Quantum impossible differential attacks: Applications to AES and SKINNY

Nicolas David[1], María Naya-Plasencia[1], and André Schrottenloher[2]

Inria, France
firstname.lastname@inria.fr
Cryptology Group, CWI, Amsterdam, The Netherlands
firstname.lastname@m4x.org

Abstract. In this paper we propose the first efficient quantum version of key-recovery attacks based on impossible differentials, which was left as an open problem in previous work. These attacks work in two phases: first, a number of differential pairs are collected, and second: these pairs are filtered with respect to partial key candidates. In particular, we show how to translate the pair filtering step into a quantum procedure. We provide two applications on SKINNY-128-256 and AES-192/256. These results do not threaten the security of these ciphers but allow us to better understand the post-quantum security margin of these primitives.

Keywords: quantum cryptanalysis · impossible differential attacks · block ciphers

1 Introduction

During the last few years, the interest of the community in understanding the resistance of symmetric primitives to quantum adversaries has considerably increased. Some authors have proposed quantized versions of classical attacks like [20,6] as well as new quantum dedicated attacks [3,5,19,4].

In [7] the authors performed a quantum security analysis of AES. Though none of the proposed quantum attacks reach more rounds than the classical ones, this is because they are compared to an exhaustive search of the key using Grover's algorithm [18], which provides a new generic bound in the quantum setting. In a post-quantum world, the security might well be determined with respect to this new bound, and these quantum attacks define the post-quantum *security margin* of the primitives studied. This security margin needs to be studied with the same care as it was classically.

In [7] the authors studied generic ways for quantizing Square attacks [13,14,17] and DS Meet-in-the-middle attacks [15,16] on AES-256, which gave the best known attacks compared to Grover's algorithm. Though classically, *impossible differential attacks* also provide some trade-offs and comparable complexity, the authors of [7] mention that they did not find a proper way to quantize them, nor a "significant speed-up".

Impossible differential attacks, introduced simultaneously by Knudsen [21] and Biham, Biryukov and Shamir [2], exploit a differential transition that cannot occur to build a distinguisher or to extract information on the secret key of a cipher. Since [7], there has been no further study of quantum impossible differential attacks, except a proposal [27,26] to use quantum algorithms to efficiently find impossible paths (but no actual speedup of the attack).

This paper. The results presented in this paper improve our knowledge in several directions.

1. We propose the first efficient quantum impossible differential attacks with a competitive speed up regarding classical attacks. An impossible differential key-recovery attack runs in two phases: first, given black-box encryption and decryption access, we build a set of pairs with some truncated input-output difference pattern. Second, partial key candidates are sieved, by removing those which, on some of the given pairs, would make the impossible differential appear. Our main contribution is an efficient quantum algorithm for this *pair filtering step*.
2. We give some results on the applications of these attacks to the popular block ciphers AES and SKINNY, summarized in Table 1 and compared to the best existing post-quantum attacks (by this we imply attacks that are better than Grover's exhaustive key search). We also fill in the gap from [7] by proposing the first quantum impossible differential attacks on AES-192/256.

Organization. We start in Section 2 by introducing (classical) impossible differential attacks. Section 3 recalls the algorithms that we can use for generating the pairs while Section 4 describes the process of pair filtering.

2 Classical Impossible Differential Attacks

In this section, we provide a generic depiction of classical impossible attacks, that will be helpful for translating them into quantum algorithms. We give a generic formula for their complexity which is from [9].

2.1 Principle

The goal of this cryptanalysis technique is to recover some bits of the secret key K of a black-box encryption oracle. This is done by discarding all the wrong key guesses, with the help of a pair of plaintexts that leads to an impossible pattern under its partial encryption with the wrong key guesses.

Let E be an n-bit block cipher with r rounds. We write $E = E_{\text{out}} \circ E_{\text{imp}} \circ E_{\text{in}}$, as on Figure 1, where $E_{\text{out}}, E_{\text{imp}}$ and E_{in} have $r_{\text{out}}, r_{\text{imp}}$ and r_{in} rounds respectively ($r = r_{\text{in}} + r_{\text{imp}} + r_{\text{out}}$).

An impossible differential attack is based on an impossible differential of maximal length, that is, a pair of differentials Δ_X, Δ_Y such that the probability that Δ_X propagates to Δ_Y after r_{imp} rounds in 0. We will then append r_{in} and

Table 1. Summary of best post-quantum attacks on SKINNY-128-256 and AES-192/256 (with lower complexities than Grover's search). Symbol * means we have extrapolated the complexity on 21 rounds from the original attack on 24 rounds for comparison, though it is still too expensive post-quantumly (the best classical one is the 24-round attack). For SKINNY-128-256 our results clearly provide the best quantum attack and therefore the security margin. For AES-256, we obtain a better memory than [16], and a time complexity comparable to [7]. Our best attack is dominated by the cost of generating the pairs. ** indicates that the memory considered is quantum memory with quantum random-access (otherwise, this is classical memory).

Algorithm	# rounds	Ref.	Time	Memory	Data	Setting
SKINNY-128-256	21	This paper	$2^{112.7}$	$(2^{103.17})^{**}$	$2^{112.5}$	Q_2
SKINNY-128-256	21	[25]*	$2^{167.17}$	$2^{103.17}$	2^{128}	Classical
SKINNY-128-256	20	[25]*	$2^{126.46}$	$2^{54.6}$	$2^{126.46}$	Classical
AES-192	7	[7]-Grover	$2^{105.6}$	neg	neg	Q1
AES-256	7	[7]-Grover	$2^{137.3}$	neg	neg	Q1
AES-256	7	[7]-square	2^{121}	2^{38}	2^{37}	Q1
AES-192	7	[7]-square	$2^{103.4}$	$2^{38} + (2^{27})^{**}$	2^{37}	Q1
AES-256	7	[7]-square	2^{107}	$2^{38} + (2^{27})^{**}$	2^{37}	Q1
AES-256	7	[16]	$2^{99} + 2^{98}$	2^{96}	2^{99}	Classical
AES-192/256	7	This paper	$2^{101.5}$	$(2^{78.5})^{**}$	$2^{99.8}$	Q2
AES-192/256	7	This paper	$2^{99.8} + 2^{95.2}$	$(2^{78.5})^{**}$	$2^{99.8}$	Q2
AES-256	8	[7]-Grover	2^{138}	neg	neg	Q1
AES-256	8	[7]-DS-MITM	2^{136}	2^{88}	2^{88}	Q1

r_{out} rounds of the cipher respectively before and after the impossible differential. We name *impossible pattern* the tuple of quantities $(\Delta_X, \Delta_Y, r_{\text{imp}}, r_{\text{in}}, r_{\text{out}})$.

Next, we define two sets of differences D_{in} and D_{out} such that Δ_X maps backwards to D_{in} through E_{in}, and Δ_Y maps forwards to D_{out} through E_{out}. If we are given a pair of plaintexts with difference in D_{in}, such that the output difference falls in D_{out}, then due to the impossible pattern, we can discard any key that satisfies:

$$\left(E_{\text{in}}(k)(x) \oplus E_{\text{in}}(k)(y) = \Delta_X\right) \wedge \left(E_{\text{out}}^{-1}(k)(E(x)) \oplus E_{\text{out}}^{-1}(k)(E(y)) = \Delta_Y\right) \quad . \tag{1}$$

The goal of the attack is to discard as many keys as possible using many plaintext-ciphertext pairs.

In this paper, we adopt a representation inspired from [9]. The attack is a two-steps procedure:

Pair Generation: In this part, we focus on solving the following problem:

$$\begin{array}{rl} \text{Data:} & N \in \mathbb{N},\ E : \{0,1\}^n \to \{0,1\}^n \\ & D_{\text{in}} \subset \{0,1\}^n\ ,\ D_{\text{out}} \subset \{0,1\}^n \\ \text{Question:} & \text{Find } N \text{ pairs } (x,y) \in \{0,1\}^{2n} \text{ such that} \\ & x \oplus y \in D_{\text{in}} \text{ and } E(x) \oplus E(y) \in D_{\text{out}} \end{array}$$

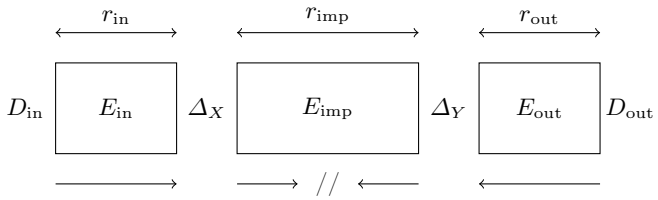

Fig. 1. Impossible differential attack, with the notations used in this paper. The differential $\Delta_X \leftrightarrow \Delta_Y$ through the middle rounds E_{imp} is impossible.

This is a *limited-birthday* problem. A good analysis of it is given in [9], with an efficient algorithm for solving it. At the end of this step, we will obtain a set of N differential pairs. We will denote by $\mathcal{T}_0$ the table of such pairs obtained.

Data Complexity. A probabilistic analysis allows us to estimate the value of N and thus the data complexity. To filter out all the wrong subkeys, we should have $N = \mathcal{O}(2^{c_{\text{in}}+c_{\text{out}}} \cdot \mid K_{\text{in}} \cup K_{\text{out}} \mid)$ where c_{in} (resp. c_{out}) denotes the number of bit conditions required for a pair of plaintext (resp. ciphertext) to propagate to the central part.

Time Complexity. We let Δ_{in} and Δ_{out} be such that $|D_{\text{in}}| = 2^{\Delta_{\text{in}}}$ and $|D_{\text{out}}| = 2^{\Delta_{\text{out}}}$. The complexity of the Pair Generation problem was studied in [9],

$$C_N = \max\Big(\min_{\Delta \in \{\Delta_{\text{in}}, \Delta_{\text{out}}\}} \sqrt{N 2^{n+1-\Delta}}, N 2^{n+1-(\Delta_{\text{in}}+\Delta_{\text{out}})}\Big) \ .$$

Pair Filtering: In this step, we assume given the table $\mathcal{T}_0$ of size N computed above. The goal of this step is to split the set of subkeys $K_{\text{in}} \cup K_{\text{out}}$ into two sets: one that contains all the subkeys that have been invalidated by some pair of $\mathcal{T}_0$ and another that contains the other subkeys. A very interesting optimization in this step is called the *early abort* technique. It was introduced in [22] and is described in detail in [8].

To explain how the filtering is done, we will introduce *test functions*. Let us assume that $K_{\text{in}} \cup K_{\text{out}}$ can be decomposed as: $K_{\text{in}} \cup K_{\text{out}} = K_1 \times K_2 \times \cdots \times K_\ell$, where $K_1, \ldots, K_\ell$ typically represent some bytes or bits of the subkeyspace. Together with this decomposition, we will have ℓ test functions:

$$T_i \ : \mathcal{T}_0 \times K_1 \times \ldots \times K_i \to \{0,1\} \ .$$

This corresponds to taking some part of the subkey, some part of the pair, and checking whether they meet some condition. Typically, we start from the differences D_{in} and D_{out} and compute partially the first and last rounds; the successive T_i check that the partial encryption and decryption of the pair satisfies a truncated differential pattern that ultimately leads to the impossible differential (Δ_X, Δ_Y) at rounds r_{in} and r_{out}.

Next, we define the set of pairs satisfying all the T_i:

$$\mathcal{T}_\ell(k) = \{p \in \mathcal{T}_0 \mid \forall i, T_i(p, k_1, \ldots, k_i) = 1\} \ . \tag{2}$$

Thus, the test functions are defined so that: $\mathcal{T}_\ell(k) \neq \emptyset$ if and only if, there exist a given pair $p \in \mathcal{T}_0$ such that k makes the impossible differential appear for p. The computation of $\mathcal{T}_\ell(k)$ thus yields a probabilistic procedure that discards a wrong subkey with some probability. We define the sequence σ_i so that the sizes of the tables $\mathcal{T}_0, \ldots, \mathcal{T}_{\ell-1}$ are $N, \sigma_1 N, \ldots, (\prod_{i=1}^{\ell-1} \sigma_i)N$, therefore $0 < \sigma_i < 1$. The early abort technique consists in building sequentially the $\mathcal{T}_i(k_1, \cdots, k_i)$ from $\mathcal{T}_{i-1}(k_1, \cdots, k_{i-1})$ and guessing k_i. Its complexity is:

$$\begin{aligned} &|K_1| \left(\underbrace{N}_{\text{Build } \mathcal{T}_1} + |K_2| \left(\sigma_1 N + |K_3| \left(\sigma_1\sigma_2 N + \ldots + |K_\ell| \left(\prod_{i=1}^{\ell-1} \sigma_i \right) N \right) \right) \right) \\ &\quad = N \left(|K_1| + \sigma_1 |K_1||K_2| + \sigma_1\sigma_2 |K_1||K_2||K_3| + \ldots + \prod_{i=1}^{\ell-1} \sigma_i \prod_{i=1}^{\ell} |K_i| \right) \end{aligned} \tag{3}$$

Details can be found in [9].

3 Pair Generation: Quantum Limited Birthday Problem

For the pair generation problem, we use the formula given in [20] for the case $N = 1$ (when there is a single pair to be found):

$$Q_1 = \mathcal{O}\left(\max\left(2^{(n-\Delta_{out})/3}, 2^{(n-\Delta_{out})/2-\Delta_{in}/3}\right) \right) \ . \tag{4}$$

We will then pay $Q_N = N \cdot Q_1$ to recover N pairs. Depending on the parameters, it may be more advantageous to resort to classical structures.

If both Δ_{out} and Δ_{in} are large, we can use another approach based on BHT collision search [11], which gives an alternative complexity: $\mathcal{O}\left(N 2^{\frac{2(n-\Delta_{in}-\Delta_{out})}{3}}\right)$.

4 Quantum Pair Filtering

In this section, we design a quantum version of the early-abort algorithm and study its time complexity.

4.1 Preliminaries

We refer to [24] for a broad introduction to quantum computing and the quantum circuit model. We will use below the ket notation of quantum states $|\cdot\rangle$. When studying a cipher E, our unit of computation will be an evaluation of E or of E^{-1}, either as a classical, or a quantum circuit. In general we use the qRAM model, in which all qubits of a quantum circuit can be accessed in superposition with cost 1.

Quantum Search. We use *quantum search* to refer to Amplitude Amplification [10], which generalizes Grover's algorithm [18].

Theorem 1 (Theorem 2 in [10]). *Let $\mathcal{A}$ be a quantum algorithm with no input and without measurements. Assume that $\mathcal{A}$'s outputs have a probability a of being "good" (and that they can be easily tested). Assume that there exists a quantum circuit for $\mathcal{A}$ running in time T_A. Then there exists an algorithm that, with no input, produces a good output. It runs in time:* $2\left\lfloor \frac{\pi}{4}\frac{1}{\arcsin\sqrt{a}}\right\rfloor T_A \leq \frac{\pi}{2}\frac{T_A}{\sqrt{a}}$ *and succeeds with probability* $\max(a, 1-a)$.

In practice, a is small and we do not lose much by upper bounding $\frac{1}{\arcsin\sqrt{a}} \leq \frac{1}{\sqrt{a}}$. Our main use of Amplitude Amplification in this paper is its *Exact* version (Theorem 4 in [10]). If a is known exactly, then the probability of failure of the procedure can be brought down to 0. The technique consists only in performing a final partial iteration (also used in [12]), so the total time complexity can be upper bounded by: $\left(\frac{\pi}{2\sqrt{a}}+2\right)T_A \leq \frac{\pi}{2}\left(\frac{1}{\sqrt{a}}+2\right)T_A$. Below we omit the $+2$ factor (which remains negligible) to simplify the writing, but we note that the obtained upper bounds are exact, and not asymptotic.

4.2 Assumptions on the Attack

The classical early abort enumerates all the key guesses for which there exists no invalidating pair, using ℓ nested loops. The quantum version of this algorithm uses ℓ Exact Amplitude Amplification subroutines. In order to ensure its correctness, we first need to make some *classical* assumptions on the Impossible Differential pattern and the initial set of pairs $\mathcal{T}_0$.

Assumption 1 *Given the initial table $\mathcal{T}_0$, there exists a single key $(k_1, \ldots, k_\ell)$ such that $\mathcal{T}_\ell(k_1, \ldots, k_\ell) = \emptyset$.*

Assumption 2 *No intermediate table exceeds twice its expected size:*

$$\forall k_1, \ldots, k_i, |\mathcal{T}_i(k_1, \ldots, k_i)| = N(k_1, \ldots, k_i) \leq \left(\prod_{j=1}^{i} \sigma_j\right) N \ .$$

4.3 Filtering of a Table

We introduce additional notations to make precise time and memory complexity estimates. Since we want to count the time complexity relatively to a cipher evaluation, we introduce: • t_i the time to evaluate the condition T_i; • t the time to perform a $4n$-bit register operation such as: swapping or copying a register that contains a pair. We count the memory complexity in number of pairs (a pair can be stored on a $4n$-bit register). We denote by $M_i = 2\,\mathbb{E}(N_i) = 2\left(\prod_{j\leq i}\sigma_j\right)N_0$ the maximal size of all intermediate tables $\mathcal{T}_i$.

First of all, we compute the time to filter an intermediate table.

Lemma 1. *For all i, there exists a quantum circuit F_i that maps:*

$$|\mathcal{T}_i(k_1,\ldots,k_i)\rangle\,|k_{i+1}\rangle\,|0\rangle \mapsto |\mathcal{T}_i(k_1,\ldots,k_i)\rangle\,|k_{i+1}\rangle\,|\mathcal{T}_{i+1}(k_1,\ldots,k_i,k_{i+1})\rangle\,|*\rangle \quad ,$$

where $*$ are computation qubits that depend only on $k_1,\ldots,k_i,k_{i+1}$. The time complexity, relative to a cipher evaluation, is bounded by:

$$\begin{cases} M_i(t_{i+1}+t) \text{ if qRAM is allowed} \\ M_i(t_{i+1}+\frac{(\log_2 M_i)^2}{4}t) \text{ otherwise} \end{cases} \tag{5}$$

and the memory complexity by M_i or $M_i\frac{(\log_2 M_i)^2}{4}$ (mainly due to the $*$ state).

Here we are simply computing the filtering function for all elements of the table. It should be noted that the $|*\rangle$ is here to ensure reversibility of these operations. We could perform uncomputations to erase it immediately, but we prefer to wait until the table $\mathcal{T}_{i+1}(k_1,\ldots,k_i,k_{i+1})$ is not needed anymore. Then we will erase not only the table, but also all of the computations that led to it.

4.4 Exact Pair Filtering

Lemma 2. *Let $1 \le i \le \ell$. Let t_i be the time (in quantum operations) to compute the condition T_i.*

There exists a quantum circuit (unitary) U_i that, on an input state of the form

$$|k_1,\ldots,k_i\rangle\,|\mathcal{T}_{i-1}(k_1,\ldots,k_{i-1})\rangle \quad ,$$

flips the phase (multiplies it by -1) iff there exists a completion $k_{i+1},\ldots,k_\ell$ such that $k_1,\ldots,k_\ell$ is the good key. It runs in time:

$$2\sum_{j=i}^{\ell} M_{j-1}(t_j+t)\left(\frac{\pi}{2}\right)^{j-i}\sqrt{\prod_{m=i}^{j}|K_m|} \quad , \text{ using a memory } M_{i-1}.$$

This is proven recursively, starting from the final unitary U_ℓ, up to U_1. Each time, we add a new level of quantum search. We use Exact Amplitude Amplification all the time, because we are ensured (by our assumptions) that exactly a single key must survive all the filters (thus there is no need for handling errors, which would usually happen with quantum searches). Then, using U_1 in a quantum search for k_1, where we start from the table $\mathcal{T}_0$, we obtain our complete pair filtering algorithm.

Corollary 1. *Under our assumptions, there exists a quantum pair filtering algorithm that finds the single good $k_1,\ldots,k_\ell$ in time:*

$$2N\sum_{j=1}^{\ell}(t_j+t)\left(\prod_{m=1}^{j-1}\sigma_m\right)\left(\frac{\pi}{2}\right)^{j}\sqrt{\prod_{m=1}^{j}|K_m|} \quad ,$$

and using little more than N memory.

A more in depth approach on the quantum memory follows.Since the memory required at step i is M_i, the total memory will be

$$\sum_i M_i = \sum_i 2\left(\prod_{j\leq i}\sigma_j\right)N_0 \leq 2\sum_i \max(\sigma)^i N_0 \leq 2\frac{1-\max\sigma^{l+1}}{1-\max\sigma}N \ .$$

If we focus on N, the σ_i and the $|K_i|$, we can simplify this formula as follows:

$$N\left(\sqrt{|K_1|}+\sigma_1\sqrt{|K_1||K_2|}+\sigma_1\sigma_2\sqrt{|K_1||K_2||K_3|}+\ldots+\prod_{i=1}^{\ell-1}\sigma_i\sqrt{\prod_{i=1}^{\ell}|K_i|}\right)$$
$$=|K_1|^{\frac{1}{2}}\left(N+|K_2|^{\frac{1}{2}}\left(\sigma_1 N+|K_3|^{\frac{1}{2}}\left(\sigma_1\sigma_2 N+\ldots+|K_\ell|^{\frac{1}{2}}\left(\prod_{i=1}^{\ell-1}\sigma_i\right)N\right)\right)\right) \quad (6)$$

Compared with the classical formula (Equation 3), it can be seen that we have been able to put a square root on each $|K_i|$. In some cases, it can also be advantageous to use quantum search when constructing the table (this accelerates the filtering).

5 Applications

SKINNY. SKINNY-128-256 is an SPN tweakable block cipher inspired by the AES and introduced in [1]. Skinny relies on the *tweakey* framework, so the 128-bit tweak and the 128-bit key are glued together in a 256-bit tweakey. The goal of our cryptanalysis is to recover this 256-bit string.

Classical Attack. We obtained a classical impossible differential attack of Skinny that achieves 21 rounds based on work from [25]. Since the goal is to recover the full 256-bit tweakey, the generic bound is 2^{256}. With our notation, the attack is parameterized by $\Delta_{\text{in}} = 32$, $\Delta_{\text{out}} = 72$, $c_{\text{in}} = 24$, $c_{\text{out}} = 72, N = 2^{103.17}$. N is chosen so that a single subkey survives the filtering. The filtering procedure itself has 12 successive steps of partial sub(twea)key guesses, with a complexity $2^{167.17}$.

Quantum Attack. Since 2 plaintext-ciphertext pairs are required to discriminate the right tweakey, the exhaustive search with Grover's algorithm has a complexity of $2^{129.65}\cdot t_{Enc}$ where t_{Enc} is the time required to perform an encryption. Our quantum attack will be valid if we manage to outperform this.

Generation. By Equation 4, with the parameters $\Delta_{in} = 32$, $\Delta_{out} = 72$, $c_{in} = 24$, $c_{out} = 72, N = 2^{103.17}$, we need $Q_N = 2^{112.5}$ encryptions to generate the pairs (in the classical setting $C_N = 2^{128}$ encryptions).

Filtering. The quantum key recovery follows the same steps as the classical one, and we use Corollary 1 to determine its complexity. Corollary 1 also features a $(t+t_i)$ term that carries the complexity of $4n$-bit operations and the complexity of performing the test T_i. One can assert $(t+t_i) < t_{Enc}$, we then obtain the numbers reported in Table 2.

Table 2. Time complexity of the different steps for 21-round SKINNY-128-256.

Complexity	Quantum time	Quantum mem.	Classical time	Classical mem.
Pair Generation	$2^{112.5}$	2^{24}	2^{128}	1
Pair Filtering	$2^{110.5}$	$2^{103.17}$	$2^{167.17}$	$2^{103.17}$
Total	$2^{112.75}$	$2^{103.17}$	$2^{167.17}$	$2^{103.17}$
Generic	$2^{129.65}$	1	2^{256}	1

AES. Our attack on 7-round AES is a quantum version of the attack of [23]. It is built by appending two rounds before and one round after a 4-round impossible differential. The parameters are $N = 2^{78.5}$, $\Delta_{in} = 64$, $\Delta_{out} = 32$. The pair generation step requires quantum time $2^{99.8}$. In the pair filtering step, we filter with respect to 4 subkey spaces of 32 bits each. The filtering probabilities are $\sigma_1 = 2^{-16}$, $\sigma_2 = 2^{-16}$, $\sigma_3 = 2^{-24}$. Using Corollary 1 we would obtain a complexity $2^{101.5}$, comparable to previous works. However we improve this to $2^{95.2}$ using an improved pair filtering algorithm. Indeed, while Lemma 1 essentially applies the test function on all pairs in the current table, it is possible to construct the next table faster using Grover searches.

References

1. Beierle, C., Jean, J., Kölbl, S., Leander, G., Moradi, A., Peyrin, T., Sasaki, Y., Sasdrich, P., Sim, S.M.: The SKINNY family of block ciphers and its low-latency variant MANTIS. In: CRYPTO (2). Lecture Notes in Computer Science, vol. 9815, pp. 123–153. Springer (2016)
2. Biham, E., Biryukov, A., Shamir, A.: Cryptanalysis of skipjack reduced to 31 rounds using impossible differentials. J. Cryptol. 18(4), 291–311 (2005)
3. 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)
4. Bonnetain, X., Jaques, S.: Quantum period finding against symmetric primitives in practice. IACR Cryptol. ePrint Arch. 2020, 1418 (2020)
5. Bonnetain, X., Naya-Plasencia, M.: Hidden shift quantum cryptanalysis and implications. In: ASIACRYPT (1). Lecture Notes in Computer Science, vol. 11272, pp. 560–592. Springer (2018)
6. 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)
7. Bonnetain, X., Naya-Plasencia, M., Schrottenloher, A.: Quantum security analysis of AES. IACR Trans. Symmetric Cryptol. 2019(2), 55–93 (2019)
8. Boura, C., Lallemand, V., Naya-Plasencia, M., Suder, V.: Making the impossible possible. J. Cryptol. 31(1), 101–133 (2018)
9. Boura, C., Naya-Plasencia, M., Suder, V.: Scrutinizing and improving impossible differential attacks: Applications to clefia, camellia, lblock and simon. In: ASIACRYPT (1). Lecture Notes in Computer Science, vol. 8873, pp. 179–199. Springer (2014)

10. Brassard, G., Hoyer, P., Mosca, M., Tapp, A.: Quantum amplitude amplification and estimation. Contemporary Mathematics 305, 53–74 (2002)
11. Brassard, G., Høyer, P., Tapp, A.: Quantum cryptanalysis of hash and claw-free functions. In: LATIN. Lecture Notes in Computer Science, vol. 1380, pp. 163–169. Springer (1998)
12. Chi, D.P., Kim, J.: Quantum database search by a single query. In: Selected papers from the First NASA International Conference on Quantum Computing and Quantum Communications. pp. 148–151 (1998)
13. Daemen, J., Knudsen, L.R., Rijmen, V.: The block cipher square. In: FSE. Lecture Notes in Computer Science, vol. 1267, pp. 149–165. Springer (1997)
14. Daemen, J., Rijmen, V.: AES proposal: Rijndael. Submission to NIST AES competition (1999)
15. Demirci, H., Selçuk, A.A.: A meet-in-the-middle attack on 8-round AES. In: FSE. Lecture Notes in Computer Science, vol. 5086, pp. 116–126. Springer (2008)
16. Derbez, P., Fouque, P., Jean, J.: Improved key recovery attacks on reduced-round AES in the single-key setting. In: EUROCRYPT. Lecture Notes in Computer Science, vol. 7881, pp. 371–387. Springer (2013)
17. Ferguson, N., Kelsey, J., Lucks, S., Schneier, B., Stay, M., Wagner, D.A., Whiting, D.: Improved cryptanalysis of rijndael. In: FSE. Lecture Notes in Computer Science, vol. 1978, pp. 213–230. Springer (2000)
18. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: STOC. pp. 212–219. ACM (1996)
19. 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)
20. Kaplan, M., Leurent, G., Leverrier, A., Naya-Plasencia, M.: Quantum differential and linear cryptanalysis. IACR Trans. Symmetric Cryptol. 2016(1), 71–94 (2016)
21. Knudsen, L.: Deal-a 128-bit block cipher. complexity 258(2), 216 (1998)
22. Lu, J., Kim, J., Keller, N., Dunkelman, O.: Improving the efficiency of impossible differential cryptanalysis of reduced camellia and MISTY1. In: CT-RSA. Lecture Notes in Computer Science, vol. 4964, pp. 370–386. Springer (2008)
23. Mala, H., Dakhilalian, M., Rijmen, V., Modarres-Hashemi, M.: Improved impossible differential cryptanalysis of 7-round AES-128. In: INDOCRYPT. Lecture Notes in Computer Science, vol. 6498, pp. 282–291. Springer (2010)
24. Nielsen, M.A., Chuang, I.: Quantum computation and quantum information (2002)
25. Sadeghi, S., Mohammadi, T., Bagheri, N.: Cryptanalysis of reduced round SKINNY block cipher. IACR Trans. Symmetric Cryptol. 2018(3), 124–162 (2018)
26. Xie, H., Yang, L.: Quantum impossible differential and truncated differential cryptanalysis. CoRR abs/1712.06997 (2017)
27. Xie, H., Yang, L.: Using bernstein-vazirani algorithm to attack block ciphers. Des. Codes Cryptogr. 87(5), 1161–1182 (2019)