# Beyond quadratic speedups in quantum attacks on symmetric schemes

Xavier Bonnetain, André Schrottenloher, Ferdinand Sibleyras

# Beyond quadratic speedups in quantum attacks on symmetric schemes *

Xavier Bonnetain[1], André Schrottenloher[2], and Ferdinand Sibleyras[3]

[1] Université de Lorraine, CNRS, Inria, Nancy, France
[2] Cryptology Group, CWI, Amsterdam, The Netherlands
[3] NTT Social Informatics Laboratories

**Abstract.** In this paper, we report the first quantum key-recovery attack on a symmetric block cipher design, using classical queries only, with a more than quadratic time speedup compared to the best classical attack.
We study the 2XOR-Cascade construction of Gaži and Tessaro (EUROCRYPT 2012). It is a key length extension technique which provides an $n$-bit block cipher with $\frac{5n}{2}$ bits of security out of an $n$-bit block cipher with $2n$ bits of key, with a security proof in the ideal model. We show that the offline-Simon algorithm of Bonnetain et al. (ASIACRYPT 2019) can be extended to, in particular, attack this construction in quantum time $\widetilde{\mathcal{O}}(2^n)$, providing a 2.5 quantum speedup over the best classical attack.
Regarding post-quantum security of symmetric ciphers, it is commonly assumed that doubling the key sizes is a sufficient precaution. This is because Grover's quantum search algorithm, and its derivatives, can only reach a quadratic speedup at most. Our attack shows that the structure of some symmetric constructions can be exploited to overcome this limit. In particular, the 2XOR-Cascade cannot be used to generically strengthen block ciphers against quantum adversaries, as it would offer only the same security as the block cipher itself.

**Keywords:** Post-quantum cryptography, quantum cryptanalysis, key-length extension, 2XOR-Cascade, Simon's algorithm, quantum search, offline-Simon.

## 1 Introduction

In 1994, Shor [51] designed polynomial-time quantum algorithms for factoring and computing discrete logarithms, both believed to be classically intractable. This showed that a large-scale quantum computer could break public-key cryptosystems based on these problems, such as RSA and ECC, which unluckily are the most widely used to date.

The impact of quantum computers on secret-key cryptography is, at first sight, much more limited. The lack of *structure* in secret-key cryptosystems

seems to defeat most exponential quantum speedups. It can be expected to do so, as it was shown in [8] that relative to an oracle, quantum speedups for worst-case algorithms can be polynomial at most, unless the oracle satisfies some additional structure. This structure, that is essential for exponential speedups, is usually known as a "promise". For example, in Shor's abelian period-finding algorithm [51], the promise is that the oracle is a periodic function.

Another well-known quantum algorithm, Grover's quantum search [28], can speed up an exhaustive key search by a quadratic factor. That is, an attacker equipped with a quantum computer can find the $\kappa$-bit key of a strong block cipher in about $\mathcal{O}\left(2^{\kappa/2}\right)$ operations instead of the $\mathcal{O}(2^{\kappa})$ trials necessary for a classical attacker. Despite being merely polynomial, this is already an interesting advantage for this hypothetical attacker. Due to Grover's search, symmetric cryptosystems are commonly assumed to retain roughly half of their classical bits of security, and it is recommended to double their key length when aiming at post-quantum security [48].

*Superposition Attacks.* In [43], Kuwakado and Morii designed a polynomial-time *quantum distinguisher* on the three-round Luby-Rackoff construction, although it has a classical security proof. Later on, they showed a polynomial-time key-recovery attack on the Even-Mansour construction [44], a classically proven block cipher constructed from a public permutation [25].

Both of these attacks can assume ideal building blocks (random functions in the case of the Luby-Rackoff construction, a random permutation in the case of Even-Mansour), as they focus on the *algebraic structure* of the construction. The target problem (distinguishing or key-recovery) is simply reduced to the problem of finding the hidden period of a periodic function, which can be solved efficiently.

However, in order to run these attacks, the quantum adversary needs to access the construction as a *quantum oracle* (in *superposition*). This means that the black-box must be part of a quantum circuit. When it comes to provable security in the quantum setting, this is a natural assumption, followed by most of the works in this direction (see [6,53] for instance). However, it does not seem too hard to avoid quantum queries at the implementation level[4].

Many other symmetric constructions have been shown to be broken under superposition queries in the past few years [38,10,45,14]. All these attacks have been exploiting the algebraic structure of their targets in similar ways, using different period or shift-finding algorithms.

*Attacks based on Quantum Search. Quantum search*, the equivalent of classical exhaustive search, is a very versatile tool that allows to design many algorithms beyond a mere exhaustive search of the key. However, by only combining quan-

[4] Though it seems also impossible in some restricted cases, for example *white-box encryption*. Here the adversary tries to recover the key of a block cipher whose specification is completely given to him. He can realize the quantum oracle using this specification.

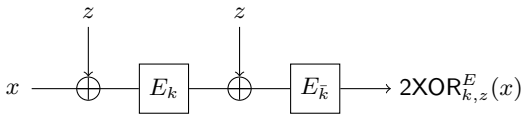


**Fig. 1.** The 2XOR construction of [27]. $E$ is an ideal $n$-bit block cipher, $z$ is an $n$-bit key, $k$ is a $\kappa$-bit key and $\bar{k}$ is $\pi(k)$ for some chosen permutation $\pi$ without fixpoints.

tum search with itself, one cannot obtain a better speedup than quadratic. More precisely[5]:

> Let $\mathcal{A}$ be a quantum algorithm, with a final measurement, that is built by combining a constant number of quantum search procedures. Let $\mathcal{T}$ be its time complexity and $\mathcal{M}$ its memory complexity. Then there exists a classical randomized algorithm $\mathcal{A}'$ that returns the same results using $\mathcal{M}$ memory and time $\mathcal{O}(\mathcal{T}^2)$.

In other words, if our only quantum algorithmic tool is quantum search, then any quantum attack admits an equivalent classical attack of squared complexity, and that uses a similar memory. In particular, if the quantum procedure goes below the exhaustive key search ($\mathcal{O}(2^{\kappa/2})$), then the corresponding classical procedure can be expected to go below the classical exhaustive key search ($\mathcal{O}(2^{\kappa})$).

*Attacks beyond Quantum Search.* So far, when superposition queries are forbidden, all known quantum attacks on symmetric designs (e.g., key-recovery attacks on block ciphers, forgery attacks on MACs) have only been confirmed to reach time speedups less than (or equal to) quadratic: the best that quantum search, and other extended frameworks [47], can offer.

At ASIACRYPT 2019, Bonnetain *et al.* [12] presented new attacks on the Even-Mansour and FX block ciphers that somehow went "beyond quantum search only". Their algorithm combines Simon's algorithm [52] and quantum search, inspired by an attack of Leander and May [45]. In some scenarios, it allows to reach a quadratic speedup *and* an asymptotic memory improvement at the same time. For example, they obtained an attack on an $n$-bit Even-Mansour cipher, with $2^{n/3}$ classical queries, in quantum time $\widetilde{\mathcal{O}}(2^{n/3})$, and memory $\mathsf{poly}(n)$, instead of a classical attack with time $\mathcal{O}(2^{2n/3})$ *and memory* $2^{n/3}$.

*Contributions of this Paper.* In this paper, we show that the offline-Simon algorithm of [12] can be extended to attack some symmetric constructions with a (provable) quantum time speedup 2.5. Our main example is the *double-XOR Cascade construction* (2XOR in what follows) of Figure 1, introduced by Gazi and Tessaro [27].

From an $n$-bit block cipher with key length $\kappa$, the 2XOR builds a block cipher with key length $n + \kappa$. It can be seen as a strengthening of the FX construction (which would have a single block cipher call) that enhances the security when

[5] For completeness, we include a short proof of this claim in Appendix A.

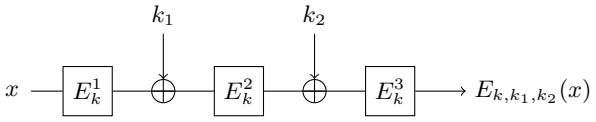


**Fig. 2.** *Doubly-extended FX* (DEFX) construction. $E^1, E^2, E^3$ are possibly independent block ciphers, but using the same $\kappa$-bit key $k$. $k_1$ and $k_2$ are independent $n$-bit whitening keys.

the adversary can make many queries. Indeed, in the ideal cipher model, any classical key-recovery of $\mathsf{2XOR}^E_{k,z}$ requires at least $\mathcal{O}\left(2^{\kappa+n/2}\right)$ evaluations of $E$, even in a regime where the adversary has access to the full codebook of $\mathsf{2XOR}^E_{k,z}$.

In the quantum setting, one can prove (see Section 5.3) that a quantum adversary needs at least $\mathcal{O}\left(2^{\kappa/2}\right)$ quantum queries to either $E$, $\mathsf{2XOR}^E_{k,z}$ or their inverses. In Section 4.2, we show the following:

> Given $2^u$ classical chosen-plaintext queries to $\mathsf{2XOR}^E_{k,z}$, a quantum attacker can retrieve the key $k, z$ in quantum time $\mathcal{O}\left(n2^u + n^3 2^{(\kappa+n-u)/2}\right)$.

In particular, when $\kappa = 2n$, a classical adversary knowing the full codebook needs a time $\mathcal{O}(2^{\frac{5n}{2}})$ to recover the key, whereas a quantum adversary requires only $\widetilde{\mathcal{O}}(2^n)$. In that case, $\mathsf{2XOR}^E_{k,z}$ offers actually *no improvement* over the $FX$ construction, in the quantum setting.

Beyond 2XOR, we use offline-Simon to attack the extended construction of Figure 2 with the same complexity. We identify other settings where a quantum adversary can gain this 2.5 advantage, e.g., a key-recovery on ECBC-MAC where part of the plaintext is unknown. We also extend our study to the case of *known plaintext queries*, where all but a fraction of the codebook is known, and show that offline-Simon still works in this setting.

This 2.5 speedup was not observed before in [12] because the authors considered constructions such as FX, which would omit the calls to $E^1_k$ and $E^3_k$. In that case, there exists improved classical time-data trade-offs that allow to reach precisely the square of the quantum time complexities, and offline-Simon only improves the memory consumption.

Whether this 2.5 speedup is the best achievable is an interesting question. We conjecture that variants of offline-Simon could reach a cubic speedup on appropriate problems, but we have not identified any corresponding cryptographic scenario.

*Organization of the Paper.* We start in Section 2 by defining most of the block cipher constructions that will be considered in this paper, and their classical security results. We include results of quantum cryptanalysis for comparison. Details of the attacks are defferred to Section 3, where we also cover some definitions and necessary background of quantum cryptanalysis, notably quantum search, Simon's algorithm and offline-Simon.

We regroup our results and applications in Section 4. We introduce a construction similar to 2XOR (EFX) and propose self-contained proofs of classical

and quantum security. Next, we detail our quantum attack in a chosen-plaintext setting. We also show that when almost all the codebook is known, known-plaintext queries can replace chosen-plaintext queries in offline-Simon. This allows us to devise an attack against EFX and a strengthened variant which we call DEFX.

We discuss the limits of these results in Section 6. We conjecture that a variant of offline-Simon could reach a cubic gap, though no corresponding cryptographic problem has been identified for now. We also discuss the apparent similarity of this 2.5 speedup with a 2.5 gap in query complexity [4].

*Notations.* Throughout this paper, we will use the following notation:

- $E$ will be an $n$-bit state, $\kappa$-bit key block cipher: a family of $2^\kappa$ efficiently computable (and invertible) permutations of $\{0,1\}^n$. Security proofs consider the *ideal model*, where $E$ is selected uniformly at random. Attacks (distinguishers, key-recoveries) are randomized algorithms whose success probability is studied on average over the random choice of $E$. We will also use $E^i$ to denote independent block ciphers.
- $\Pi$ is a permutation of $\{0,1\}^n$, also selected uniformly at random.
- $\omega$ is the matrix multiplication exponent. In practical examples, we can replace $\omega$ by 3 since the matrices considered are quite small (at most $256 \times 256$ for standard values of $n$).

## 2 Classical Constructions and Previous Results

In this section, we recapitulate the constructions considered in this paper. For each of them, we recall classical security bounds, quantum security bounds when they exist, and corresponding quantum attacks. These results are summarized in Table 1. The quantum attacks will be detailed in Section 3.

### 2.1 Context

We will use, for its simplicity, the Q1 / Q2 terminology of [39,31,12], which is the most common in quantum cryptanalysis works. Alternative names exist, such as "quantum chosen-plaintext attack" (qCPA) instead of Q2, found in most provable security works (e.g., [6]) and [36,18].

- A "Q2" attacker has access to a black-box quantum oracle holding some secret. We let $O_f$ denote a quantum oracle for $f$ (we will use the "standard" oracle representation, defined in Section 3).
- A "Q1" attacker can only query a black-box *classically*. Naturally, Q2 attackers are stronger than Q1, since one can always emulate a classical oracle with a quantum one (it suffices to prepare the queries in computational basis states). The Q1 setting also encompasses any situation where there is no secret, for example preimage search in hash functions.

**Table 1.** Summary of classical and quantum attacks considered in this paper. $D$ is the amount of classical queries to the construction. CPA = classical chosen-plaintext with classical computations. Q1 = classical chosen-plaintext with quantum computations (non adaptive). Q2 = quantum queries. KPA = classical known-plaintext. In quantum attacks, classical bits and qubits of memory are counted together for simplicity. We stress that all the quantum attacks considered here have only polynomial memory requirements. Complexities are displayed up to a constant. We do not consider attacks with preprocessing, or multi-user attacks. We assume $\kappa \geq n$.

| Target | Setting | Queries | Time | Mem. | Ref. |
|---|---|---|---|---|---|
| EM | Adaptive CPA | $2^{n/2}$ | $2^{n/2}$ | negl. | [24] |
| | KPA | $D \leq 2^{n/2}$ | $2^n/D$ | $D$ | [24] |
| | Q2 | $n$ | $n^\omega$ | $n^2$ | [44] |
| | Q1 | $D \leq 2^{n/3}$ | $\sqrt{2^n/D}$ | $n^2$ | [12] |
| FX | KPA | $D \leq 2^n$ | $2^{\kappa+n}/D$ | $D$ | [24] |
| | Adaptive CPA | $D \leq 2^{n/2}$ | $2^{\kappa+n}/D$ | negl. | [20] |
| | Adaptive CPA | $D \geq 2^{n/2}$ | $2^{\kappa+n}/D$ | $D^2 2^{-n}$ | [20] |
| | Q2 | $n$ | $n^\omega 2^{\kappa/2}$ | $n^2$ | [12] |
| | Q1 | $D \leq 2^n$ | $\max(D, \sqrt{2^{\kappa+n}/D})$ | $n^2$ | [12] |
| 2XOR | KPA | $D \leq 2^{n/2}$ | $2^{\kappa+n}/D$ | $D$ | [27] (adapted) |
| | Q2 | $n$ | $n^\omega 2^{\kappa/2}$ | $n^2$ | Section 4 |
| | Q1 | $D \leq 2^n$ | $\max(D, \sqrt{2^{\kappa+n}/D})$ | $n^2$ | Section 4 |

The constructions studied in this paper are block ciphers, studied in the *ideal* (cipher or permutation) model. In particular, if $F = F_k[E]$ is the construction and $E$ is its internal component, we assume that $E$ is drawn uniformly at random, and let an attacker query $F$ and $E$ separately. The security proofs show lower bounds on the number of queries to $F$ and $E$ that an attacker must make to succeed. Such bounds can be proven for classical and quantum attackers alike. A Q2 attacker will have access to both $F$ and $E$ in superposition. Though a Q1 attacker will have only classical access to $F$, *he still has quantum access to* $E$. Indeed, although supposedly chosen at random, $E$ remains a public component, with a public implementation. Thus, in the ideal model, Q1 attackers still make black-box quantum queries to $E$.

*Attack Scenarios.* Usually, an idealized cipher construction is proven to be a *strong pseudorandom permutation* (sPRP, see Definition 1 in Appendix 5). In this security notion, an adversary is asked to distinguish the construction $F_k[E]$ for a random $k$, from a random permutation, by making either forward or backward queries.

Obviously, a key-recovery attack is also a valid sPRP distinguisher. For all the constructions recalled in Table 1, the security is proven with the sPRP game, and the attacks are key-recovery attacks.

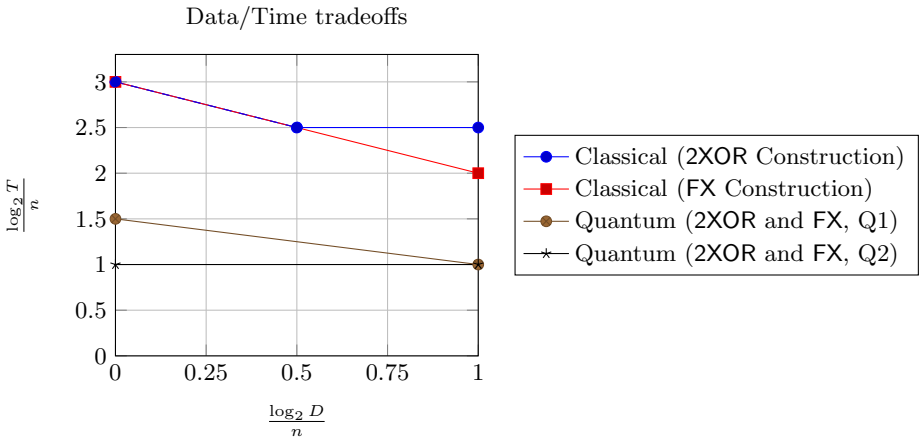


**Fig. 3.** Detail of Table 1: comparison of the FX and 2XOR security in function of the number of queries for $\kappa = 2n$.

### 2.2 The Even-Mansour Cipher

The Even-Mansour cipher [25] is a minimalistic construction which is ubiquitous in idealized designs. It starts from a public $n$-bit permutation $\Pi : \{0,1\}^n \to \{0,1\}^n$ and two $n$-bit keys $k_1, k_2$ ($k_1 = k_2$ would be enough). The cipher is defined as: $\mathsf{EM}_{k_1,k_2}(x) = \Pi(x \oplus k_1) \oplus k_2$. If $\Pi$ is a random permutation, then an adversary making $T$ queries to $\Pi$ and $D$ queries to EM cannot recover the key with success probability more than $\mathcal{O}(TD/2^n)$. Matching attacks are known [19,24]. The quantum security was first studied by Kuwakado and Morii [44], who gave a $\mathcal{O}(n^\omega)$ Q2 attack using $\mathcal{O}(n)$ queries (the attack will be presented later on). Several Q1 attacks were given in [44,30,12]. Only the latter (the most efficient) is displayed in Table 1.

### 2.3 Key-length Extension Techniques

Different ways of extending the key lengths of block ciphers have been proposed in the literature. Two well-known examples are the *FX construction* and the *Cascade* construction (or multiple-encryption).

*FX-Construction.* In [40], Kilian and Rogaway proposed key whitenings as a solution to increase the effective key length of a block cipher $E$:

$$\mathsf{FX}_{k_1,k_2,k}(x) = E_k(x \oplus k_1) \oplus k_2 \ .$$

They showed that in the ideal model, an adversary making $D$ queries to FX needs to make $T = 2^{n+\kappa}/D$ to $E$ to recover the key. This is matched by the attacks of [24,20].

The FX construction can also be seen as an Even-Mansour cipher where the public permutation $\Pi$ is replaced by an $n$-bit block cipher of unknown $\kappa$-bit key. This is why the attack strategies are similar.

*Quantum Security of FX.* In [37], it was shown that given $D$ *non-adaptive* classical chosen-plaintext queries, a quantum adversary needs at least $\sqrt{2^{n+\kappa}/D}$ queries to $E$ to recover the key of FX. This bound is matched by an attack of [12], which is also non-adaptive. It seems likely that the same bound holds for adaptive queries, although this has not been formally proven.

*Randomized Cascades.* The *double-XOR Cascade construction* (2XOR) was proposed in [27]:

$$\mathsf{2XOR}^E_{k,z}(m) = E_{\bar{k}}(E_k(m \oplus z) \oplus z)$$

where $\bar{k}$ is $\pi(k)$ for some known fixpoint-free permutation $\pi$, $k$ is a $\kappa$-bit key and $z$ is an $n$-bit key.

They show that if $E$ is an ideal cipher (drawn uniformly at random) and $k, z$ are chosen uniformly at random, then the sPRP advantage of an adversary making $q$ queries to $E$ is bounded by: $4\left(\frac{q}{2^{\kappa+n/2}}\right)^{2/3}$ (Theorem 3 in [27]). In particular, the adversary is free to query the whole codebook of $\mathsf{2XOR}^E_{k,z}$.

*3XOR and 3XSK.* Adding a third whitening key in the output of 2XOR yields the 3XOR construction of [26], which has an improved security. The authors also propose a construction without rekeying, where the two block ciphers are the same:

$$\mathsf{3XSK}_{k,z}[E](x) = E_k(E_k(x \oplus z) \oplus \pi(z)) \oplus z$$

where $\pi$ is a permutation such that $z \mapsto z \oplus \pi(z)$ is also a permutation. As far as we know, the addition of the third whitening key actually renders the offline-Simon attack inoperable.

## 3 Quantum Preliminaries

In this section, we recall some background of quantum cryptanalysis, going from Simon's algorithm to the offline-Simon algorithm from [12]. We assume that the reader is familiar with the basics of quantum computing [49] such as: the definitions of qubits, gates (Hadamard, Toffoli), quantum states and the ket notation $|\psi\rangle$. Note that we write quantum states without their global amplitude factors, e.g., $\frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}|x\rangle$ will be written $\sum_x |x\rangle$.

We will consider algorithms making oracle calls. A *quantum* (or *superposition*) oracle for a function $f$ will be represented as a black box unitary operator $O_f$: $O_f|x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle$.

Any classical reversible algorithm $\mathcal{A}$ can be written as a circuit using only Toffoli gates. Then, there exists a quantum circuit $\mathcal{A}'$ that uses the same amount of gates, but instead of computing $\mathcal{A}(x)$ on an input $x$, it computes $\mathcal{A}$ in superposition: $\mathcal{A}'|x\rangle = \mathcal{A}(x)$. We call $\mathcal{A}'$ a *quantum embedding* of $\mathcal{A}$. Classical algorithms are rarely written with reversibility in mind, but they can always be made reversible up to some trade-off between memory and time complexity overhead [9,46,42].

### 3.1 Quantum Search

It is well known that Grover's algorithm [28] provides a quadratic speedup on any classical algorithm that can be reframed as a black-box search problem. Amplitude Amplification [16] further allows to speed up the search for a "good" output in any probabilistic algorithm, including another quantum algorithm.

Let $\mathcal{A}$ be a classical probabilistic algorithm with no input, and whose output has a probability $p$ to be "good"; let $f$ a boolean function that effectively tests if the output is good. We are searching for a good output.

Classical exhaustive search consists in running $\mathcal{A}$ until the output is good, and we will do that $\mathcal{O}\left(\frac{1}{p}\right)$ times. Quantum search is a *stateful* procedure using $\mathcal{O}\left(\frac{1}{\sqrt{p}}\right)$ iterations of a quantum circuit that contains: a quantum implementation of $\mathcal{A}$, and a quantum implementation of $f$. In the case of Grover's algorithm, the search space is trivial, e.g., $\{0,1\}^n$. Here $\mathcal{A}$ has only to sample an $n$-bit string at random; the corresponding quantum algorithm is a Hadamard transform $H^{\otimes n}|0\rangle = \sum_{x\in\{0,1\}^n} |x\rangle$.

**Theorem 1 (From [16]).** *Assume that there exists a quantum circuit for $\mathcal{A}$ using $T_A$ operations, and a quantum circuit for $f$ using $T_f$ operations. Then there exists a circuit $\mathsf{QSearch}(\mathcal{A}, f)$ that, with no input, produces a good output of $\mathcal{A}$. It runs in time: $\left\lfloor \frac{\pi}{4} \frac{1}{\arcsin\sqrt{p}} \right\rfloor (2T_A + T_f)$ and succeeds with probability $\max(p, 1-p)$.*

### 3.2 Simon's Algorithm

In [52], Simon gave the first example of an exponential quantum time speedup relative to an oracle.

*Problem 1 (Boolean period-finding).* Given access to an oracle $f : \{0,1\}^n \to \{0,1\}^m$ and the promise that:

- (Periodic case) $\exists s \neq 0, \forall x, \forall y \neq x, [f(x) = f(y) \Leftrightarrow y = x \oplus s]$; or:
- (Injective case) $f$ is injective (i.e., $s = 0$).

Find $s$.

Simon showed that when $f$ is a black-box classical oracle, this problem requires $\Omega(2^{n/2})$ queries, after which a classical adversary will find a collision of $f$, i.e., a pair $x, y$ such that $f(x) = f(y)$. He can then set $s = x \oplus y$ and verify his guess with a few more queries.

However, given access to a quantum oracle $O_f$, a very simple algorithm solves this problem in $\mathcal{O}(n)$ quantum queries and $\mathcal{O}(n^\omega)$ classical postprocessing, where $\omega$ is the matrix multiplication exponent. This algorithm consists in repeating $\mathcal{O}(n)$ times a subroutine (Algorithm 1) which: • samples a random $n$-bit value $y$ in the injective case; • samples a random $n$-bit value $y$ such that $y \cdot s = 0$ in the periodic case. After $\mathcal{O}(n)$ samples, we can solve a linear system to find the case and recover $s$.

**Algorithm 1** Simon's subroutine.

| | |
|---|---|
| 1: Start in the state | $\triangleright \ \lvert 0_n\rangle \lvert 0_m\rangle$ |
| 2: Apply a Hadamard transform | $\triangleright \ \sum_x \lvert x\rangle \lvert 0_m\rangle$ |
| 3: Query $f$ | $\triangleright \ \sum_x \lvert x\rangle \lvert f(x)\rangle$ |
| 4: Measure the output register | |
| 5: Apply another Hadamard transform | |
| 6: Measure the input register, return the value $y$ obtained | |

In the injective case, Step 4 gives us a value $f(x_0)$ and makes the state collapse on $|x_0\rangle$ for some unknown $x_0$. The next Hadamard transform turns this into: $\sum_y (-1)^{x_0 \cdot y} |y\rangle$, and so, all $y$ are measured with the same probability

In the periodic case, the state collapses to a superposition of the two preimages $x_0$ and $x_0 \oplus s$: $\frac{1}{\sqrt{2}} (|x_0\rangle + |x_0 \oplus s\rangle)$. The next Hadamard transform turns this into:

$$\sum_y \left( (-1)^{x_0 \cdot y} + (-1)^{(x_0 \oplus s) \cdot y} \right) |y\rangle \quad ,$$

and thus, the amplitudes of some of the $y$ turn to zero. These $y$ *cannot* be measured. They are such that: $(-1)^{x_0 \cdot y} + (-1)^{(x_0 \oplus s) \cdot y} = 0 \implies s \cdot y = 1$, which means that we only measure random orthogonal vectors (besides, they all have the same amplitude).

*Simon's Algorithm in Cryptanalysis.* A typical example is the polynomial-time key-recovery on Even-Mansour of Kuwakado and Morii [44]. Given access to an Even-Mansour cipher $\mathsf{EM}_{k_1,k_2}$ of unknown key, define $f(x) = \mathsf{EM}_{k_1,k_2}(x) \oplus \Pi(x)$. It is periodic of period $k_1$. $\Pi$ is public, thus quantum-accessible. Given quantum oracle access to $\mathsf{EM}$, we can recover $k_1$.

Here, as most of the time in crypanalysis, the function $f$ cannot be promised to be *exactly* injective or periodic, and additional collisions will occur. Still, in our case, the output size of the periodic function is too large for these collisions to have any influence on the query cost [11].

The same principle is used in most of the known quantum polynomial-time attacks in symmetric cryptography [43,44,38,10,14,45]. A cryptanalysis problem, such as the recovery of the key or of an internal value, is encoded as a period-recovery problem.

### 3.3 Grover-meet-Simon

In [45], Leander and May proposed to combine Simon's algorithm with quantum search to attack the FX construction:

$$\mathsf{FX}_{k,k_1,k_2}(x) = E_k(x \oplus k_1) \oplus k_2 \quad .$$

Indeed, if we guess correctly the internal key $k$, then we can break the resulting Even-Mansour cipher. In fact, one can actually *recognize the good* $k$ by running an Even-Mansour attack: it will be successfull only with the correct $k$.

More generally, the *Grover-meet-Simon* algorithm solves the following problem.

*Problem 2.* Given access to a function $F(x,y) : \{0,1\}^n \times \{0,1\}^\kappa \to \{0,1\}^n$ such that there exists a unique $y_0$ such that $F(\cdot, y_0)$ is periodic, find $y_0$ and the corresponding period.

The algorithm is a quantum search over the value $y \in \{0,1\}^\kappa$. In order to guess a key $y$, it runs Simon's algorithm internally on the function $F(\cdot, y)$. It ends after $\mathcal{O}\left(n2^{\kappa/2}\right)$ quantum queries to $F$ and $\mathcal{O}\left(n^\omega 2^{\kappa/2}\right)$ quantum time.

Having no interfering periods for *all the functions* of the family $F(\cdot, y)$ allows to obtain an overwhelming probability of success for each test, and ensures the correctness of the algorithm. Again, this condition is satisfied for objects of cryptographic interest, and a tighter analysis is given in [11]. In the case of FX, we define $F(x,y) = \mathsf{FX}_{k_1,k_2,k}(x) \oplus E_y(x)$.

*Reversible Simon's Algorithm.* Let us focus on the test used inside the FX attack: it is a quantum circuit that, on input $|y\rangle|0\rangle$, returns $|y\rangle|b\rangle$ where $b = 1$ iff $x \mapsto F(y,x) = \mathsf{FX}_{k_1,k_2,k}(x) \oplus E_y(x)$ is periodic.

This quantum circuit first makes $c = \mathcal{O}(n)$ oracle queries to $F(y,x)$, building the state:

$$\bigotimes_{1\le i\le c} \sum_x |x\rangle |F(y,x)\rangle = \bigotimes_{1\le i\le c} \sum_x |x\rangle |\mathsf{FX}_{k_1,k_2,k}(x) \oplus E_y(x)\rangle \quad . \tag{1}$$

These $c$ queries are all uniform superpositions over $x$, and require to query FX. From this state, Simon's algorithm is run reversibly, without measurements. After a Hadamard transform, the input registers contain a family of $\mathcal{O}(n)$ vectors, whose dimension is computed. If the dimension is smaller than $n$, then the function is likely to be periodic.

We say "likely" because there is some probability to fail. These failures do not disrupt the algorithm, as shown in [45,12,11].

These computations can be reverted and the state of Equation 1 is obtained again. It can now be reverted to $|0\rangle$ by doing the same oracle queries to $F(y,x)$.

### 3.4 Offline-Simon

The offline-Simon algorithm of [12] can be seen as an optimization of Grover-meet-Simon, where all queries to $\mathsf{FX}_{k_1,k_2,k}$ are removed from the algorithm, except for the very first ones.

Crucially, the FX queries remain independent of the internal key guess $y$, and they are always made on the same uniform superposition $\sum_x |x\rangle$. Thus, we can consider that the following state:

$$|\psi\rangle = \bigotimes_{1\le i\le c} \sum_x |x\rangle |\mathsf{FX}_{k_1,k_2,k}(x)\rangle \quad ,$$

is given to the test circuit and returned afterwards. Intuitively, the state $|\psi\rangle$ stores all the data on FX that is required to run the attack, in a very compact way, since it fits in $\mathcal{O}\left(n^2\right)$ qubits.

With the queries done once beforehand and reused through the algorithm, the analysis is slightly different, but $\mathcal{O}(n)$ queries are still sufficient to succeed [12,11].

*Requirements.* Not all Grover-meet-Simon instances can be made "offline". For this, we need the function $F(x, y)$ to have a special form, such as $F(x, y) = f(x) \oplus g(x, y)$ where $f$ (FX in our case) is be the *offline* function, and $g$ ($E$ in our case) the *online* one. In that case, to find the single $y_0$ for which $F(\cdot, y_0)$ is periodic, it suffices to make $\mathcal{O}(n)$ queries to $f$ at the beginning of the algorithm.

*Offline-Simon and Q1 Attacks.* As Offline-Simon uses only a polynomial number of queries, such queries can become very costly without significantly increasing the time cost of the algorithm. In particular, we can now replace the quantum queries by *classical queries* and obtain interesting time-data trade-offs. We will keep the example of FX, taken from [12], with a $\kappa$-bit internal key and a block size of $n$ bits. We assume that the adversary can make $D \leq 2^n$ chosen-plaintext queries to FX.

With the offline-Simon algorithm, we proceed as follows. We let $D = 2^u$ for some $u$, and $k_1 = k_1^l \| k_1^r$, where $k_1^l$ is a subkey of $u$ bits. We define a function with a "reduced codebook":

$$\begin{cases} G \ : \{0,1\}^u \times \{0,1\}^{n-u} \times \{0,1\}^n \to \{0,1\}^n \\ x, y_1, y_2 \mapsto \mathsf{FX}_{k_1,k_2,k}(x \| 0_{n-u}) \oplus E_{y_2}(x \| y_1) \end{cases}$$

The key observation is that $G(\cdot, y_1, y_2)$ is periodic if and only if $y_1, y_2 = k_1^r, k$. In other words, part of the key will be handled by the quantum search, and part of it by the Simon subroutine.

We query $\mathsf{FX}_{k_1,k_2,k}(x \| 0_{n-u})$ for all $x$. We use this data to produce "manually" the query states. This requires $\widetilde{\mathcal{O}}(2^u)$ operations, but *in fine*, no Q2 queries at all. Next, the offline-Simon algorithm searches for the right value of $k_1^r, k$. This requires $\mathcal{O}\left(2^{(n+\kappa-u)/2}\right)$ iterations and $\mathcal{O}\left(n^\omega 2^{(n+\kappa-u)/2}\right)$ total time.

We end up with a time-data trade-off $D \cdot T^2 = \widetilde{\mathcal{O}}(2^{n+\kappa})$, valid for $D \leq 2^n$. This means that for a given $D$, we get a time $T = \widetilde{\mathcal{O}}\left(\sqrt{\frac{2^{n+\kappa}}{D}}\right)$, the square-root of the classical $T = \mathcal{O}(2^{n+\kappa}/D)$. However, while the classical attacks need $D$ memory, the quantum attack uses only $\mathcal{O}\left(n^2\right)$ qubits to store the database. This shows that Simon's algorithm is a crucial tool for this attack.

## 4 New Result and Applications

In this section, we show the 2.5 gap between a classical security proof (in the ideal model) and a quantum attack. Our target is a slightly more general construction than 2XOR, that we denote by EFX, for "extended FX".

### 4.1 The EFX Construction and its Security

Given two independent $n$-bit block ciphers $E^1, E^2$, of key size $\kappa$, and two $n$-bit whitening keys $k_1, k_2$, $\mathsf{EFX}_{k,k_1,k_2}[E^1, E^2]$ (or $\mathsf{EFX}_{k,k_1,k_2}$ for short) is an $n$-bit block cipher with $2n + \kappa$ bits of key (Figure 4):

$$\mathsf{EFX}_{k,k_1,k_2}(x) = E_k^2\left(k_2 \oplus E_k^1(k_1 \oplus x)\right) \quad .$$

The 2XOR construction is a special case of EFX in which $E^1$ and $E^2$ are the same block cipher $E$ under different keys $k, k' = \pi(k)$.

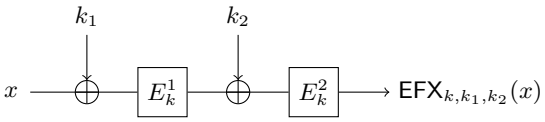


**Fig. 4.** The "extended FX" construction EFX.

*Classical Attack on EFX.* The best attack on EFX runs in time $\mathcal{O}\big(2^{\kappa+n/2}\big)$: one guesses the key $k$, then attacks the Even-Mansour cipher in time $2^{n/2}$. In fact, this is the same classical attack as for the FX construction with a slight change: after guessing the key, one has to perform reverse queries of the additional block cipher on the known ciphertext values.

Just like the attack on FX, only $2^{n/2}$ known-plaintext queries are required for this (using the slidex attack on Even-Mansour [24]). However, having access to the whole codebook of EFX does not seem to bring any improvement on the key-recovery since we'll still have to make matching queries to the additional block cipher.

More generally, let $D$ and $T$ be the number of online and offline queries respectively, the best attack runs in $DT = \mathcal{O}(2^{\kappa+n})$ for $D \leq 2^{n/2}$ or else $T = \mathcal{O}\big(2^{\kappa+n/2}\big)$ for $D \geq 2^{n/2}$.

*Classical Proof of Security.* The classical attack that we sketched above is essentially the best possible in the ideal cipher model. This can be deduced by the combination of the classical FX security bound [41] and the one derived by Gaži and Tessaro [27]. In Section 5 we also give a new proof of Theorem 2 that derives both of these bounds in a single go.

**Theorem 2.** *Consider the* EFX *construction (Figure 4) and its* sPRP *game with $n$-bit state size and $\kappa$-bit ideal blockcipher key. An adversary $\mathcal{A}$ making $D$ online queries and $T$ offline queries has an advantage bounded by both:*

$$\mathbf{Adv}^{\mathsf{sprp}}_{\cdot}(\mathcal{A}) \leq \frac{3}{2} \cdot \frac{TD}{2^{\kappa+n}} + \left(\frac{T^2 D}{2^{2\kappa+2n}}\right)^{\frac{1}{3}} + \left(\frac{2^{4n} T^2 D}{2^{2(\kappa+1)}(2^n - D + 1)^3 (2^n - (T/D \cdot 2^{2n}/2^{\kappa})^{1/3} - D + 1)^3}\right)^{\frac{1}{3}}$$

*and:*

$$\mathbf{Adv}^{\mathsf{sprp}}_{\cdot}(\mathcal{A}) \leq \frac{3}{2} \cdot \frac{T}{2^{\kappa+n/2}}$$

**Corollary 1.** *Consider the* EFX *construction (Figure 4) and its* sPRP *game. To obtain an $\Omega(1)$ advantage, it is required to have both $DT = \Omega(2^{\kappa+n})$ and $T = \Omega(2^{\kappa+n/2})$.*

*Quantum Proof of Security.* In Section 5.3, we study analogously the security in the *quantum ideal cipher model*. We show that any quantum algorithm must make at least $\mathcal{O}\left(2^{\kappa/2}\right)$ queries to EFX and its block ciphers to distinguish EFX from a random permutation, with constant probability of success. Our attack matches the bound (up to a polynomial factor).

### 4.2 Quantum Attacks

We can now explain how to attack EFX in the quantum setting.

**Theorem 3.** *There exists a quantum attack that, given $2^u$ classical chosen-plaintext queries to* EFX*, finds the complete key $k, k_1, k_2$ of the cipher in quantum time $\mathcal{O}\left(n2^u + n^\omega 2^{(\kappa+n-u)/2}\right)$. It succeeds with overwhelming probability when $E^1, E^2$ are chosen u.a.r.*

*Proof.* The attack is very similar to the offline-Simon attack on FX given in Section 3.4. We write $k_1 = k_1^l \| k_1^r$ where $k_1^l$ is of $u$ bits and $k_1^r$ is of $n-u$ bits. We query the cipher on inputs of the form $x = *\|0_{n-u}$, which take all $u$-bit prefixes, and are zero otherwise. We then use a quantum search over the complete key $k$ ($\kappa$ bits) and $k_1^r$.

The only difference with the FX attack is in the way we test a guess $y_1, y_2$ of $k_1^r, k$. The database of queries now contains:

$$\bigotimes_i \sum_{x\in\{0,1\}^u} |x\rangle\,|\mathsf{EFX}(x\|0_{n-u})\rangle = \bigotimes_i \sum_{x\in\{0,1\}^u} |x\rangle\,|E_k^2(k_2 \oplus E_k^1(k_1^l \oplus x\|k_1^r)\rangle \quad .$$

This means that given our guess $y_1, y_2$, we cannot just XOR the value of $E_{y_2}^1(x\|y_1)$ in place as we did before, because of the call to $E_k^2$.

Fortunately, since we have guessed $y_2$ (that is, the key $k$), we can map *in place*:

$$\sum_{x\in\{0,1\}^u} |x\rangle\,|E_k^2(k_2 \oplus E_k^1(k_1^l \oplus x\|k_1^r)\rangle$$
$$\mapsto \sum_{x\in\{0,1\}^u} |x\rangle\,|(E_{y_2}^2)^{-1}\left(E_k^2(k_2 \oplus E_k^1(k_1^l \oplus x\|k_1^r)\right)\rangle \quad ,$$

which, when $y_2 = k$, is exactly:

$$\sum_{x\in\{0,1\}^u} |x\rangle\,|k_2 \oplus E_k^1(k_1^l \oplus x\|k_1^r)\rangle \quad .$$

From there, we can XOR $E_{y_2}^1(x\|y_1)$ into the register and see if the function obtained is periodic. Both operations (the XOR and the permutation) are reversed afterwards, and we can move on to the next iteration.

While the periodic function can have additional collisions, its output size ($n$ bits) is actually larger than its input size ($u$ bits). Thus, with overwhelming probability, these collisions have no influence on the algorithm [11]. □

In particular, when $\kappa = 2n$ and using $2^{n-1}$ classical queries, the attack would run in time $\mathcal{O}(n^\omega 2^n)$, compared to the classical $\mathcal{O}\left(2^{5n/2}\right)$.

*Remark 1.* For a given $y_2$, $E_{y_2}$ is a permutation of known specification, of which we can compute the inverse. Thus the mapping $|z\rangle \mapsto |E_{y_2}(z)\rangle$ can be done in two steps using an ancillary register:

$$|z\rangle|0\rangle \mapsto |z\rangle|E_{y_2}(z)\rangle \mapsto |z \oplus E_{y_2}^{-1}(E_{y_2}(z))\rangle|E_{y_2}(z)\rangle = |0\rangle|E_{y_2}(z)\rangle \quad .$$

For more details on the implementation of such functions, see [11].

*Remark 2.* If the second block cipher call is done at the beginning, and not at the end, the same attack can be done with chosen-ciphertext queries.

Let us note that within this attack, we are actually using offline-Simon to solve the following problem.

*Problem 3.* Given access to a function $f : \{0,1\}^n \to \{0,1\}^n$ and a family of permutations $g_y : \{0,1\}^n \to \{0,1\}^n$, indexed by $y \in \{0,1\}^\kappa$, such that there exists a single $y_0 \in \{0,1\}^\kappa$ such that $g_{y_0}(f)$ is periodic, find $y_0$.

In the FX attack, $g_y$ was the permutation: $x \mapsto g_y(x) = x \oplus E_y(x)$. Here we simply apply in place another block cipher call, before XORing.

### 4.3 Attack with Known-Plaintext Queries

The presentation of offline-Simon in [12,11,13], which we followed in the previous section, constructs an *exact* starting database, that is, a superposition of tuples $(x, f(x))$ with all $x$es forming an affine space. Note that to construct such a vector space, there are some constraints on the queries. There are three scenarios to efficiently achieve this:

- The full codebook is queried,
- The queries are chosen,
- The queries are predictible and regular (for example, queries with a nonce incremented each time).

Hence, if we only have access to random known queries, we need to get the full codebook for our attack, which is a drastic limitation. In this section, we show that the algorithm still works if some values *are missing*. That is, instead of:

$$|\psi\rangle = \bigotimes_{i=0}^{c} \sum_{x \in \{0,1\}^n} |x\rangle|f(x)\rangle\,,$$

$$\text{we start from } |\psi'\rangle = \bigotimes_{i=0}^{c} \sum_{x \in X} |x\rangle|f(x)\rangle + \sum_{x \notin X} |x\rangle|0\rangle \quad ,$$

where $X \subsetneq \{0,1\}^n$ is the set of queries that we were allowed to make. In other words, we replace the missing output by the value 0.

Intuitively, if $X$ is close to $\{0,1\}^n$, the algorithm should not see that. It is actually easy to show by treating offline-Simon as a black-box.

**Lemma 1.** *Consider an instance of* offline-Simon *with a starting database of* $c = \mathcal{O}(n)$ *states, that succeeds with probability* $p$*. Suppose that we now start from a database where a proportion* $\alpha$ *of queries is missing (that is,* $|X| = (1-\alpha)2^n$*). Then* offline-Simon *still succeeds with probability at least* $p(1-\sqrt{2c\alpha})^2$*.*

*Proof.* We can bound the distance between $|\psi\rangle$ and the $|\psi'\rangle$ defined above. Both are sums of $2^{nc}$ basis vectors with uniform amplitudes. There are less than $c\alpha 2^{nc}$ such vectors that appear in $|\psi\rangle$ and that do not appear in $|\psi'\rangle$, and vice-versa, as the value of $f(x)$ is incorrect in each $c$ states in $|\psi'\rangle$ for at most $\alpha 2^n$ values. Thus:

$$\| \, |\psi\rangle - |\psi'\rangle \, \|^2 \leq 2c\alpha \implies \| \, |\psi\rangle - |\psi'\rangle \, \| \leq \sqrt{2c\alpha} \ .$$

Let $|\phi\rangle$ and $|\phi'\rangle$ be the states obtained after running offline-Simon with respectively $|\psi\rangle$ and $|\psi'\rangle$. We know that if we measure $|\phi\rangle$, we succeed with probability $p$. However, we are actually measuring $|\phi'\rangle$. We let $|\phi_e\rangle = |\phi'\rangle - |\phi\rangle$ the (non-normalized) error vector. We bound:

$$\langle\phi|\phi_e\rangle \leq \| \, |\phi\rangle \, \| \| \, |\phi'\rangle - |\phi\rangle \, \| = \| \, |\psi'\rangle - |\psi\rangle \, \| \leq \sqrt{2c\alpha} \ ,$$

using the fact that a unitary operator (such as offline-Simon) preserves the euclidean distance. When measuring $|\phi'\rangle$, we project onto $|\phi\rangle$ with probability:

$$(1 - \langle\phi|\phi_e\rangle)^2 \geq (1 - \sqrt{2c\alpha})^2 \ ,$$

and in that case we succeed with probability $p$.

*Remark 3.* If $\alpha = \mathcal{O}(1/n)$, then offline-Simon succeeds with constant probability.

Note that Lemma 1 only matters when we cannot choose the missing queries, i.e., in a known-plaintext setting. In a chosen-plaintext setting, it would always be more efficient to directly query an affine space.

*Attack on EFX.* Thanks to Lemma 1, we can attack EFX with known-plaintext queries provided that we have almost all the codebook, bypassing the need for a vector space in the inputs.

**Theorem 4.** *There exists a quantum attack that, given* $(1-\mathcal{O}(1/n))2^n$ *classical* known-plaintext *queries to* EFX*, finds the complete key* $k, k_1, k_2$ *of the cipher in quantum time* $\mathcal{O}\left(n2^n + n^\omega 2^{\kappa/2}\right)$*.*

In particular, we can also attack an even more generic version of EFX, with three calls to independent block ciphers $E^1, E^2, E^3$. We call it DEFX, for *doubly-extended FX* (see Figure 2):

$$\mathsf{DEFX}(x) = E_k^3(k_2 \oplus E_k^2(k_1 \oplus E_k^1(x))) \ .$$

In this version, it suffices to remark that $\mathsf{DEFX}(x) = \mathsf{EFX}(E_k^1(x))$. We build states of the form $\sum_x |x\rangle \, |\mathsf{DEFX}(x)\rangle$ containing almost all the codebook. When we have guessed the right key $k$, we can map these states to:

$$\sum_x |E_k^1(x)\rangle \, |\mathsf{EFX}(E_k^1(x))\rangle = \sum_{x'} |x'\rangle \, |\mathsf{EFX}(x')\rangle \ ,$$

by applying $E_k^1$ in place on the first register, and continue the attack as before.

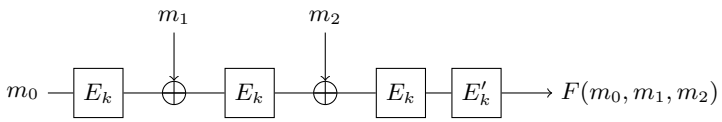


**Fig. 5.** Three-block ECBC-MAC.

### 4.4 Applications

The 2XOR-Cascade (2XOR for short) of [27] is an instance of EFX, and the results of Section 4.2 immediately apply. This construction can also appear in other situations.

*Encrypt-last-block-CBC-MAC with Unknown Plaintexts.* ECBC-MAC is an ISO standard [35, MAC algorithm 2], variant of CBC-MAC, where the output of CBC-MAC is reencrypted.

Let us consider a three-block ECBC-MAC (Figure 5):

$$m_0, m_1, m_2 \mapsto F(m_0, m_1, m_2) = E'_k(E_k(m_2 \oplus E_k(m_1 \oplus E_k(m_0)))) \ ,$$

with a block cipher $E$ of $n$ bits, $2n$ bits of key $k$, and $k' = \phi(k)$ is derived from $k$. Assume that the adversary observes $F(m_0, m_1, m_2)$ for known values of $m_0$ (for example, a nonce) and *fixed, but unknown* values of $m_1, m_2$.

Then the problem of recovering $k, m_1, m_2$ altogether is equivalent to attacking a DEFX construction where the cascade encryption with two different keys derived from $k$ is seen as another blockcipher with key $k : E'_k(E_k(x)) = E^2_k(x)$. More precisely, we assume that the adversary can query for $2^n(1-\alpha)$ values of $m_0$, where $\alpha = \mathcal{O}(1/n)$. In that case, Corollary 1 implies that any classical attack will require $\mathcal{O}\left(2^{5n/2}\right)$ computations. Our quantum attack has a time complexity $\mathcal{O}(n^\omega 2^n)$.

This means that, up to a polynomial factor, it is no harder for the quantum adversary to recover the key of this ECBC-MAC instance, although only the first block is known, than it would be in a chosen-plaintext scenario (where a direct quantum search of $k$ becomes possible).

This enhanced key-recovery attack applies as well if the first block is a nonce that the adversary does not choose (as soon as he is allowed $(1 - \mathcal{O}(1/n))2^n$ queries).

*Iterated Even-Mansour Ciphers.* A natural setting where this construction will occur is with iterated Even-Mansour ciphers with $r$ rounds, such as the one represented in Figure 6. They have been considered in a variety of contexts. In particular, a classical cryptanalysis of all 4-round such ciphers with two keys $k_0, k_1$, for all sequences of $k_0$ and $k_1$, is given in [22] (Table 2). For 4 rounds and two keys, offline-Simon does not seem to bring a more than quadratic improvement in any case. However, if the number of rounds increases, we can schedule the keys in order to reproduce a DEFX construction, for example with:

$$k_0, k_0, k_1, k_0, k_1, k_0 \ .$$

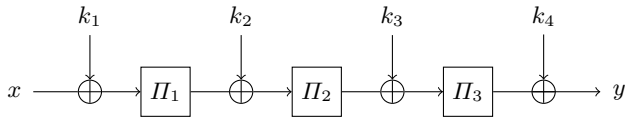


**Fig. 6.** An iterated Even-Mansour cipher with 4 keys. The $\Pi_i$ are independent $n$-bit permutations.

Here the best classical attack seems to be guessing $k_0$, then breaking the Even-Mansour scheme, in time $2^{3n/2}$. By Theorem 3, the quantum attack runs in time $\widetilde{\mathcal{O}}\left(2^{2n/3}\right)$ which represents a more-than-quadratic speedup.

While such constructions have been proposed, they tend to avoid these unfavorable key schedules. The LED-128 block cipher [29], which can be analyzed as an iterated Even-Mansour scheme [21], alternates only between its two subkeys $k_0$ and $k_1$. Also, note that in these applications, the quantum attacks do not go below the classical query complexity lower bound ($\mathcal{O}\left(2^{nr/(r+1)}\right)$ for $r$-round Even-Mansour ciphers).

## 5 Proving security

In this section, we show classical and quantum lower bounds on the security of EFX. We start with a classical proof of sPRP security.

### 5.1 Security Game

We want to prove the super Pseudorandom property of the EFX construction based on ideal ciphers. That means we allow an hypothetic adversary to do forward and inverse queries to ideal cipher oracles as well as an encryption oracle. In the real world, the three keys $k, k_1, k_2$ are first randomly drawn then the encryption oracle also makes use of the ideal cipher oracles to compute the output. In the ideal world, a new permutation is randomly drawn and used to produce the output. This is the sPRP security game as in Definition 1.

**Definition 1 (sPRP Security).** *Let $E_{1,k}(a)$ and $E_{2,k}(a)$ be two ideal ciphers with $\kappa$-bit key $k$ and $n$-bit input $a$, and $\mathcal{P}$ be the set of all $n$ to $n$ bit permutations. The sPRP security game advantage of an adversary for the EFX construction is defined as:*

$$\mathbf{Adv}^{sprp}_{EFX}(\mathcal{A}) = \mathbf{Pr}(\mathcal{A}^{E^{1/-1}_{\cdot,\cdot}(\cdot),EFX^{1/-1}_{k,k_1,k_2}(\cdot)} \to 1) - \mathbf{Pr}(\mathcal{A}^{E^{1/-1}_{\cdot,\cdot}(\cdot),p^{1/-1}(\cdot)} \to 1)\ .$$

*with the randomness of $k, k_1, k_2 \xleftarrow{\$} \{0,1\}^{\kappa+2n}$, $p \xleftarrow{\$} \mathcal{P}$, the ideal ciphers $E_1$, $E_2$, and $\mathcal{A}$.*

*Then, the sPRP security is the maximum advantage over all adversaries $\mathcal{A}$.*

**Transcript.** As the adversary makes queries to the oracles we record the interactions in a transcript. We denote $\mathcal{X}$ the set of all inputs of encryption queries and outputs of decryption queries with $D = |\mathcal{X}|$ the number of online queries. Conversely, $\mathcal{Y}$ is the set of all outputs of encryption and input of decryption. And $\mathcal{Q}_i^j$ is the set of all inputs of forward queries and output of backward queries to the ideal cipher $E_j$ parametrized with the key $i$ with $T_i^j = |\mathcal{Q}_i^j|$ and $T = \sum_{i\in\{0,1\}^\kappa; j\in\{1,2\}} T_i^j$ the total number of offline queries.

At the end of the interaction with the oracles, we help the adversary by providing additional information before the output decision. Hence we define the final transcript $\tau$ as:

$$\tau = \{k, k_1, k_2\} \cup \{(x, u, y), \forall x \in \mathcal{X}\} \cup \bigcup_{i\in\{0,1\}^\kappa; j\in\{1,2\}} \{(a, b), \forall a \in \mathcal{Q}_i^j\}$$

where $b = E_{j,i}(a)$ in both real and ideal worlds. In the real world,

$$y = \mathtt{EFX}_{k,k_1,k_2}(x) = k_2 \oplus E_{2,k}(k_1 \oplus E_{1,k}(x))$$

and, after interaction, we provide for the keys $k, k_1, k_2$ as well as the intermediary values $u = E_{1,k}(x)$ for all $x \in \mathcal{X}$. In the ideal world, $y = p(x)$ that is the output of a randomly chosen permutation and we simulate the keys and intermediate values after interaction as in Algorithm 2.

### 5.2 H-coefficient Technique

To prove Theorem 2, we will use the H-coefficient technique of Theorem 5.

**Theorem 5 (H-coefficient technique).** *Let $\mathcal{A}$ be a fixed computationally unbounded deterministic adversary that has access to either the real world oracle $\mathcal{O}_{\mathrm{re}}$ or the ideal world oracle $\mathcal{O}_{\mathrm{id}}$. Let $\Theta = \Theta_{\mathrm{g}} \sqcup \Theta_{\mathrm{b}}$ be some partition of the set of all attainable transcripts into* good *and* bad *transcripts. Suppose there exists $\epsilon_{\mathrm{ratio}} \geq 0$ such that for any $\tau \in \Theta_{\mathrm{g}}$,*

$$\frac{\mathbf{Pr}(X_{\mathrm{re}} = \tau)}{\mathbf{Pr}(X_{\mathrm{id}} = \tau)} \geq 1 - \epsilon_{\mathrm{ratio}}\,,$$

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

$$\mathbf{Pr}(\mathcal{A}^{\mathcal{O}_{\mathrm{re}}} \to 1) - \mathbf{Pr}(\mathcal{A}^{\mathcal{O}_{\mathrm{id}}} \to 1) \leq \epsilon_{\mathrm{ratio}} + \epsilon_{\mathrm{bad}}\,. \tag{2}$$

**Bad Transcripts** A transcript is said to be bad when Algorithm 2 return the empty set or when $T_k^1 + T_k^2 > \alpha T/2^\kappa$ for some value $\alpha$ to be determined later. Equivalently, a transcript is said to be bad when either $T_k^1 + T_k^2 > \alpha T/2^\kappa$ or $\exists (a, b, x) \in \mathcal{Q}_k^1 \times \mathcal{Q}_k^2 \times \mathcal{X} : a = b \oplus k_1, E_{1,k}(x) = a$ or $\exists (a, b, y) \in \mathcal{Q}_k^1 \times \mathcal{Q}_k^2 \times \mathcal{Y} : a = b \oplus k_1, y = E_{2,k}(b)$ or $\exists (x, a) \in (\mathcal{X} \cap \mathcal{Q}_k^1) \times \mathcal{Q}_k^2 : E_{1,k}(a) = p(x) \oplus k_2$.

Firstly, we bound the probability of $T_k^1 + T_k^2 > \alpha T/2^\kappa$ with the randomness of $k$ using the Markov inequality:

$$\mathbf{Pr}(T_k^1 + T_k^2 > \alpha T/2^\kappa) \leq 1/\alpha \tag{3}$$

**Algorithm 2** Building Ideal Transcripts

**input:** $\{(x, p(x)), \forall x \in \mathcal{X}\} \cup \bigcup_{i \in \{0,1\}^\kappa} \{(a, E_{t,i}(a)), \forall a \in \mathcal{Q}_i^t, t \in \{1,2\}\}$ .
**output:** $\{k, k_1, k_2\} \cup \{(x, u), \forall x \in \mathcal{X}\}$ .
**procedure** IdealTranscript
    $\{k, k_1, k_2\} \xleftarrow{\$} \{0,1\}^{n+2\kappa}$
    $\tau^\star \leftarrow \{k, k_1, k_2\}$
    $\mathcal{U} \leftarrow \emptyset$
    **for all** $a \in \mathcal{Q}_k^1$ **do**
        $\mathcal{U} \leftarrow \mathcal{U} \cup \{E_{1,k}(a)\}$
    **end for**
    **for all** $a \in \mathcal{Q}_k^2$ **do**
        **if** $a \oplus k_1 \in \mathcal{U}$ **then**
            **if** $E_{1,k}^{-1}(a \oplus k_1) \in \mathcal{X}$ **or** $\exists x \in \mathcal{X} : E_{2,k}(a) \oplus k_2 = p(x)$ **then**
                **return** $\emptyset$ ▷ Bad Event
            **end if**
        **else**
            $\mathcal{U} \leftarrow \mathcal{U} \cup \{a \oplus k_1\}$
        **end if**
    **end for**
    **for all** $x \in \mathcal{X}$ **do**
        **if** $x \in \mathcal{Q}_k^1$ **and** $\exists a \in \mathcal{Q}_k^2 : E_{2,k}(a) = p(x) \oplus k_2$ **then**
            **return** $\emptyset$ ▷ Bad Event
        **else if** $x \in \mathcal{Q}_k^1$ **then**
            $\tau^\star \leftarrow \tau^\star \cup \{(x, E_{1,k}(x))\}$
        **else if** $\exists a \in \mathcal{Q}_k^2 : E_k(a) = p(x) \oplus k_2$ **then**
            $\tau^\star \leftarrow \tau^\star \cup \{(x, a \oplus k_1)\}$
        **else**
            $u \xleftarrow{\$} \{0,1\}^n / \mathcal{U}$
            $\mathcal{U} \leftarrow \mathcal{U} \cup \{u\}$
            $\tau^\star \leftarrow \tau^\star \cup \{(x, u)\}$
        **end if**
    **end for**
    **return** $\tau^\star$
**end procedure**

Then we bound the probability of $\exists(a,b,x) \in \mathcal{Q}_k^1 \times \mathcal{Q}_k^2 \times \mathcal{X} : a = b \oplus k_1, E_{1,k}(x) = a$ with the randomness of $k$ and $k_1$:

$$\begin{aligned}
&\mathbf{Pr}(\exists(a,b,x) \in \mathcal{Q}_k^1 \times \mathcal{Q}_k^2 \times \mathcal{X} : a = b \oplus k_1, E_{1,k}(x) = a) \\
=&\sum_{i \in \{0,1\}^\kappa} \mathbf{Pr}(\exists(a,b,x) \in \mathcal{Q}_i^1 \times \mathcal{Q}_i^2 \times \mathcal{X} : a = b \oplus k_1, E_{1,i}(x) = a)\mathbf{Pr}(k = i) \\
\leq& 2^{-\kappa} \sum_{i \in \{0,1\}^\kappa} \min\left(\frac{\min(T_i^1, D) \cdot T_i^2}{2^n}, 1\right) \\
\leq& 2^{-\kappa-n} \sum_{i \in \{0,1\}^\kappa} \min\left(T_i^1 \cdot T_i^2, D \cdot T_i^2, 2^n\right)
\end{aligned}$$

As we wish to get a born depending on $T$ but not on the repartition of the offline queries $T_i^j$, we assume the worst case that is the repartition giving the highest value. Notice that $\sum_i (T_i^1 \cdot T_i^2)$ can be optimized by maximizing a few terms. In our case, we obtain an upper-bound by letting $T_i^1 = T_i^2 = \min(D, 2^{n/2})$ for $T/(2 \cdot \min(D, 2^{n/2}))$ different values of $i$ and $T_i^1 = T_i^2 = 0$ otherwise (the strategy of optimizing the values up to $\max(2^n/D, 2^{n/2})$ gives the same bound.):

$$\mathbf{Pr}(\exists(a,b,x) \in \mathcal{Q}_k^1 \times \mathcal{Q}_k^2 \times \mathcal{X} : a = b \oplus k_1, E_{1,k}(x) = a) \leq \frac{T \cdot \min(D, 2^{n/2})}{2^{\kappa+n+1}} \quad (4)$$

We can derive the same bound the same way for the two remaining bad events $\exists(a,b,y) \in \mathcal{Q}_k^1 \times \mathcal{Q}_k^2 \times \mathcal{Y} : a = b \oplus k_1, y = E_{2,k}(b)$ and $\exists(x,a) \in (\mathcal{X} \cap \mathcal{Q}_k^1) \times \mathcal{Q}_k^2 : E_{1,k}(a) = p(x) \oplus k_2$. Putting it together:

$$\epsilon_{\text{bad}} = \mathbf{Pr}(\tau \text{ is bad}) \leq \frac{1}{\alpha} + 3\frac{T \cdot \min(D, 2^{n/2})}{2^{\kappa+n+1}} \quad (5)$$

**Good Transcripts** Assuming that $\tau$ is a good transcript, we want to upper-bound the ratio between the probabilities of $\tau$ happening in the real world and in the ideal world. Let $\mathcal{A} = \{E_{2,k}(a) \oplus k_2 : a \in \mathcal{Q}_k^2\}$ and $\mathcal{B} = \{E_{1,k}(b) \oplus k_1 : b \in \mathcal{Q}_k^1\}$.

In the real world, the probability comes from the drawing of the keys and from every fresh queries to the ideal block cipher oracles:

$$\begin{aligned}
&1/\mathbf{Pr}(X_{\text{re}} = \tau) = \\
&2^{\kappa+2n} \left(\prod_{i \in \{0,1\}^\kappa / \{k\}; j \in \{1,2\}} (2^n)_{(T_i^j)}\right) \left((2^n)_{(|\mathcal{Q}_k^1 \cup \mathcal{X}|)} \cdot (2^n)_{(|\mathcal{A} \cup \mathcal{Y}|)}\right)
\end{aligned}$$

In the ideal world, the probability comes from the ideal block cipher oracles, the encryption oracle and Algorithm 2:

$$1/\mathbf{Pr}(X_{\mathrm{id}} = \tau) =$$
$$2^{\kappa+2n} \left( \prod_{i\in\{0,1\}^\kappa; j\in\{1,2\}} (2^n)_{(T_i^j)} \right) \left( (2^n)_{(D)} \cdot (2^n - |\mathcal{B} \cup \mathcal{Q}_k^2|)_{(D-|\mathcal{Q}_k^1\cap\mathcal{X}|-|\mathcal{A}\cap\mathcal{Y}|)} \right)$$

Putting it together:

$$\begin{aligned}
\frac{\mathbf{Pr}(X_{\mathrm{re}} = \tau)}{\mathbf{Pr}(X_{\mathrm{id}} = \tau)} &\geq \frac{(2^n)_{(T_k^1)}(2^n)_{(T_k^2)}(2^n)_{(D)}(2^n - |\mathcal{B} \cup \mathcal{Q}_k^2|)_{(D-|\mathcal{Q}_k^1\cap\mathcal{X}|-|\mathcal{A}\cap\mathcal{Y}|)}}{(2^n)_{(|\mathcal{Q}_k^1\cup\mathcal{X}|)}(2^n)_{(|\mathcal{A}\cup\mathcal{Y}|)}} \\
&\geq \frac{(2^n)_{(T_k^1)}(2^n)_{(T_k^2)}(2^n)_{(D)}(2^n - T_k^1 - T_k^2 + |\mathcal{B} \cap \mathcal{Q}_k^2|)_{(D-|\mathcal{Q}_k^1\cap\mathcal{X}|-|\mathcal{A}\cap\mathcal{Y}|)}}{(2^n)_{(D+T_k^1-|\mathcal{Q}_k^1\cap\mathcal{X}|)}(2^n)_{(D+T_k^2-|\mathcal{A}\cap\mathcal{Y}|)}} \\
&\geq \frac{(2^n)_{(T_k^1)}(2^n)_{(T_k^2)}(2^n)_{(D)}(2^n - T_k^1 - T_k^2)_{(D-|\mathcal{Q}_k^1\cap\mathcal{X}|-|\mathcal{A}\cap\mathcal{Y}|)}}{(2^n)_{(D+T_k^1-|\mathcal{Q}_k^1\cap\mathcal{X}|)}(2^n)_{(D+T_k^2-|\mathcal{A}\cap\mathcal{Y}|)}}
\end{aligned}$$

First notice that when $D = 2^n$ then $\mathcal{X} = \mathcal{Y} = \{0,1\}^n$ and we have $\frac{\mathbf{Pr}(X_{\mathrm{re}}=\tau)}{\mathbf{Pr}(X_{\mathrm{id}}=\tau)} \geq 1$. Thus we can derive a first bound independent of $D$ ignoring the first bad event (or taking a very high $\alpha$):

$$\mathbf{Adv}^{\mathsf{sprp}}_{\mathsf{EFX}}(\mathcal{A}) \leq 3\frac{T}{2^{\kappa+n/2+1}}$$

We have to work a bit more to get a bound for when $D \leq 2^{n/2}$:

$$\begin{aligned}
\frac{\mathbf{Pr}(X_{\mathrm{re}} = \tau)}{\mathbf{Pr}(X_{\mathrm{id}} = \tau)} &\geq \frac{(2^n)_{(T_k^1)}(2^n)_{(T_k^2)}(2^n)_{(D)}(2^n - T_k^1 - T_k^2)_{(D)}}{(2^n)_{(D+T_k^1)}(2^n)_{(D+T_k^2)}} \\
&\geq \frac{(2^n)_{(D)}(2^n - T_k^1 - T_k^2)_{(D)}}{(2^n - T_k^1)_{(D)}(2^n - T_k^2)_{(D)}} \\
&\geq \left( \frac{(2^n - D + 1)(2^n - T_k^1 - T_k^2 - D + 1)}{(2^n - T_k^1 - D + 1)(2^n - T_k^2 - D + 1)} \right)^D \\
&\geq \left( 1 + \frac{T_k^1 \cdot T_k^2}{(2^n - D + 1)(2^n - T_k^1 - T_k^2 - D + 1)} \right)^{-D} \\
&\geq 1 - \frac{D \cdot T_k^1 \cdot T_k^2}{(2^n - D + 1)(2^n - T_k^1 - T_k^2 - D + 1)}
\end{aligned}$$

Adding the fact that $T_k^1 + T_k^2$ is upper-bounded by $\alpha T/2^\kappa$ we get:

$$\epsilon_{\mathrm{ratio}} \leq \frac{\alpha^2 T^2 D}{2^{2\kappa+2}(2^n - D + 1)(2^n - \alpha T/2^\kappa - D + 1)} \tag{6}$$

**Conclusion** Hence using the H-coefficient Technique of Theorem 5 we get two upper-bound for the advantage of a classical information theoretic adversary. One mostly useful for $D \leq 2^{n/2}$:

$$\mathbf{Adv}^{\mathsf{sprp}}_{\mathtt{EFX}}(\mathcal{A}) \leq \frac{1}{\alpha} + 3\frac{T \cdot \min(D, 2^{n/2})}{2^{\kappa+n+1}} + \frac{\alpha^2 T^2 D}{2^{2\kappa+2}(2^n - D + 1)(2^n - \alpha T/2^\kappa - D + 1)}$$

And one independent of $D$:

$$\mathbf{Adv}^{\mathsf{sprp}}_{\mathtt{EFX}}(\mathcal{A}) \leq 3\frac{T}{2^{\kappa+n/2+1}}$$

Note that we are free to choose the value $\alpha$ to optimize the bound. We decided to take $1/\alpha = \left(T^2 D/2^{2(\kappa+n)}\right)^{\frac{1}{3}}$ and that concludes the proof of Theorem 3.

### 5.3 Quantum lower bound

For completeness, we also prove quantum lower bounds on the security of EFX. The bounds we obtain are weak, in the sense that the security of the construct matches the security of its underlying primitive. However, they are also tight, as the offline-Simon algorithm makes for a matching upper bound.

We prove security in the *quantum ideal cipher model*, introduced in [34]. As for the ideal cipher model, we allow encryption and decryption queries to the block cipher $E^{\pm}$ and to the construction $C^{\pm}$. The only difference is that quantum queries are allowed, instead of classical queries. This means we prove security with quantum access to the construction $C^{\pm}$, which implies the bound when access to $C^{\pm}$ is only classical.

We note $\mathcal{C}(k, n)$ the distributions of $n$-bit block, $k$-bit key block ciphers, and $\mathcal{P}(n)$ the distribution of $n$-bit permutations. We will prove in this section the indistiguishability between $E^{\pm}, C^{\pm}$ with $E^{\pm} \in \mathcal{C}(k, n)$ and $E^{\pm}, P^{\pm}$ with $E^{\pm} \in \mathcal{C}(m, n)$ and $P^{\pm} \in \mathcal{P}(n)$ up to $2^{\kappa/2}$ queries.

We rely on the hardness of unstructured search:

**Lemma 2 (Optimality of Grover's algorithm [54]).** *Let $D_0$ be the degenerate distribution containing only the $\kappa$-bit input all-zero function, and $D_1$ be the distribution of $\kappa$-bit input boolean functions with only one output equal to 1. Then, for any quantum adversary $\mathcal{A}$ that does at most $q$ queries,*

$$\mathbf{Adv}^{\mathsf{dist}}_{D_0, D_1}(\mathcal{A}) \leq \frac{4q^2}{2^\kappa} \quad .$$

We will now reduce the problem of distinguishing the construction from a random permutation to the unstructured search distinguisher.

**Lemma 3 (Distinguishing the EFX construction).** *Let $E_1, E_2 \xleftarrow{\$} \mathcal{C}(\kappa, n)^2$, $P \xleftarrow{\$} \mathcal{P}(n)$, $K_1, K_2 \xleftarrow{\$} \{0,1\}^{\kappa+n}$, $\mathit{EFX} = E_2(K_1, E_1(K_1, x \oplus K_2) \oplus K_2)$. Then for any quantum adversary $\mathcal{A}$ that does at most $q$ queries,*

$$\mathbf{Adv}^{\mathsf{dist}}_{(E_1, E_2, \mathit{EFX}), (E_1, E_2, P)}(\mathcal{A}) \leq \frac{4q^2}{2^\kappa} \quad .$$

*Proof.* We want to reduce this distinguishing problem to the previous one. First, as in the classical proof, we can remark that the distributions of $E_1, E_2, \texttt{EFX}$ is equal to the distribution of $F, E_2, P$ with $E_2 \xleftarrow{\$} \mathcal{C}(k,n)$, $P \xleftarrow{\$} \mathcal{P}(n)$, $F(K_1, x) = P(E_2^{-1}(K_1, x \oplus K_2) \oplus K_2)$, and for other $K$, $F(K,x) \xleftarrow{\$} \mathcal{P}(n)$.

Hence, we can consider the following construction: we take $E_1, E_2 \in \mathcal{C}(\kappa, n)^2$, $P \in \mathcal{P}(n)$, $f \in \{0,1\}^\kappa \to \{0,1\}$, $K_2 \in \{0,1\}^n$. We can construct

$$F(K,x) = \begin{cases} P(E_2^{-1}(K, x \oplus K_2) \oplus K_2) & \text{if } f(K) = 1 \\ E_1(K,x) & \text{otherwise} \end{cases}.$$

Now, we can leverage Lemma 2 on the distribution of $(F, E_2, P)$: if $f$ is all-zero, we have the distribution of $(E_1, E_2, P)$. If $f$ has a unique 1, we have the distribution of $(E_1, E_2, \texttt{EFX})$. Hence, any adversary that distinguishes EFX can also distinguish unstructured search.

*Remark 4.* This proof can be directly adapted to the case where we only have one cipher, but two related keys are used.

*Remark 5 (Tightness).* This bound is tight when quantum query access is allowed. With only classical query access, the attack matches the bound only when $n \leq \kappa/2$. To prove security for smaller $n$ (or with a lower amount of classical data), one could adapt the quantum security proofs for the FX construction [37], as the construction of interest is FX plus an additional encryption.

## 6 On the Maximal Gap

As we have recalled above, exponential speedups can be obtained when the quantum adversary can make superposition queries. For classical queries in symmetric cryptography, the best speedup remained quadratic for a long time. It is likely to remain polynomial, but as we manage to reach a 2.5 gap, it is natural to ask by how much we might extend it. In this section, we connect this question to known results in quantum query complexity. We show that the offline-Simon technique should be able to reach a cubic speedup, but without any cryptographic application at the moment.

Note that if we formulate the question only as "largest speedup when only classical queries are given", it will not properly represent the class of symmetric cryptography attacks that we are interested in. Indeed, Shor's algorithm provides an exponential speedup on a problem with only classical queries.

However, there is still a major difference, in that we are interested in constructions *with security proofs in the ideal model* (e.g., ideal ciphers, random oracles, random permutations). Here the definition of a *largest gap* is more reasonable: all the quantum speedups known are polynomial at best. Besides, we can focus on *query complexity* only, without making any consideration on the memory used or time efficiency of the algorithms.

### 6.1 Relation with Query Complexity

The question of finding the *largest possible gap* in our context bears some similarities with the question of comparing randomized and quantum query complexities of total boolean functions. In this setting, the best gap known is cubic, which follows from [2] and [7,50]. Initially, the technique of *cheat sheets* developed in [2] allowed the authors to obtain a gap 2.5. We will explain the reasons behind this coincidence.

*Definitions.* First of all, we need to recapitulate some essential definitions and results of query complexity. We will focus only on a very restricted subset of results. Let us consider a boolean function $f : \{0,1\}^N \to \{0,1\}$. The definition of $f$ is known, and the only way to evaluate it is then to know some bits of its input string $x_0, \ldots, x_{N-1}$. Here, $N$ can be thought of as an exponential number.

When $f$ is defined over all its input, we call it a *total* function, as opposed to a *partial* function defined only over some domain $D \subseteq \{0,1\}^N$. For example, the $\mathsf{or}_N : \{0,1\}^n \to \{0,1\}$ function computes the $\mathsf{OR}$ of all its bits.

For any $f$, we define:

- the *deterministic* query complexity $D(f)$: the minimum number of queries that have to be made by a deterministic algorithm computing $f(x)$ on every input $x$;
- the *bounded-error randomized* query complexity $R(f)$: the minimum number of queries made by a randomized algorithm that outputs $f(x)$ with probability at least $2/3$ *on every input* $x$;
- the *quantum* query complexity $Q(f)$: the minimum number of queries made by a quantum algorithm that outputs $f(x)$ with probability $2/3$.

For example, the classical query complexity of $\mathsf{or}_N$ is $N$, and its quantum query complexity is $\Theta(\sqrt{N})$ (thanks to Grover's algorithm and its matching lower bound).

Clearly, we have in general $Q(f) \leq R(f) \leq D(f)$. In classical cryptography we are usually interested in the measure $R(f)$, and in post-quantum cryptography in $Q(f)$. It has been known for a long time that for *total* boolean functions, polynomial relations hold between these measures. In particular, Beals et al. [8] showed that for any total function $f$, $D(f) = \mathcal{O}\big(Q(f)^6\big)$, and so $R(f) = \mathcal{O}\big(Q(f)^6\big)$. This was improved very recently in [3] to $D(f) = \mathcal{O}\big(Q(f)^4\big)$ (and so $R(f) = \mathcal{O}\big(Q(f)^4\big)$). The quartic relation with $D(f)$ is tight (by a separation given in [4]), but the best proven gap with $R(f)$ is cubic only, and this is conjectured in [4] to be optimal.

*Promise Problems.* These results underlie the idea that quantum speedups "need structure": indeed, an exponential quantum speedup can occur only if $f$ assumes some *promise* on its input (for example for Simon's algorithm, that it encodes a periodic function). Let us now take an example: the attack on $\mathsf{EFX}$ of Theorem 3.

Recovering the key of an $\mathsf{EFX}$ instance *could* be seen as computing a boolean function $f$ with a promise. It would be done as follows: the input of the function encodes $\mathsf{EFX}_{k,k_1,k_2}[E](x)$ for all $x$ and $E_z(x)$ for all $(z,x)$; that is, the complete

tables of the EFX cipher and the ideal cipher upon which it is built. The function must compute the key $k, k_1, k_2$ used in EFX. Although the second table ($E$) could be any value, since any block cipher can be selected at random, the function satisfies the promise that the first table actually encodes EFX[$E$].

There is, however, a significant difference between the query complexity of $f$ and the security of EFX. The proof of security in the ideal cipher model reasons about adversaries as *average-case* algorithms. Similarly, the classical and quantum attacks work *on average* over all ciphers $E$. Typically, when running the Grover-meet-Simon attack, there are bad cases, corresponding to some rare choices of $E$, in which the algorithm will not be able to return the key. But the relations in query complexity concern only worst-case complexities. Indeed, it was shown in [5] that no polynomial relation holds between the average-case complexities of total functions. Our attack is an average-case algorithm, and so, we cannot say anything about $Q(f)$.

*2.5 Separation Result.* In [2], the authors proved the existence of a total function $f$ for which $R(f) = \widetilde{\Omega}\left(N^{2.5}\right)$ and $Q(f) = \widetilde{\mathcal{O}}(N)$. Since this 2.5 exponent is reminiscent of ours, we briefly review how it was obtained.

The authors start by defining a function with a promise, by composing Forrelation (a promise problem) and And-Or (a boolean function which has a provable quadratic quantum speedup). We do not need to define Forrelation here. Simon's problem could have been used instead, as a speedup poly($n$) vs. $\mathcal{O}\left(2^{n/2}\right)$ is sufficient.

By combining And-Ors of size $N^2$ with a Forrelation of size $N$, one obtains a quantum algorithm running in time $Q(f) = \widetilde{\mathcal{O}}(N)$, because Forrelation requires $\widetilde{\mathcal{O}}(1)$ queries and And-Or requires $\mathcal{O}(N)$ queries using Grover's algorithm. The corresponding classical algorithm runs in time $\widetilde{\mathcal{O}}\left(N^{2.5}\right)$, due to the gap in both problems. Next, the authors introduce a generic *cheat sheet* framework which allows to turn partial functions into total ones. The *cheat sheet* variant of a function $f$, $f_{CS}$, is more costly. But this additional cost comes from a *certificate* function, which checks if the input satisfies the promise. In the case studied in [2], the certificate simply consists in checking the outputs of the And-Ors, and checking that the Forrelation instance satisfies its promise: all of this can be done in quantum time $\widetilde{\mathcal{O}}(N)$. So the *cheat sheet* variant of the above function provides the said query complexity gap.

The offline-Simon attack does actually the opposite of the function above. Instead of computing a Simon instance out of many individual And-Or results, it computes an Or of many independent Simon instances: we are looking for the single periodic function in a family of functions. This is why the 2.5 exponents coincide.

Besides, since we want to make only classical queries, we have to pay an additional cost $N$ corresponding to the classical queries to EFX. This additional cost coincides with the cost of verifying the Forrelation instance. This is why, similarly to the cheat sheet technique, the offline-Simon structure will allow a cubic gap at most. Yet, these are only similarities, as there is no connection between worst-case and average-case algorithms.

*Cubic Separation Result.* As written above, only a quartic relation between $Q(f)$ and $R(f)$ for total functions is proven, while the best separation known at the moment is cubic (and this is conjectured to be optimal). It stems from replacing the Forrelation problem in [2] by *k-fold forrelation* [1]. For any $k$, *k-fold forrelation* has a classical query complexity $\widetilde{\Omega}\left(2^{n(1-1/k)}\right)$ and a quantum query complexity $\mathcal{O}(k)$. This gap was conjectured in [1] and recently proven in [7,50]. Note that *k-fold forrelation* realizes an *optimal gap* between randomized and quantum query complexities for *partial* functions.

### 6.2 Improving the Gap in Offline-Simon

In full generality, the offline-Simon algorithm can be seen as an algorithm that:

- queries a construction $F$ with unknown key, and populates a table with these queries
- searches for some secret key $k$ using a quantum search where, in order to test a given $k$, queries to the table are made, and a *superposition attack* on some construction is launched.

In particular, when attacking FX with classical queries only, each iteration of the quantum search reproduces the attack on the Even-Mansour cipher – and uses Simon's algorithm. But we could take this design more generally, and replace the Even-Mansour attack by any other attack using superposition queries. Thus, there is a link between the maximal gap achievable by the offline strategy and the maximal gap of superposition attacks.

The gap in the Even-Mansour attack is $\mathsf{poly}(n)$ vs. $\mathcal{O}\left(2^{n/2}\right)$. We could try to increase it up to $\mathsf{poly}(n)$ vs. $\mathcal{O}(2^n)$. This is the best we can hope for, because we consider an n-bit construction: $\mathcal{O}(2^n)$ is its maximal query complexity. All exponential speedups in quantum cryptanalysis that we know to date, including Q2 attacks on symmetric primitives, and attacks on asymmetric schemes, are based on variants of Simon's and Shor's algorithms. The classical counterpart of these algorithm is a collision search and, as such, they only reach a speedup $\mathsf{poly}(n)$ vs. $\mathcal{O}\left(2^{n/2}\right)$ at best.

However, we can replace this problem by *k-fold forrelation*, and take advantage of its enhanced gap $\mathsf{poly}(n)$ vs. $\mathcal{O}\left(2^{n(1-\epsilon)}\right)$. We conjecture that this gives us a cubic speedup. However, forrelation is not a problem that arises naturally in cryptography. Finding a cryptographically relevant example of a gap between 2.5 and 3 is an interesting open question.

## 7 Conclusion

In this paper, we gave the first example of a more than quadratic speedup of a symmetric cryptanalytic attack in the classical query model. This 2.5 speedup is actually provable in the ideal cipher model. It is a direct counterexample to the folklore belief that *doubling the key sizes* of symmetric constructions is sufficient to protect against quantum attackers. In particular, generic key-length

extension techniques should be carefully analyzed: the 2XOR Cascade proposed in [27] offers practically no additional security in the quantum setting.

The most obvious open question is by how much this gap may be increased. The algorithm we used, offline-Simon, does not seem capable of reaching more than a 2.5 gap. Although a cubic separation seems achievable, we couldn't manage to obtain one with problems of cryptographic interest. This is reminiscent of the cubic gap which is conjectured to be the best achievable between the randomized and quantum query complexities of total functions [2]. However, there is a stark difference between the problems at stake, and in our case, it is not even known if a polynomial relation holds in general.

**Acknowledgements.** The authors would like to thank Akinori Hosoyamada and the reviewers of EUROCRYPT 2022 for helpful comments. A.S. has been supported by the ERC Advanced Grant N° 740972 (ALGSTRONGCRYPTO).

## References

1. Aaronson, S., Ambainis, A.: Forrelation: A problem that optimally separates quantum from classical computing. SIAM J. Comput. 47(3), 982–1038 (2018)
2. Aaronson, S., Ben-David, S., Kothari, R.: Separations in query complexity using cheat sheets. In: STOC. pp. 863–876. ACM (2016)
3. Aaronson, S., Ben-David, S., Kothari, R., Tal, A.: Quantum implications of huang's sensitivity theorem. Electron. Colloquium Comput. Complex. 27, 66 (2020)
4. Ambainis, A., Balodis, K., Belovs, A., Lee, T., Santha, M., Smotrovs, J.: Separations in query complexity based on pointer functions. In: STOC. pp. 800–813. ACM (2016)
5. Ambainis, A., de Wolf, R.: Average-case quantum query complexity. In: STACS. Lecture Notes in Computer Science, vol. 1770, pp. 133–144. Springer (2000)
6. Anand, M.V., Targhi, E.E., Tabia, G.N., Unruh, D.: Post-quantum security of the CBC, CFB, OFB, CTR, and XTS modes of operation. In: PQCrypto. Lecture Notes in Computer Science, vol. 9606, pp. 44–63. Springer (2016)
7. Bansal, N., Sinha, M.: k-forrelation optimally separates quantum and classical query complexity. In: STOC. pp. 1303–1316. ACM (2021)
8. Beals, R., Buhrman, H., Cleve, R., Mosca, M., de Wolf, R.: Quantum lower bounds by polynomials. J. ACM 48(4), 778–797 (2001)
9. Bennett, C.H.: Time/space trade-offs for reversible computation. SIAM J. Comput. 18(4), 766–776 (1989)
10. Bonnetain, X.: Quantum key-recovery on full AEZ. In: SAC. Lecture Notes in Computer Science, vol. 10719, pp. 394–406. Springer (2017)
11. Bonnetain, X.: Tight bounds for Simon's algorithm. In: Longa, P., Ràfols, C. (eds.) LATINCRYPT 2021. Lecture Notes in Computer Science, Springer (2021)
12. Bonnetain, X., Hosoyamada, A., Naya-Plasencia, M., Sasaki, Y., Schrottenloher, A.: Quantum attacks without superposition queries: The offline simon's algorithm. In: ASIACRYPT (1). Lecture Notes in Computer Science, vol. 11921, pp. 552–583. Springer (2019)
13. Bonnetain, X., Jaques, S.: Quantum period finding against symmetric primitives in practice. IACR Trans. Cryptogr. Hardw. Embed. Syst. 2022(1) (2021)
14. Bonnetain, X., Naya-Plasencia, M., Schrottenloher, A.: On quantum slide attacks. In: SAC. Lecture Notes in Computer Science, vol. 11959, pp. 492–519. Springer (2019)

15. Bonnetain, X., Naya-Plasencia, M., Schrottenloher, A.: Quantum security analysis of AES. IACR Trans. Symmetric Cryptol. 2019(2), 55–93 (2019)
16. Brassard, G., Hoyer, P., Mosca, M., Tapp, A.: Quantum amplitude amplification and estimation. Contemporary Mathematics 305, 53–74 (2002)
17. Chauhan, A.K., Kumar, A., Sanadhya, S.K.: Quantum free-start collision attacks on double block length hashing with round-reduced AES-256. IACR Trans. Symmetric Cryptol. 2021(1), 316–336 (2021)
18. Cid, C., Hosoyamada, A., Liu, Y., Sim, S.M.: Quantum cryptanalysis on contracting feistel structures and observation on related-key settings. In: INDOCRYPT. Lecture Notes in Computer Science, vol. 12578, pp. 373–394. Springer (2020)
19. Daemen, J.: Limitations of the even-mansour construction. In: ASIACRYPT. Lecture Notes in Computer Science, vol. 739, pp. 495–498. Springer (1991)
20. Dinur, I.: Cryptanalytic time-memory-data tradeoffs for fx-constructions with applications to PRINCE and PRIDE. In: EUROCRYPT (1). Lecture Notes in Computer Science, vol. 9056, pp. 231–253. Springer (2015)
21. Dinur, I., Dunkelman, O., Keller, N., Shamir, A.: Key recovery attacks on 3-round even-mansour, 8-step led-128, and full AES2. In: ASIACRYPT (1). Lecture Notes in Computer Science, vol. 8269, pp. 337–356. Springer (2013)
22. Dinur, I., Dunkelman, O., Keller, N., Shamir, A.: Cryptanalysis of iterated even-mansour schemes with two keys. In: ASIACRYPT (1). Lecture Notes in Computer Science, vol. 8873, pp. 439–457. Springer (2014)
23. Dong, X., Sun, S., Shi, D., Gao, F., Wang, X., Hu, L.: Quantum collision attacks on aes-like hashing with low quantum random access memories. In: ASIACRYPT (2). Lecture Notes in Computer Science, vol. 12492, pp. 727–757. Springer (2020)
24. Dunkelman, O., Keller, N., Shamir, A.: Minimalism in cryptography: The even-mansour scheme revisited. In: EUROCRYPT. Lecture Notes in Computer Science, vol. 7237, pp. 336–354. Springer (2012)
25. Even, S., Mansour, Y.: A construction of a cipher from a single pseudorandom permutation. J. Cryptol. 10(3), 151–162 (1997)
26. Gazi, P., Lee, J., Seurin, Y., Steinberger, J.P., Tessaro, S.: Relaxing full-codebook security: A refined analysis of key-length extension schemes. In: FSE. Lecture Notes in Computer Science, vol. 9054, pp. 319–341. Springer (2015)
27. Gazi, P., Tessaro, S.: Efficient and optimally secure key-length extension for block ciphers via randomized cascading. In: EUROCRYPT. Lecture Notes in Computer Science, vol. 7237, pp. 63–80. Springer (2012)
28. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: STOC. pp. 212–219. ACM (1996)
29. Guo, J., Peyrin, T., Poschmann, A., Robshaw, M.J.B.: The LED block cipher. In: CHES. Lecture Notes in Computer Science, vol. 6917, pp. 326–341. Springer (2011)
30. Hosoyamada, A., Sasaki, Y.: Cryptanalysis against symmetric-key schemes with online classical queries and offline quantum computations. In: CT-RSA. Lecture Notes in Computer Science, vol. 10808, pp. 198–218. Springer (2018)
31. Hosoyamada, A., Sasaki, Y.: Quantum demirci-selçuk meet-in-the-middle attacks: Applications to 6-round generic feistel constructions. In: SCN. Lecture Notes in Computer Science, vol. 11035, pp. 386–403. Springer (2018)
32. Hosoyamada, A., Sasaki, Y.: Finding hash collisions with quantum computers by using differential trails with smaller probability than birthday bound. In: EUROCRYPT (2). Lecture Notes in Computer Science, vol. 12106, pp. 249–279. Springer (2020)

33. Hosoyamada, A., Sasaki, Y.: Quantum collision attacks on reduced SHA-256 and SHA-512. In: CRYPTO (1). Lecture Notes in Computer Science, vol. 12825, pp. 616–646. Springer (2021)
34. Hosoyamada, A., Yasuda, K.: Building quantum-one-way functions from block ciphers: Davies-meyer and merkle-damgård constructions. In: ASIACRYPT (1). LNCS, vol. 11272, pp. 275–304. Springer (2018)
35. ISO Central Secretary: Information technology – Security techniques – Message Authentication Codes (MACs) – Part 1: Mechanisms using a block cipher. Standard ISO/IEC 9797-1:2011, International Organization for Standardization, Geneva, CH (Mar 2011), https://www.iso.org/standard/50375.html
36. Ito, G., Hosoyamada, A., Matsumoto, R., Sasaki, Y., Iwata, T.: Quantum chosen-ciphertext attacks against feistel ciphers. In: CT-RSA. Lecture Notes in Computer Science, vol. 11405, pp. 391–411. Springer (2019)
37. Jaeger, J., Song, F., Tessaro, S.: Quantum key-length extension. CoRR abs/2105.01242 (2021)
38. Kaplan, M., Leurent, G., Leverrier, A., Naya-Plasencia, M.: Breaking symmetric cryptosystems using quantum period finding. In: CRYPTO (2). Lecture Notes in Computer Science, vol. 9815, pp. 207–237. Springer (2016)
39. Kaplan, M., Leurent, G., Leverrier, A., Naya-Plasencia, M.: Quantum differential and linear cryptanalysis. IACR Trans. Symmetric Cryptol. 2016(1), 71–94 (2016)
40. Kilian, J., Rogaway, P.: How to protect DES against exhaustive key search. In: CRYPTO. Lecture Notes in Computer Science, vol. 1109, pp. 252–267. Springer (1996)
41. Kilian, J., Rogaway, P.: How to protect DES against exhaustive key search (an analysis of DESX). Journal of Cryptology 14(1), 17–35 (Jan 2001)
42. Knill, E.: An analysis of bennett's pebble game. CoRR abs/math/9508218 (1995)
43. Kuwakado, H., Morii, M.: Quantum distinguisher between the 3-round feistel cipher and the random permutation. In: ISIT. pp. 2682–2685. IEEE (2010)
44. Kuwakado, H., Morii, M.: Security on the quantum-type even-mansour cipher. In: ISITA. pp. 312–316. IEEE (2012)
45. Leander, G., May, A.: Grover meets simon - quantumly attacking the fx-construction. In: ASIACRYPT (2). Lecture Notes in Computer Science, vol. 10625, pp. 161–178. Springer (2017)
46. Levin, R.Y., Sherman, A.T.: A note on Bennett's time-space tradeoff for reversible computation. SIAM J. Comput. 19(4), 673–677 (1990)
47. Magniez, F., Nayak, A., Roland, J., Santha, M.: Search via quantum walk. SIAM J. Comput. 40(1), 142–164 (2011)
48. National Academies of Sciences, Engineering, and Medicine: Quantum Computing: Progress and Prospects. The National Academies Press, Washington, DC (2018)
49. Nielsen, M.A., Chuang, I.: Quantum computation and quantum information (2002)
50. Sherstov, A.A., Storozhenko, A.A., Wu, P.: An optimal separation of randomized and quantum query complexity. In: STOC. pp. 1289–1302. ACM (2021)
51. Shor, P.W.: Algorithms for quantum computation: Discrete logarithms and factoring. In: FOCS. pp. 124–134. IEEE Computer Society (1994)
52. Simon, D.R.: On the power of quantum computation. SIAM J. Comput. 26(5), 1474–1483 (1997)
53. Song, F., Yun, A.: Quantum security of NMAC and related constructions - PRF domain extension against quantum attacks. In: CRYPTO (2). LNCS, vol. 10402, pp. 283–309. Springer (2017)
54. Zalka, C.: Grover's quantum searching algorithm is optimal. Physical Review A 60(4), 2746 (1999)

# Appendix

## A Extended Quantum Search

We use the following recursive definition of algorithms that combine quantum searches.

**Definition 2.** *We define the class of "extended quantum search algorithms" (*ExtQSearch*), recursively, as follows:*

- *a Toffoli gate;*
- *a quantum circuit formed from a sequence of* ExtQSearch *algorithms;*
- *a quantum circuit* $\mathsf{QSearch}(\mathcal{A})$ *where* $\mathcal{A}$ *is in the class* ExtQSearch*, and the test function of the quantum search is trivial (e.g.,* $\mathcal{A}$ *outputs a boolean flag indicating if its result is correct).*

Similar definitions can be found in the literature, e.g., [15]. We use the fact that Toffoli gates are universal for reversible computing; any quantum circuit made only of Toffoli gates can be translated into a classical algorithm with the same time and memory complexities.

**Lemma 4 (Classical-quantum search correspondence).** *Let* $\mathcal{A}$ *be a quantum algorithm of the class* ExtQSearch*. Assume that it uses a memory (counted in number of qubits)* $\mathcal{M}$ *and a time (in quantum gates)* $\mathcal{T}$*. Then there exists a probabilistic classical algorithm* $\mathcal{A}'$ *that emulates* $\mathcal{A}$*, i.e., returns the same results as if one measured the output of* $\mathcal{A}$*. This algorithm uses on average less than* $\mathcal{T}^2$ *logic gates and uses a memory of* $\mathcal{M}$ *bits.*

*Proof.* In this proof, we will assume that *quantum searches are exact*, meaning that after the number of iterations required by Theorem 1, the algorithm creates a uniform superposition of "good" outputs. This is not always the case, but it can only strengthen our result.

We prove the correspondence by writing down the algorithm $\mathcal{A}'$.

1. Assume that $\mathcal{A}$ is a Toffoli gate or a sequence of Toffoli gates, then this quantum circuit is actually a classical reversible circuit. The result follows trivially.

2. Assume that $\mathcal{A}$ is a sequence of calls to $m$ algorithms of the class QSearch, denoted $\mathcal{A}_1, \ldots, \mathcal{A}_m$, of quantum time complexities $\mathcal{T}_1, \ldots, \mathcal{T}_m$ and maximal memory complexity $\mathcal{M}$. Assume that the correspondence holds for all the $\mathcal{A}_i$. Then there exist classical algorithms $\mathcal{A}'_i$ returning the same results in time $\mathcal{T}_1^2, \ldots, \mathcal{T}_m^2$ and with the same memory $\mathcal{M}$. The classical circuit for $\mathcal{A}'$ has the same layout as the quantum one, in which we replace all the $\mathcal{A}_i$ by the $\mathcal{A}'_i$. It can be easily seen that $\mathcal{A}'$ outputs the same distribution as $\mathcal{A}$. It has an average time complexity:

$$\sum \mathcal{T}_i^2 \leq \left(\sum \mathcal{T}_i\right)^2 ,$$

and uses the same memory as $\mathcal{A}$, since the $\mathcal{A}'_i$ have the same memories, and the circuit layout is the same.

3. Finally, assume that $\mathcal{A}$ is a quantum search: $\mathsf{QSearch}(\mathcal{B})$ where $\mathcal{B}$ is in the class $\mathsf{ExtQSearch}$ and allows a trivial test. Let $T_B$ be the quantum time complexity of $\mathcal{B}$, $p$ its success probability. By Theorem 1, the time complexity $T_A$ of $\mathcal{A}$ is greater than: $\left\lfloor \frac{\pi}{4} \frac{1}{\arcsin \sqrt{p}} \right\rfloor (2T_B)$.

For $x \in [0;1]$, we have $\arcsin x \geq x$, which implies $\frac{1}{\arcsin \sqrt{x}} \leq \frac{1}{\sqrt{x}}$. However $\frac{\pi}{2}\sqrt{x} \geq \arcsin \sqrt{x}$. Thus we can deduce that $T_A \geq \frac{1}{\sqrt{p}} T_B \implies T_A^2 \geq \frac{1}{p} T_B^2$. In the classical algorithm, we merely run $\mathcal{B}'$ until a good output is found. The average complexity is thus:

$$T_{A'} = \frac{1}{p} T_{B'} \leq \frac{1}{p} T_B^2 \leq T_A^2 \ ,$$

which proves the result.

In particular for key-recovery attacks (the main focus of our paper), we can deduce the following corollary:

> If a cipher admits a quantum key-recovery attack (faster than Grover's algorithm) in the class $\mathsf{ExtQSearch}$, then it also admits a classical key-recovery attack.

Indeed, the classical procedure given by Lemma 4 will have a time complexity below classical exhaustive search of the key. In other words, attacks based on quantum search fully comply with the paradigm of "doubling the secret sizes". It should be noted that this holds regardless of the query setting (Q1 or Q2).

This does not mean that these attacks are not interesting, since it is good to know if a quantum adversary can leverage the existing classical weaknesses of a cryptosystem. But it means that quantum search alone *cannot introduce new breaks* regarding key-recovery.

*Remark 6.* We stress that this discussion concerns only key-recovery attacks. There are other quantum generic attacks that *do not* offer a quadratic speedup, and in that case, procedures based on quantum search can effectively reduce the security margins classically established. This is the case in hash function cryptanalysis [32,23,33,17].