{l} \sum_{x\in\{0,1\}^n} |x\rangle|f(x)\rangle.$$

2 Perform Hadamard gates $H^{\otimes n}$ to each state $|x\rangle$:

$$\bigotimes_{j=1}^{l} \sum_{x,u\in\{0,1\}^n} (-1)^{x\cdot u}|u\rangle|f(x)\rangle.$$

3 Measure all $|u\rangle$ registers to obtain l vectors $u_1, u_2, \ldots, u_l$.

4 Compute the dimension $d = dim(Span(u_1, u_2, \ldots, u_l))$. If $d \neq n-1$, return $\perp$. If $d = n-1$, compute the vector $s \neq 0 \in \{0,1\}^n$ that is orthogonal to $V = span(u_1, u_2, \ldots, u_l)$ by making use of classical linear algebra.

Abbreviations
SoEM, Sun of Even-Mansour; MAC, Message Authentication Codes; ECC, Elliptic Curve Cryptography.

Author contributions
Zhenqiang Li and Shuqin Fan proposed the attack, conducted the complexity analysis, and wrote the main manuscript text. Fei Gao, Yonglin Hao and Hongwei Sun checked the writing. Xichao Hu and Dandan Li reviewed the manuscript. All authors read and approved the final manuscript.

Funding information
This work was supported by the National Natural Science Foundation of China (Grant Nos. 62372048, 62272056, 62471070), the Fundamental Research Funds for Heilongjiang Universities (Grant No. 2024-KYYWF-0137).

Data availability
No datasets were generated or analysed during the current study.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
The Author confirms: that the work described has not been published before; that it is not under consideration for publication elsewhere; that its publication has been approved by all co-authors.

Competing interests
The authors declare no competing interests.

Author details
[1]State Key Laboratory of Cryptology, Beijing, 100878, China. [2]State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, 100876, China. [3]School of Computer and Big Data (School of Cybersecurity), Heilongjiang University, Harbin, 150080, China. [4]School of Computer Science (National Pilot Software Engineering School), Beijing University of Posts and Telecommunications, Beijing, 100876, China.

Received: 12 April 2025 Accepted: 26 May 2025 Published online: 05 June 2025

References

1. Harrow AW, Hassidim A, Lloyd S. Quantum algorithm for linear systems of equations. Phys Rev Lett. 2009;103(15):150502.
2. Wan L-C, Yu C-H, Pan S-J, Gao F, Wen Q-Y, Qin S-J. Asymptotic quantum algorithm for the Toeplitz systems. Phys Rev A. 2018;97(6):062322.
3. Liu H-L, Wu Y-S, Wan L-C, Pan S-J, Qin S-J, Gao F, Wen Q-Y. Variational quantum algorithm for the Poisson equation. Phys Rev A. 2021;104(2):022418.
4. Lloyd S, Mohseni M, Rebentrost P. Quantum principal component analysis. Nat Phys. 2014;10(9):631–3.
5. Pan S-J, Wan L-C, Liu H-L, Wang Q-L, Qin S-J, Wen Q-Y, Gao F. Improved quantum algorithm for a-optimal projection. Phys Rev A. 2020;102(5):052402.
6. Yu C-H, Gao F, Lin S, Wang J. Quantum data compression by principal component analysis. Quantum Inf Process. 2019;18(8):249.
7. Song Y, Wu Y, Wu S, Li D, Wen Q, Qin S, Gao F. A quantum federated learning framework for classical clients. Sci China, Phys Mech Astron. 2024;67(5):250311.
8. Yu C-H, Gao F, Wen Q-Y. An improved quantum algorithm for ridge regression. IEEE Trans Knowl Data Eng. 2019;33(3):858–66.
9. Li L, Li J, Song Y, Qin S, Wen Q, Gao F. An efficient quantum proactive incremental learning algorithm. Sci China, Phys Mech Astron. 2025;68(1):1–9.
10. Liu N, Rebentrost P. Quantum machine learning for quantum anomaly detection. Phys Rev A. 2018;97(4):042315.
11. Shor PW. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Rev. 1999;41(2):303–32.
12. Grover LK. A fast quantum mechanical algorithm for database search. In: Proceedings of the twenty-eighth annual ACM symposium on theory of computing. 1996. p. 212–9.
13. Simon DR. On the power of quantum computation. SIAM J Comput. 1997;26(5):1474–83.
14. Kuwakado H, Morii M. Quantum distinguisher between the 3-round feistel cipher and the random permutation. In: 2010 IEEE international symposium on information theory. IEEE; 2010. p. 2682–5.
15. Kuwakado H, Morii M. Security on the quantum-type even-mansour cipher. In: 2012 international symposium on information theory and its applications. IEEE; 2012. p. 312–6.
16. Kaplan M, Leurent G, Leverrier A, Naya-Plasencia M. Breaking symmetric cryptosystems using quantum period finding. In: Advances in cryptology–CRYPTO 2016: 36th annual international cryptology conference, Santa Barbara, CA, USA, August 14-18, 2016. Proceedings, part II. vol. 36. Berlin: Springer; 2016. p. 207–37.
17. Sun H-W, Cai B-B, Qin S-J, Wen Q-Y, Gao F. Quantum attacks on beyond-birthday-bound macs. Phys A, Stat Mech Appl. 2023;625:129047.
18. Zhandry M. How to construct quantum random functions. J ACM. 2021;68(5):1–43.
19. Brassard G, Høyer P, Tapp A. Quantum cryptanalysis of hash and claw-free functions. In: LATIN'98: theoretical informatics: third Latin American symposium campinas, Brazil, April 20–24, 1998. Proceedings. vol. 3. Springer; 1998. p. 163–9.

20. Zhang S. Promised and distributed quantum search. In: International computing and combinatorics conference. Berlin: Springer; 2005. p. 430–9.
21. Leander G, May A. Grover meets Simon–quantumly attacking the fx-construction. In: Advances in cryptology–ASIACRYPT 2017: 23rd international conference on the theory and applications of cryptology and information security, Hong Kong, China, December 3-7, 2017. Proceedings, part II. vol. 23. Springer; 2017. p. 161–78.
22. Bonnetain X, Hosoyamada A, Naya-Plasencia M, Sasaki Y, Schrottenloher A. Quantum attacks without superposition queries: the offline Simon's algorithm. In: International conference on the theory and application of cryptology and information security. Berlin: Springer; 2019. p. 552–83.
23. Cai B, Gao F, Leander G. Quantum attacks on two-round even-mansour. Front Phys. 2022;10:1028014.
24. Anand R, Ghosh S, Isobe T, Shiba R. Quantum key recovery attacks on 4-round iterated even-mansour with two keys. In: International conference on information security. Berlin: Springer; 2024. p. 87–103.
25. Bonnetain X, Schrottenloher A, Sibleyras F. Beyond quadratic speedups in quantum attacks on symmetric schemes. In: Annual international conference on the theory and applications of cryptographic techniques. Berlin: Springer; 2022. p. 315–44.
26. Chen YL, Lambooij E, Mennink B. How to build pseudorandom functions from public random permutations. In: Advances in cryptology–CRYPTO 2019: 39th annual international cryptology conference, Santa Barbara, CA, USA, August 18–22, 2019. Proceedings, part I. vol. 39. Berlin: Springer; 2019. p. 266–93.
27. Even S, Mansour Y. A construction of a cipher from a single pseudorandom permutation. J Cryptol. 1997;10:151–61.
28. Shinagawa K, Iwata T. Quantum attacks on sum of even-mansour pseudorandom functions. Inf Process Lett. 2022;173:106172.
29. Zhang P. Quantum related-key attack based on Simon's algorithm and its applications. Symmetry. 2023;15(5):972.
30. Nielsen MA, Chuang IL. Quantum computation and quantum information. England; 2010.
31. Brassard G, Hoyer P, Mosca M, Tapp A. Quantum amplitude amplification and estimation. Contemp Math. 2002;305:53–74.
32. Kilian J, Rogaway P. How to protect des against exhaustive key search. In: Advances in cryptology—CRYPTO'96: 16th annual international cryptology conference Santa Barbara, California, USA August 18–22, 1996. Proceedings. vol. 16. Berlin: Springer; 1996. p. 252–67.
33. Bonnetain X, Jaques S. Quantum period finding against symmetric primitives in practice. IACR Cryptol. ePrint Arch. 2020;2020:1418.
34. Bonnetain X. Tight bounds for Simon's algorithm. In: Progress in cryptology–LATINCRYPT 2021: 7th international conference on cryptology and information security in Latin America, Bogotá, Colombia, October 6–8, 2021. Proceedings. vol. 7. Springer; 2021. p. 3–23.

Publisher's note