A Quantum Circuit to Execute a Key-Recovery Attack Against the DES and 3DES Block Ciphers

Simone Perriello, Alessandro Barenghi, Gerardo Pelosi

Department of Electronics, Information and Bioengineering - DEIB
Politecnico di Milano, 20133 Milano, Italy
Email: simone.perriello@polimi.it, alessandro.barenghi@polimi.it, gerardo.pelosi@polimi.it

***Abstract*—Quantum computing enabled cryptanalytic techniques are able to concretely reduce the security margin of existing cryptographic primitives. While this reduction is only polynomial for symmetric cryptosystems, it still provides a reduction in their security margin.**

In this work, we propose a detailed quantum circuit designed to cryptanalyze both the Data Encryption Standard (DES) cryptosystem, and its successor Triple-DES (3DES), currently standardized in ISO/IEC 18033-3, and still widely employed in satellite data and bank card encryption. To do so, we introduce the first quantum circuit implementation of the 8 substitution tables (a.k.a. S-boxes), applying a bitslicing strategy, which is currently the most efficient classical combinatorial circuit design in terms of number of two inputs Boolean gates. Secondly, we present the complete quantum circuits required to attack both DES and 3DES leveraging Grover's algorithm. We provide finite regime, closed form equations, delineating the circuits complexities in terms of the number of qubits, gates, depth and number of qubits multiplied by depth. The complexity analysis is based on two distinct gate sets: a NOT-CNOT-Toffoli (NCT) extended with the Hadamard gate; and the fault-tolerant Clifford+T. Finally, akin to the classical attack to the 3DES, we introduce a meet-in-the-middle strategy relying on an exponential amount of Quantum Random Access Memory. Our findings show that the 3DES with keying option 2, the most widely employed variant of 3DES, can be attacked with a circuit depth of approximately 2^{67} and less than a thousand qubits. This is close to the 2^{64} value suggested by NIST for the depth achievable sequentially by a single quantum computer in a decade. Our technique can be further sped up parallelizing the approach onto multiple devices, pointing to the practicality of cryptanalyzing 3DES in such a scenario.

***Index Terms*—Block-ciphers, Grover, DES, 3DES, quantum cryptanalysis**

I. Introduction

Quantum computing has emerged as a significant disruptor in the field of cryptography, posing novel challenges to the security of cryptographic algorithms. Shor's proposal [1] in 1994 of a polynomial-time quantum algorithm marked a significant advancement in *quantum cryptanalysis*, particularly impacting modern public-key cryptography schemes such as RSA, ECDSA, and ECDH. Conversely, symmetric cryptography has been long assumed to be essentially immune to quantum computing, with the only concern being the quadratic speedup offered by the adaptation of Grover's algorithm to key-recovery attacks. To break a symmetric cipher, an attacker possessing pairs of plaintexts and corresponding ciphertexts encrypted with a given length-n key — a scenario classically denoted as *Known Plaintext Attack (KPA)* model — can embed the publicly available encryption routine as Grover's *oracle*. By creating a superposition of all the 2^n possible keys, the attack retrieves the correct key with an asymptotic computational time complexity of $\mathcal{O}(\sqrt{2^n})$, that is, quadratically faster with respect to a brute-force attack performed using a classical computer. While doubling the cipher's key size is a common countermeasure to achieve post-quantum security, this approach overlooks the cost of the encryption routine in terms of quantum gates. The adaptation is even more challenging when considering that symmetric ciphers rely on non-linear components — called substitution tables, substitution boxes or *S-boxes* for short — making their quantum circuit implementation non-trivial. Ensuring robust post-quantum cryptography requires a more nuanced evaluation of a cipher's resistance to quantum threats, allowing for a precise assessment of the practical feasibility of quantum-accelerated attacks[1].

In the recent literature, many works have been produced trying to accurately assess the complexity of an attack to different symmetric ciphers based on quantum computers. While the majority of the works focuses on the quantum cryptanalysis of AES [2]–[4], fostered by NIST standardization calls for post-quantum asymmetric schemes [5] and digital signatures [6] using the quantum computational complexity of AES as the reference bar to assess the security of all proposals, quantum attacks to other block ciphers are also being investigated [7]–[10]. The ongoing effort to assess the security of symmetric ciphers lead to the discovery of several vulnerabilities, relying however on an attack model in which both the encryption routine and the key are implemented in the classical circuit [11]–[17]. Such kind of model, referred to as Q2 model as opposed to the standard Q1 model previously described, exploits either Simon's algorithm [18] alone or in combination with Grover's algorithm, and offers key-recovery attacks going beyond the standard quadratic speedup. Although of great theoretical interest, however, the Q2 model does not pose a threat to already existing and deployed

[1] *On the Practical cost of Grover for AES Key Recovery*, Sarah D. - NCSC https://csrc.nist.gov/csrc/media/Presentations/2024/practical-cost-of-grover-for-aes-key-recovery/images-media/sarah-practical-cost-grover-pqc2024.pdf

cryptographic schemes, as they are expected to be running on a classical computer.

Motivations. To the best of our knowledge, the post-quantum security of the *Data Encryption Standard (DES)*, whose first design dates back to the 1970s, and the *Triple Data Encrpytion Standard (3DES)* symmetric cipher, created to overcome the security weaknesses of DES while maintaining backward compatibility with it, has not been explored before. Notably, 3DES is currently standardized in the ISO/IEC 18033-3, a suite of symmetric encryption ciphers updated for the last time in 2021.

While significant effort has been put in the quantum cryptanalysis of other symmetric encryption algorithms, DES and 3DES still remain unexplored, with the only exception being two distinct proposals attacking Simplified DES [19], [20], a variant of DES used for educational purposes. The significance of studying DES and 3DES ciphers is testified by their extensive usage across critical systems. As an example, many satellites continue to heavily rely on these standards for data transmission[2]. Furthermore, popular applications like Mozilla Firefox and Mozilla Thunderbird utilize 3DES for encrypting login credentials[3], and the vast majority of modern ATMs still employ either DES or 3DES to encrypt PINs and send them to remote servers for processing. The widespread usage of 3DES in finance is further evidenced by its widespread usage in the electronic payment industry, with card payment standards like *Europay, Mastercard, Visa (EMV)* adopting 3DES for key management[4], justifying their choice with the 3DES inclusion in the ISO/IEC-18033-3 suite of symmetric ciphers. Moreover, AWS recently deployed a payment service allowing 3DES usage[5].

Finally, 3DES continues to be supported by several prominent cryptography software libraries. These include *Bouncy Castle*, a comprehensive collection of cryptographic APIs utilized in Java and C#, including the Android operating system; *Crypto++*, an open-source C++ library; *OpenSSL*, a software library facilitating secure communications over networks; and the *Trusted Platform Module (TPM)* standard, essential for secure cryptoprocessors such as those mandated by Windows 11.

Our contributions. We present the first ever design and implementation of a Grover-based key-recovery attack to both DES and 3DES in the Q1 model. We pose a special focus on the quantum implementation of the 8 S-boxes employed in both ciphers, for which we adapted the *bitslicing* technique to rephrase their non-linear function in terms of the *NOT-CNOT-Toffoli (NCT)* gate set. We report detailed costs for all of them.

[2]The N2YO satellite tracker lists 31 active Iridium satellites launched before 2001 (the year in which AES superseded DES as a NIST standard) at https://www.n2yo.com/satellites/?c=15

[3]The encryption/decryption routines source code is avaiable at https://searchfox.org/mozilla-central/source/security/nss/lib/pk11wrap/pk11sdr.c

[4]The EMV security and key management specifications are available at https://www.emvco.com/specifications/book-2-security-and-key-management

[5]The official AWS tutorial is available at https://docs.aws.amazon.com/payment-cryptography/latest/userguide/getting-started.html

For both attacks, we report the complexity metrics in terms of number of qubits (a.k.a., width), number of gates and depth in terms of two distinct gate sets: the NCT+Hadamard gate set and the Clifford+T gate set, the latter considered to be the most promising one for fault-tolerant quantum computation. Additionally, we report the depth×width metric proposed in [21], considered to be a more realistic characterization of a quantum circuit complexity in a computational model in which each qubit is independently controlled by a classical device. In this respect, the limitations of quantum technologies may come either by the availability of a reduced number of qubits, or their coherence time, and for this reason we consider both a low-depth and a low-width implementation. Moreover, following [22], we also report the depth× number of gates, which is considered more relevant when the computation must be parallelized across multiple quantum devices limited by a fixed value of depth.

Finally, we show an adaption of the classical *meet-in-the-middle (MITM)* strategy to the quantum case. By using a Quantum Random Access Classical Memory (QRACM) [23], we report complexity measures when considering either a $\sqrt[3]{M}$ or a $\sqrt{M}$ access cost to the memory, M being the size of the quantum memory.

Through a rigorous comparison with respect to the state-of-the-art proposals offering a quantum implementation of symmetric block ciphers, we show how both DES and 3DES fail to reach the same security margin of the others, failing below the minimum level demanded by NIST for post-quantum security.

II. Background

A. DES and 3DES block ciphers

DES. The Data Encryption Standard (DES), devised by IBM in the early 1970s and established as a federal standard by the US National Bureau of Standards in 1977, stands as one of the earliest and most exhaustively examined symmetric block ciphers. Operating on a 64-bit block of data and a 64-bit secret key (of which only 56 bits are utilized), DES relies on a Feistel-network structure, involving the iterative application of 16 *rounds*, each consisting of the same sequence of operations, with the primary operation being the *round function* F. The function F takes as input a *round key* K_i, derived from the secret key k through a *key scheduling* algorithm, and a 32-bit portion of the data block, producing as output another 32-bit block. The Feistel structure enables both the encryption routine E_k and the decryption routine D_k to rely on the same sequence of operations, with the only difference being in how the round keys are scheduled.

The encryption process of DES is depicted in Fig. 1. The key scheduling algorithm, illustrated in the rightmost portion of Fig. 1, operates on the input 64-bit key k to generate 16 distinct 48-bit round keys K_i. This operation relies on two fixed permutation tables, known as $\texttt{PC}-1$ and $\texttt{PC}-2$. $\texttt{PC}-1$ selects 56 bits from the 64-bit key k, with the remaining 8 bits either discarded or used as parity-check bits. Subsequently, each of the 16 rounds involves splitting the 56 bits into two 28-bit halves, independently left-rotating each half a number

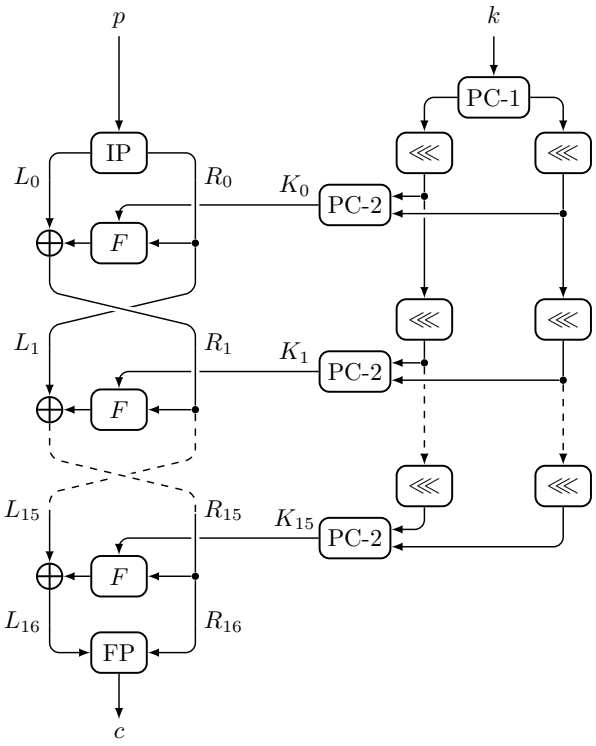

Fig. 1. DES encryption. p is the 64-bit plaintext, i.e., the message to be encrypted; k is the 64-bit encryption key; c is the 64-bit ciphertext.

of times determined by the round number. $\texttt{PC}-2$ then selects 24 bits from each half to compose the 48-bit round key K_i.

Conversely, the leftmost portion of Fig. 1 depicts the sequence of operations applied to the 64-bit plaintext p block to obtain the ciphertext c. Initially, the plaintext undergoes an *Initial Permutation (*IP*)*, rearranging the bits according to a predefined table. The block is then divided into two 32-bit halves, the left portion L_0 and the right portion R_0. Subsequently, 16 rounds of identical operations ensue. Specifically, at the end of each round i, with $0 \leq i < 16$, L_i and R_i are updated according to the following equations:

$$L_{i+1} = L_i \oplus F(K_i, R_i) \qquad R_{i+1} = L_i.$$

It is important to note that unlike previous rounds, at the end of the last round the two halves are not swapped. Finally, the final step of E_k applies the *Final Permutation (*FP*)* to the concatenation of L_{16} and R_{16}, corresponding to the reverse of the initial permutation IP.

The security of DES encryption lies in the round function F, which, through its combination of linear and non-linear components, provides both diffusion and confusion [24]. Specifically, F consists of the following operations: 1) *Expansion Permutation (*EP*)*: expands R_i from 32 to 48 bits by duplicating half of the bits; 2) KeyMix: XORs the round key K_i into the 48 bits obtained in the previous stage; 3) *Substitution (*$\texttt{S}-\texttt{boxes}$*)*: applies 8 distinct S-boxes independently on a 6-bit portion of the previous stage, producing 8 4-bit outputs according to a non-linear transformation; 4) *Permutation (*P*)*: rearranges the result according to a fixed permutation. The 8 S-boxes provide the essence of DES security since, without them, the cipher would be linear, and trivially breakable. Their implementation is often provided in the form of fixed lookup tables, in which each of the 6 input bits are used to select an independent cell element storing 4 bits.

With the evolution of cryptographic techniques and the emergence of more advanced threats, DES faced limitations, particularly in its susceptibility to brute-force attacks due to its relatively small key size. Indeed, in the *Known-Plaintext Attack (KPA)* model, an attacker having both the plaintext p and its encrypted version c can perform an exhaustive search through all the 2^{56} (since only 56 out of 64 bits of the key are actually used) possible keys, invoking each time the encryption routine, and checking if the result is equal to c.

3DES. In response to the vulnerability of DES, the triple-DES (3DES) cipher emerged as a successor, offering enhanced security features while maintaining compatibility with existing systems. Standardized for key management in ISO/IEC 18033-3 among the others, 3DES employs a triple-encryption process relying on DES building blocks. Specifically, given three 64-bit keys k_1, k_2, k_3, the DES encryption function E_k and the DES decryption function D_k, the encryption of the plaintext p is obtained as $c = E_{k_3}(D_{k_2}(E_{k_1}(p)))$.

The 3DES cipher offers three three keying options:
O1: $k_1{=}k_2{=}k_3$, with a key-size equivalent to DES;
O2: $k_1{=}k_3$, $k_1{\neq}k_2$, with an equivalent key size of 112 bits;
O3: $k_1{\neq}k_2{\neq}k_3$, with an equivalent key size of 168 bits.
Considering that, as for DES, only 56 out 64 bits of the keys are actually used in the encryption process, the previous keying options correspond to a computation complexity of 2^{56}, 2^{112} and 2^{168} for 3DES-O1, 3DES-O2 and 3DES-O3 respectively.

ISO/IEC 18033-3 does not allow 3DES-O1, and recommends instead 3DES-O3 with respect to 3DES-O2 because of its better security with respect to plain brute-force attacks. Nonetheless, 3DES-O3 is susceptible to *Meet-In-The-Middle (MITM)* attacks, capitalizing on the availability of an exponential amount of memory to diminish the number of keys searched through a brute-force attack. Indeed, by defining an equivalent encryption function $E_{k_{23}} = E_{k_3}(D_{k_2}(p))$, 3DES-O3 encryption can be represented as $c = E'_{k_{23}}(E_{k_1}(p))$, with k_{23} being the 112-bit key obtained from the two 56-bit keys k_2 and k_3. The MITM attack first constructs a table containing 2^{56} pairs (c', k'_1), each corresponding to an encryption of the plaintext p using a distinct key k_1. Subsequently, a key-recovery attack on $E_{k_{23}}$ can be executed using the same plaintext p, wherein, after each encryption of p with k_{23}, resulting in the ciphertext c'', a search is conducted for an entry in the table for which $c' = c''$. This approach necessitates $2^{56} + 2^{112} \approx 2^{112}$ encryption operations and 2^{56} data entries, yielding a security level close to the one of 3DES-O2. Since 3DES-O2 uses 64 fewer bits for the key storage, and it does provide almost the same security of 3DES-O3, it is still the preferred keying option of choice for the majority of 3DES ciphers.

B. Algebraic structure of S-boxes

To withstand linear cryptanalytic attacks, symmetric ciphers must incorporate non-linear components, most often in the form of substitution boxes or S-boxes. An S-box represents a vectorial Boolean function $\{0,1\}^x \mapsto \{0,1\}^y$, mapping input bitstrings of length x to output bitstrings of length y. The function is usually realized as a lookup table with 2^x cells, each storing an output bitstring. The embedding of S-boxes into quantum circuits demands however their adaptation as reversible functions.

In the literature, various tools have recently been proposed to convert generic S-boxes to standard Boolean or reversible gates, optimizing metrics such as gate count, depth, or auxiliary bits or qubits. As an example, in [25]–[27] the authors propose tools based on breadth-first search strategies and graph meet-in-the-middle strategies, that however face limitations with S-boxes having more than 4-bit inputs. An alternative approach involves SAT-solvers [28], [29], although their practicality diminishes for S-boxes with 6 or more input bits. Notably, the quantum implementation of DES S-boxes remains unexplored.

In classical settings, the first work analyzing DES S-boxes in terms of standard Boolean gates was presented in [30]. The technique, termed *bitslicing* in a later work [31], leverages a Single-Instruction Multiple-Data view, wherein multiple processors operate in parallel on independent bits of the data. This method, relying solely on standard Boolean gates, offers the most efficient implementations of DES S-boxes in terms of gate count.

C. Grover's framework

Grover's original algorithm [32] can be restated as a quantum framework searching for the unique value k for which the vectorial Boolean function $f : \{0,1\}^n \mapsto \{0,1\}^q$ evaluates to the all-1's bitstring of length q. In the quantum formulation of the function f through the operator $\boldsymbol{U}_f$, the length-n bitstrings composing the input domain are thought as the labels of the 2^n basis states, stored on n input qubits, spanning the quantum state of the system across the algorithm. The framework, whose visualization in the quantum circuit model is given in Fig. 2a, relies on three stages.

1) Domain superposition preparation ($\mathtt{H}^{\otimes n}$). It corresponds to n `H` gates, each applied, in parallel, on the n input qubits. The application of the operator to the initial state $|0^n\rangle$ results in the quantum state

$$|\sigma\rangle = \sum_{i \in \{0,1\}^n} \frac{1}{\sqrt{2^n}} |i\rangle ,$$

corresponding to the uniform superposition of all the 2^n orthonormal basis states labeled as bitstrings of the domain of f.

2) Oracle ($\boldsymbol{U}_{\mathrm{ref}^{\perp}(k)}$). The operator changes the sign of the amplitude of the basis state $|k\rangle$, while leaving all the other ones unaffected. When applied to a superposition state, its effect can be visualized as a reflection of the quantum state around all the 2^n-1 basis states orthogonal to $|k\rangle$. This interpretation justify the use of the reflection notation to represent such operator, as derived from [33].

The implementation of $\boldsymbol{U}_{\mathrm{ref}^{\perp}(k)}$ in terms of standard quantum gates is usually retrieved starting from two quantum operators, that we denote as $\boldsymbol{U}_f$ and $\boldsymbol{U}_{\mathrm{ref}^{\perp}(1)}$. The first operator, $\boldsymbol{U}_f$, corresponds to a reversible implementation of the Boolean function f, and, for non-trivial functions, relies on m ancillary qubits. The function stores on a subset of q qubits, with $q < n+m$, the all-1's bitstring if and only if the basis state encoded in the input qubits is equal to $|k\rangle$. $\boldsymbol{U}_{\mathrm{ref}^{\perp}(1)}$, on the other hand, corresponds to a reflection of the quantum state around the 2^n-1 basis states orthogonal to $|1^q\rangle$, and it is implemented through a multi-qubit `Z` gate involving all the q qubits containing the output of the f evaluation, as shown in Fig. 2b (left).

Finally, since Grover's oracle requires only a change of the sign in the amplitude of $|k\rangle$, all the qubits used by $\boldsymbol{U}_f$ for the evaluation of f have to be restored to their initial state. For this reason, following a widespread *compute-uncompute* pattern, the $\boldsymbol{U}_{\mathrm{ref}^{\perp}(1)}$ operator is followed by the $\boldsymbol{U}_f^{\dagger}$ operator, corresponding to the application of the same sequence of gates used in the reversible implementation of $\boldsymbol{U}_f$, but applied in reverse order.

3) Diffusion ($\boldsymbol{U}_{\mathrm{ref}(\sigma)}$). It is the other main reflection operator used in Grover's framework, and it reflects the quantum state around the initial superposition state $|\sigma\rangle$. Its implementation can be derived starting from its reformulation as $\boldsymbol{U}_{\mathrm{ref}(\sigma)} = \mathtt{H}^{\otimes n}\boldsymbol{U}_{\mathrm{ref}^{\perp}(0)}\mathtt{H}^{\otimes n}$. The $\boldsymbol{U}_{\mathrm{ref}^{\perp}(0)}$ gate corresponds to a reflection around the basis states orthogonal to the all-0's basis state, and can be implemented in terms of a multi-qubit `Z` gate and $2n$ `X` gate, as shown in Fig. 2b (right).

Number of Grover's iterations. Repeating steps 2) and 3) approximately $\sqrt{2^n}$ times results in a probability of measuring the basis state $|k\rangle$ close to 1. In [34], the authors showed that the optimal number of iterations is $\approx 0.58\sqrt{2^n}$ to obtain a probability of observing $|k\rangle$ close to 0.84. Additionally, they generalized the framework to the case of r multiple solutions, for which the number of iterations is reduced by a factor of $\sqrt{r}$.

Grover's algorithm parallelization. In [35] the authors discuss two distinct ways to parallelize Grover's algorithm across S quantum machines, denoted as inner and outer parallelization. Inner parallelization divides the search space into S disjoint subsets, assigning each subset to a distinct machine. Since each machine's search space is reduced, while the oracle circuit stays unchanged, the number of iterations required by each machine to observe the target solution is reduced by a factor of $\sqrt{S}$. On the other hand, the outer parallelization runs S instances of the full algorithm in parallel. Assuming that the verification of the solution obtained through the quantum circuit can be efficiently checked classically, which is the case for quantum key-recovery attacks, only one out of the S instances must succeed for the whole algorithm to succeed, allowing to reduce the number of iterations of all the instances.

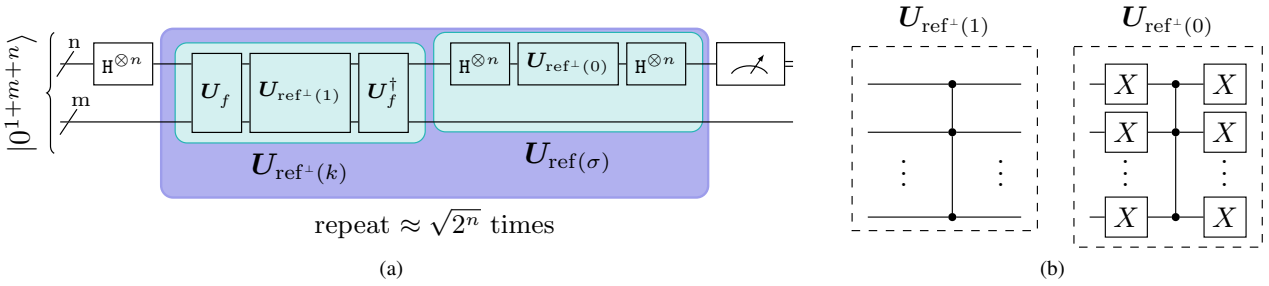

Fig. 2. (a) Circuit representation of Grover's framework in terms of the oracle reflection operator $\boldsymbol{U}_{\mathrm{ref}^\perp(k)}$ and the diffusion reflection operator $\boldsymbol{U}_{\mathrm{ref}(\sigma)}$, and their decomposition in terms of the quantum implementation of the Boolean function $\boldsymbol{U}_f$ and the two reflections $\boldsymbol{U}_{\mathrm{ref}^\perp(1)}$ and $\boldsymbol{U}_{\mathrm{ref}^\perp(0)}$. (b) Implementation of the $\boldsymbol{U}_{\mathrm{ref}^\perp(1)}$ and $\boldsymbol{U}_{\mathrm{ref}^\perp(0)}$ reflection operators in terms of multi-qubits Z gate and single-qubit X gate.

The authors show that also in this case the number of iterations required is reduced by a factor of $\sqrt{S}$.

III. Quantum Circuits for Key-Recovery

Notation. Following the notation of [36], we denote a *quantum register* — i.e., a set of qubits— using a boldface, lowercase font, additionally underlying its name, as in $\underline{\boldsymbol{a}}$. As common in the literature, we assume that all qubits are initialized to $|0\rangle$.

A. Grover-based key-recovery attack on DES

The Grover-based key-recovery attack on DES relies on the availability of a plaintext p and the corresponding ciphertext $c = E_k(p)$ encrypted used an unknown secret key k. To rephrase the attack in terms of a search procedure employing the $\boldsymbol{U}_f$ operator required by Grover's oracle, we define the vectorial Boolean function $f : \{0,1\}^{56} \mapsto \{0,1\}^{64}$. Such a function takes as input a 56-bit candidate key k' (remember that, as seen in Sec. II-A, only 56 out of 64 bits of the cipher key are actually used in DES encryption), and returns as output the bitstring 1^{64} on a quantum register of dimension 64 if and only if $E_{k'}(p) = c$, which implies $k' = k$. By accepting as input all the possible keys in superposition, and thanks to the realization of Grover's oracle through the quantum circuit implementation of $\boldsymbol{U}_f$, the quantum exhaustive search can be performed using a computational complexity proportional to $\sqrt{2^{56}}$.

To ensure accuracy in our attack on DES, we must first address the challenge of *spurious keys* — multiple keys that yield the same ciphertext for a given plaintext. In [3], the authors demonstrated that while multiple plaintext-ciphertext pairs are typically required, even a single pair provides a 0.37 probability of identifying the correct private key. Furthermore, parallelizing Grover's algorithm across multiple instances diminishes the need for multiple pairs. Hence, our attack relies on a single plaintext-ciphertext pair.

In the following, tracing the operators described in Sec. II-C, we describe all the quantum circuits needed in our circuit design. The quantum circuit representation of our proposal is given in Fig. 3a, in which we denote as: $\underline{\boldsymbol{k}}$ the quantum register containing the superposition state representing the encoding of all the possible keys k'; $\underline{\boldsymbol{p}}$ the quantum register containing the encoding of the plaintext p, and its changes across the circuit; $\underline{\boldsymbol{p_E}}$ and $\underline{\boldsymbol{p_S}}$ two auxiliary registers used in the quantum implementation of the EP block and the S-boxes, respectively.

1) Domain superposition preparation As explained in Sec. II-C, this stage is used to generate a uniform superposition of all the quantum states labeled as the bitstrings composing the domain of f. In DES key-recovery attack, such superposition contains all the possible length-56 bitstrings, corresponding to all the possible choices of a candidate key. This step is implemented through a depth-1 layer of 56 Hadamard gates, each applied on the 56 qubits of the quantum register $\underline{\boldsymbol{k}}$.

2) Oracle As we described in Sec. II-C, the oracle can be described in terms of two quantum operators: $\boldsymbol{U}_f$ and $\boldsymbol{U}_{\mathrm{ref}^\perp(1)}$. Since the $\boldsymbol{U}_{\mathrm{ref}^\perp(1)}$ has already been discussed in Sec. II-C, and it corresponds to a multi-qubit Z gates acting on 64 qubits, we will focus in the following on $\boldsymbol{U}_f$, whose realization relies on several subcircuits.

The first subcircuit employed in $\boldsymbol{U}_f$, denoted $\boldsymbol{U}_{BE(p)}$ in Fig. 3a, realizes a basis encoding of the known plaintext bitstring p in the quantum register $\underline{\boldsymbol{p}}$. Such procedure involves the application of an X gate on all the qubits of $\underline{\boldsymbol{p}}$ for which the corresponding bis of p has value 1. Since all the gates can be applied in parallel, this procedure has a trivial depth of 1; the average number of X gates is instead 32.

The second component of $\boldsymbol{U}_f$ is the $\boldsymbol{U}_{E_k}$ operator, corresponding to the quantum implementation of DES encryption circuit. Our proposal for such a circuit, representing the core of the quantum key-recovery attack to DES, is shown in Fig. 3b. With respect to the classical implementation shown in Fig. 1, we note the explicit representation of both the round function F and the Swap operations swapping the left and right portion of the plaintext at the end of each round. More importantly, we note the absence of the $\mathtt{PC}-1$ block involved in the key-schedule procedure. Indeed, while in the classical case this publicly-available, fixed permutation is used to select 56 out of the 64 bits of the input key, an exhaustive search algorithm has no use for such 8 additional bits. As a consequence, both the unneeded 8 qubits and the permutation $\mathtt{PC}-1$ can be avoided. Finally, retracing the classical steps of the encryption routine, we employ two quantum registers, $\underline{\boldsymbol{p_L}}$ and $\underline{\boldsymbol{p_R}}$, corresponding to the 32 leftmost and 32 rightmost qubits of $\underline{\boldsymbol{p}}$ respectively.

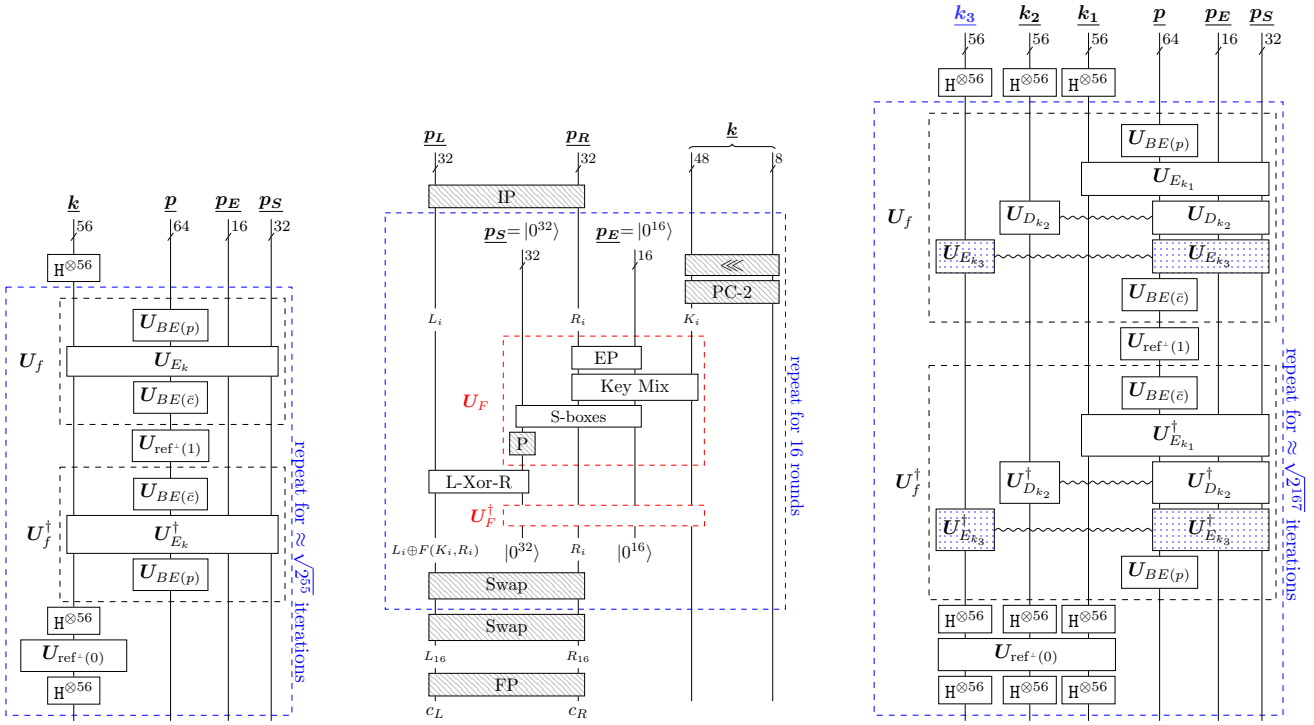

Fig. 3. The circuits execution flow is from top to bottom; boldface, underlined name denotes quantum registers. (a) Quantum key-recovery attack circuit for DES, involving basis encoding ($\boldsymbol{U}_{BE}(\cdot)$) of the plaintext p and the bitwise complement $\bar{c}$, DES encryption $\boldsymbol{U}_{E_k}$, and reflection operators ($\boldsymbol{U}_{\text{ref}^{\perp}(0)}$ and $\boldsymbol{U}_{\text{ref}^{\perp}(1)}$)(check Fig. 2b). (b) Quantum circuit for $\boldsymbol{U}_{E_k}$, tracing the components shown in Fig. 1, with shaded boxes indicating qubit relabeling. Note that $\texttt{PC}-1$ is irrelevant. (c) Quantum key-recovery attack circuit for 3DES-O3, using the decryption operator $\boldsymbol{U}_{D_{k_2}}$ relying on similar subcircuits as $\boldsymbol{U}_{E_{k_1}}$.

All the shaded operations represented if Fig. 3b, namely the Input Permutation (IP), the Final Permutation (FP), the Permutation (P), the $\texttt{PC}-2$, the Swap, and the $\lll$, correspond to fixed, publicly available, permutations of qubits. Therefore, they can be easily implemented at circuit generation time through a simple relabeling of the corresponding qubits, and hence have no gate cost in their quantum implementation. Note that the previous reasoning imply that the 16 round keys K_i are fully obtained by qubits relabeling.

On the other hand, the Expansion Permutation (EP), in the classical case, copies half of the 32 input bits into 16 auxiliary bits. In the quantum case, such a copy is performed through a depth-1 layer of CNOT gates, each one having as control qubit one of the 16 qubits of $\boldsymbol{p_R}$ to be copied, and as target one of the 16 qubits of the auxiliary register $\boldsymbol{p_E}$, initially thought in state $|0^{16}\rangle$.

The KeyMix stage XORs the bits of the round key K_i into the expanded right portion of the plaintext stored in $\boldsymbol{p_R}$ and $\boldsymbol{p_E}$. Its quantum implementation involves a depth-1 layer of 48 CNOT gates, each one having as control a qubit taken from $\boldsymbol{k}$, and as target the corresponding qubit taken from either $\boldsymbol{p_R}$ or $\boldsymbol{p_E}$.

The $\texttt{S}-\texttt{boxes}$ stage, involving non-linear operations on the input, represents the most demanding part of the entire quantum circuit. We recall that DES uses 8 non-linear S-boxes, each of which takes 6 bits in input and produces 4 bits in output. As explained in Sec. II-B, neither SAT-based tools nor state-of-the-art breadth-first heuristic are capable of efficiently explore these sizes. For this reason, to obtain a reversible implementation for our quantum circuit design, we employed the bitslicing technique of [31] detailed in Sec. II-A, obtaining at the end a classical description of all the S-boxes in terms of standard Boolean gates, namely NOT, AND, OR and XOR. From the Boolean description provided earlier, we can derive a quantum circuit implementation using the NCT gate set. In this translation, classical NOT gates become quantum X gates (that is, quantum NOT gates), XOR gates correspond to either a single or two consecutive CNOT gates, depending on qubit reuse. Similarly, AND gates are represented by Toffoli gates, while OR gates are realized using a Toffoli gate and five X gates, following De Morgan's law. The described translations are illustrated in Fig. 4.

Note that each S-box takes as input 6 qubits from $\boldsymbol{p_R}$ and $\boldsymbol{p_E}$, and produces its output on 4 auxiliary qubits belonging to the auxiliary register $\boldsymbol{p_S}$, initially in state $|0^32\rangle$. We report in Tab. I the implementation results of the S-boxes in terms of the NCT gate set.

Finally, the $\texttt{L}-\texttt{Xor}-\texttt{R}$ gate XORs the result of the S-boxes stages, stored in the $\boldsymbol{p_S}$ register, into left portion of the plaintext. This stage is implemented using a depth-1 layer of CNOT gates, each one having as control a qubit of $\boldsymbol{p_S}$, and as target the corresponding qubit of $\boldsymbol{p_L}$.

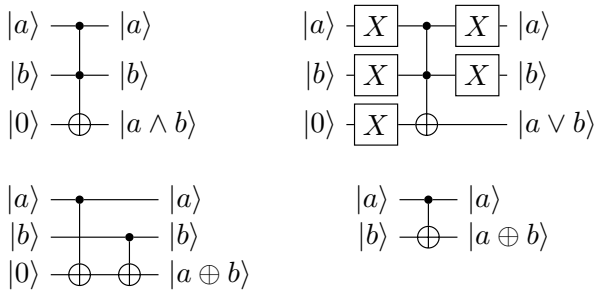

Fig. 4. Translation of the Boolean operators AND ($\wedge$), OR ($\vee$) and XOR ($\oplus$) in terms of the NOT-CNOT-Toffoli (NCT) gate set.

At the end of the $\mathtt{L} - \mathtt{Xor} - \mathtt{R}$ gate, the application of the $\boldsymbol{U}_F^\dagger$ operator allows resetting the state of the auxiliary quantum registers $\underline{\boldsymbol{p_S}}$ and $\underline{\boldsymbol{p_E}}$ to the all-0's state, and brings back the register $\underline{\boldsymbol{p_R}}$ to its original state, allowing the reuse the same quantum registers across all the 16 rounds of encryption.

Finally, the third and last component of $\boldsymbol{U}_f$, denoted $\boldsymbol{U}_{BE(\bar{c})}$, performs a basis encoding of the bitwise complement of the bitstring c in the quantum register $\underline{\boldsymbol{p}}$. In this way, if the $\boldsymbol{U}_{E_k}$ operator produced on the quantum register $\underline{\boldsymbol{p}}$ a quantum state labeled as c, the application of $\boldsymbol{U}_{BE(\bar{c})}$ will contain in output a length-64 bitstring of 1's, that can be used in the next step to implement the reflection $\boldsymbol{U}_{\mathrm{ref}^\perp(1)}$ of the oracle. The operator involves the application of an X gate to all the qubits of $\underline{\boldsymbol{p}}$ for which the corresponding bit of c has value 0. Since all the gates can be applied in parallel, this procedure has a trivial depth of 1. The number of X gate applied is, on average, equal to 32.

3) Diffusion As explained in Sec. II-C, this stage involves the application of a depth-1 layer of 56 Hadamard gates on the qubits storing the input superposition prepared in stage 1) on $\underline{\boldsymbol{k}}$, followed by the $\boldsymbol{U}_{\mathrm{ref}^\perp(0)}$ gate performing a reflection around the basis states orthogonoal to $|0^{56}\rangle$, followed by another stage 1) layer. We remark that the $\boldsymbol{U}_{\mathrm{ref}^\perp(0)}$ operator is implemented as a three layer circuit on the qubits of $\underline{\boldsymbol{k}}$, the first and the last one consisting in the parallel application of 56 X gates, while the second one being a 56-qubit Z gate.

Number of Grover's iterations. DES exhibits the complementation property, meaning that $E_k(p) = c \iff E_{\bar{k}}(\bar{p}) = \bar{c}$, in which $\bar{x}$ is the bitwise complement of x. For this reason, the expected number of solutions is 2, and the complexity of the key-recovery attack on DES is equal to $\sqrt{\frac{2^{56}}{2}} = \sqrt{2^{55}}$.

B. Quantum key-recovery attack on 3DES

The quantum circuit for 3DES, visualized in Fig. 3c, derives directly from the one of DES detailed in the previous section. In 3DES, indeed, the encryption follows a three-stage process, encryption, decryption, and encryption again, necessitating three distinct keys, k_1, k_2, and k_3. Both the encryption and the decryption routine are the ones used in DES.

Compared to DES, the key difference lies in the use of three quantum registers: $\underline{\boldsymbol{k_1}}$, $\underline{\boldsymbol{k_2}}$, and $\underline{\boldsymbol{k_3}}$, each comprising 56 qubits, as opposed to the single 56-qubit register $\underline{\boldsymbol{k}}$. The stage 1) of Grover, responsible for input preparation, now demands 168 Hadamard gates applied in parallel across the key registers. Consequently, the $\boldsymbol{U}_{\mathrm{ref}^\perp(1)}$ operator in Grover's diffusion phase features a 168-qubit Z gate. Additionally, owing to the Feistel-network structure of DES, the decryption operator $\boldsymbol{U}_{D_{k_2}}$ mirrors the encryption routine, with the only change being in the reversed order of round keys generated by the key scheduling algorithm.

While the circuit illustrated in Fig. 3c represents the general key-recovery attack for 3DES-O3, it can be easily adapted for 3DES-O2, where $k_1 = k_3$, by simply omitting $\underline{\boldsymbol{k_3}}$ and using $\underline{\boldsymbol{k_1}}$ in all the places in which $\underline{\boldsymbol{k_3}}$ is required. As a consequence, the Grover's diffusion stage features a 112-qubit Z gate.

Finally, since 3DES inherits from DES the complementation property, the number of Grover's iterations expected to measure the keys with high probability is equal to $\sqrt{2^{111}}$ for 3DES-O1 and $\sqrt{2^{167}}$ for 3DES-O2.

Meet-in-the-middle (MITM) strategy for 3DES-O3. In the classical case, an interesting aspect of 3DES-O3 is the possibility of using a MITM strategy to speedup a key-recovery attack at the expense of an exponential amount of memory. Assuming the physical feasibility of the realization of the Quantum Random-Access Memory (QRAM), we can adapt this technique to the quantum case.

In the quantum attack on 3DES-O3 employing a MITM strategy, the starting point is the generation and storage, on the QRAM, of all the 2^{56} ciphertext-key pairs (c', k_1') obtained by repeating the encryption procedure E_{k_1} on the same input plaintext p, but using each time a distinct key k_1' out of the 2^{56} possible ones.

After that, we can reuse the circuit components shown in Fig. 3c for the MITM strategy, with the main notable difference being the absence of the operators $\boldsymbol{U}_{E_{k_1}}$ and $\boldsymbol{U}_{E_{k_1}}^\dagger$. Additionally, the quantum register $\underline{\boldsymbol{k_1}}$ is thought in state $|0^{56}\rangle$, and it is not included in stage 1) of Grover's framework (and, as a consequence, in stage 3) neither). Finally, in place of the $\boldsymbol{U}_{BE(\bar{c})}$, the MITM adaptation relies on random access to the memory. Indeed, after the operator $\boldsymbol{U}_{E_{k_3}}$, the quantum register $\underline{\boldsymbol{p}}$ holds a candidate ciphertext c'', that can be used to access the QRAM and store the corresponding k_1' on $\underline{\boldsymbol{k_1}}$ if present, additionally setting an auxiliary qubit to 1 to signal success, while leaving the register untouched if not present.

In this scenario, the number of Grover's iterations required by the 3DES-O3 approach is equal to the ones required by 3DES-O2. However, each iteration must also account for the QRAM access cost.

IV. Experimental evaluation

In this section, we conduct a detailed analysis of the computational costs involved in our design, considering both the NCT+H and the Clifford+T gate sets. Additionally, we present designs optimized for low-qubit and low-depth devices. Furthermore, we offer a comparative assessment against state-of-the-art proposals for various block ciphers. All the components of our quantum circuit design have been validated

TABLE I
S-BOXES COMPLEXITY MEASURES IN TERMS OF THE NOT-CNOT-TOFFOLI (NCT) AND THE CLIFFORD+T GATE SET.

Metric	S1	S2	S3	S4	S5	S6	S7	S8
Width	63	56	57	42	62	57	57	54
NOT	99	72	82	33	79	59	77	59
CNOT	58	52	58	47	62	58	54	54
Toffoli	32	29	27	17	29	26	29	25
Depth	51	38	41	29	62	43	51	37

Remark : The 6 input and 4 output qubits are not included in the width count. The depth is obtained considered an equal weight for all gates.

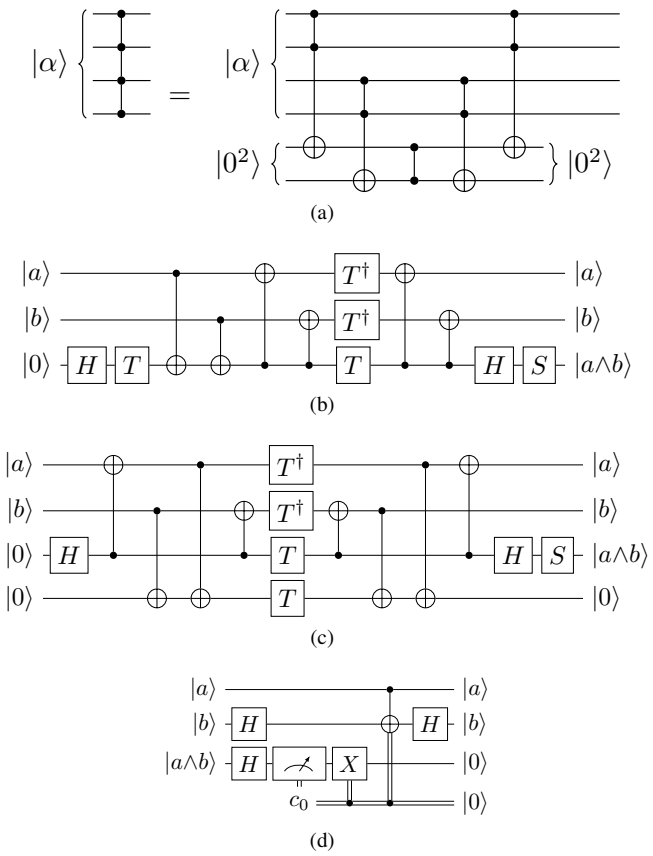

Fig. 5. (a)Multi-qubit Z decomposition into a logarithmic-depth circuit, with linear amount of Toffoli gates and ancillary qubits, as reported in [22]. (b) Decomposition of QAND into T-depth 2 circuit without ancillae [37]. (c) Decomposition of QAND T-depth 1 circuit with one ancilla [3]. (d) QAND† T-depth 0 implementation [38].

using the Atos Quantum Learning Machine simulator. The entire source code will be made available upon publication.

A. Component design under different metrics and gate sets

Low-width vs low-depth circuits. To offer a comprehensive view of the computational complexity of our proposal, we fine-tuned the design of our quantum circuit considering both the number of qubits (referred to as width) and the depth. In our approach, the segment of the circuit that allows for a significant trade-off between depth and width is the S-boxes stage of the $\boldsymbol{U}_{E_k}$ unitary. In the design optimized for *low-depth*, all 8 distinct S-boxes operate simultaneously. To enable full parallelization, all the auxiliary qubits required for each S-box must be available at this stage. Hence, the number of auxiliary qubits needed is the sum of those required by each S-box, as shown in Tab. I. The depth, conversely, is determined by the deepest S-box circuit, which is 62 for the S5 box. In contrast, the *low-width* implementation executes the 8 S-boxes sequentially, utilizing the same ancillary qubits to reduce the total width. Thus, the number of qubits needed is determined by the widest S-box circuit, which is 63 for the S1 box. The depth is calculated by summing the depths of all the S-boxes, with limited potential for parallelization across them.

NCT+H gate set. The first gate set targeted in our exploration is the reversible NCT gate set augmented with the Hadamard H gate. Such set necessitates the translation of the multi-qubit Z gates used in the implementation of the $\boldsymbol{U}_{\mathrm{ref}^{\perp}(0)}$ and $\boldsymbol{U}_{\mathrm{ref}^{\perp}(1)}$ operators employed in Grover's algorithm's oracle and diffusion stages. For instance, the diffusion stage entails a 56-qubit Z gate for DES, a 112-qubit Z gate for 3DES-O2 and 3DES-O3 with MITM, and a 168-qubit Z gate for 3DES-O3. Moreover, the oracle stage demands a 64-qubit Z gate for all variants, except 3DES-O3 with MITM, which only requires a single-qubit Z gate. We employed the strategy detailed in [22] and shown in Fig. 5a for both low-depth and low-width translations of the multi-qubit Z gates. Although such strategy is used to translate a generic multi-controlled gates in terms of Toffoli gates, plus an additional single qubit gate, nevertheless the multi-qubit Z gate can be thought as a multi-controlled Z gates on which all qubits are thought as control, and there is no target qubit. Specifically, the proposal suggests a translation requiring at most $m-2$ ancillary qubits, $2m-4$ Toffoli gates, and $2\log_2(m)$ depth for an m-qubit Z. For the low-depth translation, the strategy can reuse all the ancillary qubits previously used by the S-boxes. For low-width translation, instead, we made minor adjustments considering that, besides the 63 ancillary qubits already required by the S-boxes, the $\boldsymbol{p_S}$ and $\boldsymbol{p_E}$ quantum registers (sized 32 and 16, respectively) are in the all-0's state before applying either the oracle or the diffusion reflection operators. Consequently, we could borrow up to 111 already available ancillary qubits for the multi-qubit Z translation. Although this sufficed for the logarithmic-depth decomposition for all oracle and most diffusion reflections, 3DES-O3 without MITM required 55 additional qubits for its diffusion reflection. Nonetheless, we can still employ the same strategy devised in [22] by first dividing the qubits involved in the multi-qubit Z gates in two distinct portions, and then operate two independent logarithmic decompositions on them, one after the other. This strategy results in a depth that is almost twice with respect to the low-depth one, but at the same time does not incur in any auxiliary qubit requirements.

Clifford+T gate set. The Clifford+T gate basis, introduced in [39], is deemed the most promising for fault-tolerant quantum computation. However, implementing the T gate in a fault-tolerant manner incurs considerable overhead compared to other gates [40]. Therefore, computational complexities are

TABLE II
COMPARISON OF THE GROVER-BASED KEY-RECOVERY ATTACK ON DES AND 3DES WITH RESPECT TO THE OTHER STATE-OF-THE-ART CIPHERS USED IN ISO/IEC-18033-3 HAVING A 128-BIT KEY-SIZE. ALL THE RESULTS ARE EXPRESSED IN BASE-2 LOGARITHM.

Cipher	NCT + H					Clifford + T				
	W	G	D	D × W	D × G	W	T	TD	TD × W	TD × T
DES$^{\diamond}$	8	43	41	48	84	8	41	40	48	83
DES$^{\blacklozenge}$	9	43	39	48	82	10	41	37	46	80
3DES-O2$^{\diamond}$	8	72	70	78	142	8	71	69	77	141
3DES-O2$^{\blacklozenge}$	9	72	68	77	140	10	71	66	75	138
3DES-O3$^{\diamond}$	8	100	97	106	197	9	98	97	105	196
3DES-O3$^{\blacklozenge}$	9	100	95	105	195	10	98	94	103	193
3DES-O3-$\sqrt[3]{M}^{\diamond}$	8	72	74	82	146	-	-	-	-	-
3DES-O3-$\sqrt[3]{M}^{\blacklozenge}$	9	72	74	83	146	-	-	-	-	-
3DES-O3-$\sqrt{M}^{\diamond}$	8	72	83	91	155	-	-	-	-	-
3DES-O3-$\sqrt{M}^{\blacklozenge}$	9	72	83	93	155	-	-	-	-	-
AES-128 [38]	12	83	75	88	157	12	79	71	83	151
Camellia-128 [8]$^{\diamond}$	9	80	77	85	157	9	80	78	87	158
Camellia-128 [8]$^{\blacklozenge}$	-	-	-	-	-	10	79	69	87	148
SEED [9]	15	84	79	95	164	15	84	76	91	94
HIGHT [7]	9	82	75	85	158	9	84	78	87	96

$^{\diamond}$Low-width implementation.
$^{\blacklozenge}$Low-depth implementation.
Remark: Shaded rows refer to 3DES-O3 with MITM strategy measures for a cubic-root $\sqrt[3]{M}$ and square-root $\sqrt{M}$ access cost to the size-M QRACM. The access cost is thought as having impact on the overall circuit depth, leaving the gate count and number of qubits unaffected.

often assessed based on the T-count, i.e., the number of T gates, and the T-depth, the maximum number of T gates acting sequentially on a single qubit.

In our design, the gates necessitating T gates are only the Toffoli gates used in S-boxes and multi-qubit Z gate decompositions. Notably, our approach exclusively utilizes QAND gates, that is, Toffoli gates for which the target qubit is assumed to be in state $|0\rangle$. To implement the QAND gate efficiently, [3] offers a T-depth-1 circuit using a single ancillary qubit. Additionally, [37] proposes a QAND† gate without T gates, employing only a measurement gate. This approach involves applying a H gate on the qubit containing the QAND gate's result, followed by measurement. If the measurement outcome is 1, the qubit is reset to $|0\rangle$, and a phase change is applied to the global quantum state using a decomposed CZ gate. Regarding the low-width translation of the QAND gate, [37] presents an approach achieving a T-depth of 2 without any ancillary qubits. Furthermore, the QAND† circuit can be implemented using measurement-based uncomputation as for the previous case.

B. Quantum computational complexity measures

Our analysis, presented in Tab. II, compare the performance of our designs under different optimization strategies and gate sets. Using the NCT+H gate set, we quantify the computational complexities in terms of qubit count (W), gate count (G), and depth (D). For the Clifford+T gate set, we provide both the T-count (T) and the T-depth (TD). From the table, we observe that considering plausible gate execution times of approximately 100 nanoseconds [41], DES's depth of 2^{39} suggests that the key can be retrieved within 15 hours. Even with NIST's estimate of executing 2^{40} in about a year [5], the time to break DES is of approximately half a year. Additionally, NIST's estimates of 2^{64}, being the number of gates that a quantum computer having gate execution times comparable to classical Boolean gate execution times can execute in a decade, implies a potential breach of 3DES-O2 within 160 years.

In the same table, we additionally report the metrics of depth×width ($D\times W$) and the T-depth×width ($TD\times W$). The aggregation of depth and width measures, proposed in [21], addresses the challenge of estimating the quantum circuit complexity in terms of the different resources used. Indeed, the standard view of quantum gates as analgous to the static Boolean logic components used in the classical paradigm of computation do not directly apply to quantum computing, in which it seems more plausible to think of qubits as static components, upon which quantum gates are dynamically applied by a classical controller. For this reason, considering a classical controller attached to each qubit, the authors of [21] evaluate the quantum complexity of a circuit in terms of total interventions by all the classical controllers.

Grover's parallelization under depth constraints. As we reported in Sec. II-C), Grover's algorithm does not significantly benefit from parallelization, as it only reduces the number of iterations of a single quantum device by a factor of $\sqrt{S}$ when S quantum devices run in parallel. Since the number of gates and depth required by each quantum device remains approximately the same, the parallelization strategy increases the overall number of quantum gates required across all devices by $\sqrt{S}$. When the depth achievable by each quantum device is fixed to a maximum value, called $MAXDEPTH$ in NIST standardization call [5], we can compute the number of quantum devices required for a full parallelization of Grover's algorithm as $S = (D/MAXDEPTH)^2$, with D being the depth required by a single device in the original, non-parallelized version. Using the value $MAXDEPTH = 2^{64}$ reported by NIST, 3DES-O2 requires almost $(2^4)^2 = 256$ devices to be parallelized, and hence retrieve the key in approximately a decade.

As shown in [22, Eq.10], under these hard depth constraints the overall number of gates can be expressed as the depth of a single quantum instance multiplied by the number of gates of a single quantum instance. The result of such multiplication, denoted as *QAES* in [5], [6], justify the use of depth×number of gates ($D\times G$) and T-depth×number of T gates ($TD\times$T) as complexity metrics to take into account.

3DES-O3 MITM with QRAM. As discussed in Sec. III-B, the 3DES-O3 with MITM strategy, both in classical and quantum scenarios, heavily relies on the availability of substantial memory resources.

While standard definitions for Quantum Random Access Memory (QRAM) are not universally established, [23] offers a comprehensive survey of QRAM models. Firstly, the authors distinguish between circuit-based and gate-based QRAM, depending on whether standard quantum circuit gates or specialized gates are utilized in implementation. Additionally, they categorize QRAM as either Quantum Random Access Classical Memory (QRACM) or Quantum Random Access Quantum Memory (QRAQM). In the former, a fixed list of entries that can be queried in superposition, while in the latter, the entries themselves can exist in superposition. The authors argue that the most optimistic scenario regarding storage requirements greater than 2^{50} is the gate-based QRACM model. In this scenario, the overhead scales with $o(M)$, where M represents the size of the quantum memory.

In the key-recovery attack against 3DES-O3, the memory size required to perform a MITM strategy is of 2^{56}. To avoid overly optimistic constant-depth costs, which would yield complexity similar to 3DES-O2, and pessimistic (yet not unrealistic) linear access costs, for which the 3DES-O3 with MITM would have no benefit with respect to the plain 3DES-O3, we consider in our analysis both cubic-root and square-root access costs memory in our analysis. While the latter increases the overall depth by $\approx 2^{13}$ with respect to 3DES-O2, the former has a more modest impact of $\approx 2^4$.

C. Comparison with state-of-the-art symmetric ciphers

To gauge the security robustness of DES, we conduct a comparative analysis of computational complexities obtained from our design against other symmetric ciphers standardized in ISO/IEC-18033-3, all requiring a key-length of 128 bits and purportedly offering the same security level as 3DES-O2, which is at the moment the most adopted variant of 3DES.

Since NIST assesses the security level of submissions for post-quantum asymmetric cryptography and post-quantum signatures by referencing the quantum implementation of AES, recent literature proposals have predominantly targeted AES, which, like DES, is based on a Feistel-network structure. The best-known quantum attack on AES, as outlined in [38], serves as a benchmark for NIST evaluations. It is worth noting that the preprint updated version corrects the inaccuracies present in the published version [3] due to programming framework errors.

In [8], the authors present low-depth and low-width quantum circuits for Camellia, a cipher operating on a 128-bit block with an 18-round Feistel structure. Although only the encryption complexity measures are provided, we extrapolate the overall Grover-based key-attack complexity using similar assumptions as our work. The study by [9] presents instead a quantum key-recovery attack for SEED, a Feistel-based cipher with a 128-bit block size. SEED's round function involves XOR gates, modular additions, and S-boxes, with exponentiation in a finite field. Finally, in [7], the authors analyze the HIGHT cipher, a Korean block cipher with a 64-bit block size, based on a modular Addition-Rotation-XOR (ARX) architecture.

Comparing these results reveals that 3DES-O2, the most commonly used cipher in the DES family, requires significantly less effort to compromise compared to ciphers with the same key-size. According to NIST's criteria for evaluating cryptographic proposals based on AES computation complexity, 3DES-O2 fails to meet the threshold required for post-quantum security. The same holds true for 3DES-O3 with the MITM strategy, for which both square-root and cubic-root access costs result in computational complexity below the threshold. Finally, DES cannot be deemed secure against quantum attacks, echoing its vulnerability in classical scenarios.

V. Conclusion

In this study, we conducted an in-depth analysis of a Grover-based key-recovery attack targeting DES and 3DES, which continue to be widely utilized for data security. Our approach leverages the bitslicing technique to construct a reversible implementation of the non-linear S-boxes, a novel exploration in the literature to the best of our knowledge. Our findings reveal that 3DES with keying option 2, the most prevalent option, is more susceptible to quantum attacks compared to other ciphers purportedly offering equivalent security levels. Specifically, it fails to reach the minimum security level demanded by NIST for post-quantum security, since, compared to the AES proposal used as a reference, it requires a computational complexity lower by factors ranging from 2^{11} to 2^{17}. Additionally, 3DES with keying option 3, when combined with QRAM to facilitate a Meet-In-The-Middle strategy, exhibits a similar vulnerability.

Looking ahead, this work can lay the starting point for several future researches. Firstly, delving into the security resilience of DES and 3DES within the Q2 model [17] could provide valuable insights. Additionally, investigating the susceptibility of these ciphers to quantum differential and linear cryptanalysis attacks [42] warrants attention. Furthermore, extending our research to encompass all ciphers standardized in the ISO/IEC suite would offer a comprehensive assessment of their respective security strength.

Acknowledgment

This work has been partially supported by both ICSC — "National Research Center in High Performance Computing, Big Data and Quantum Computing" and project SERICS (PE000014) under the Italian NRRP MUR program funded by the EU — NGEU.

References

[1] P. W. Shor, "Algorithms for quantum computation: Discrete logarithms and factoring," in *35th Annual Symposium on Foundations of Computer Science, Santa Fe, New Mexico, USA, 20-22 November 1994*, IEEE Computer Society, 1994. DOI: 10.1109/SFCS.1994.365700.

[2] Z. Huang and S. Sun, "Synthesizing quantum circuits of AES with lower T-depth and less qubits," in *Advances in Cryptology - ASIACRYPT 2022 - 28th Intl. Conf. on the Theory and Application of Cryptology and Information Security, Taipei, Taiwan, December 5-9, 2022, Proc., Part III*, ser. Lecture Notes in Computer Science, vol. 13793, Springer, 2022. DOI: 10.1007/978-3-031-22969-5_21.

[3] S. Jaques, M. Naehrig, M. Roetteler, and F. Virdia, "Implementing grover oracles for quantum key search on AES and LowMC," in *Advances in Cryptology - EUROCRYPT 2020 - 39th Annual Intl. Conf. on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proc., Part II*, ser. Lecture Notes in Computer Science, vol. 12106, Springer, 2020. DOI: 10.1007/978-3-030-45724-2_10.

[4] J. Zou, Z. Wei, S. Sun, X. Liu, and W. Wu, "Quantum Circuit Implementations of AES with Fewer Qubits," in *Advances in Cryptology - ASIACRYPT 2020 - 26th Intl. Conf. on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7-11, 2020, Proc., Part II*, ser. Lecture Notes in Computer Science, vol. 12492, Springer, 2020. DOI: 10.1007/978-3-030-64834-3_24.

[5] "Submission Requirements and Evaluation Criteria for the Post-Quantum Cryptography Standardization Process," National Institute of Standards and Technology. (2016), [Online]. Available: https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf.

[6] "Call for additional digital signature schemes for the post-quantum cryptography standardization process," National Institute of Standards and Technology. (2022), [Online]. Available: https://csrc.nist.gov/csrc/media/Projects/pqc-dig-sig/documents/call-for-proposals-dig-sig-sept-2022.pdf.

[7] K. Jang, G. Song, H. Kim, H. Kwon, H. Kim, and H. Seo, "Parallel quantum addition for Korean block ciphers," *Quantum Inf. Process.*, vol. 21, no. 11, 2022. DOI: 10.1007/S11128-022-03714-3.

[8] D. Lin, B. Sun, Z. Xiang, J. Zou, and Y. Guo, "Further insights on constructing quantum circuits for Camellia block cipher," *Quantum Inf. Process.*, vol. 22, no. 12, 2023. DOI: 10.1007/S11128-023-04182-Z.

[9] Y. Oh, K. Jang, Y. Yang, and H. Seo, *Optimized quantum implementation of SEED*, Cryptology ePrint Archive, Paper 2023/1566, 2023. [Online]. Available: https://eprint.iacr.org/2023/1566.

[10] D. Lin, Z. Xiang, R. Xu, X. Zeng, and S. Zhang, "Quantum circuit implementations of SM4 block cipher based on different gate sets," *Quantum Inf. Process.*, vol. 22, no. 7, 2023. DOI: 10.1007/S11128-023-04002-4.

[11] H. Kuwakado and M. Morii, "Quantum distinguisher between the 3-round Feistel cipher and the random permutation," in *IEEE Intl. Symposium on Information Theory, ISIT 2010, June 13-18, 2010, Austin, Texas, USA, Proc.*, IEEE, 2010. DOI: 10.1109/ISIT.2010.5513654.

[12] H. Kuwakado and M. Morii, "Security on the quantum-type Even-Mansour cipher," in *Proc. of the Intl. Symposium on Information Theory and Its Applications, ISITA 2012, Honolulu, HI, USA, October 28-31, 2012*, IEEE, 2012. [Online]. Available: https://ieeexplore.ieee.org/document/6400943/.

[13] M. Kaplan, G. Leurent, A. Leverrier, and M. Naya-Plasencia, "Breaking symmetric cryptosystems using quantum period finding," in *Advances in Cryptology - CRYPTO 2016 - 36th Annual Intl. Cryptology Conf., Santa Barbara, CA, USA, August 14-18, 2016, Proc., Part II*, ser. Lecture Notes in Computer Science, vol. 9815, Springer, 2016. DOI: 10.1007/978-3-662-53008-5_8.

[14] G. Leander and A. May, "Grover meets simon - quantumly attacking the FX-construction," in *Advances in Cryptology - ASIACRYPT 2017 - 23rd Intl. Conf. on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proc., Part II*, ser. Lecture Notes in Computer Science, vol. 10625, Springer, 2017. DOI: 10.1007/978-3-319-70697-9_6.

[15] B. Ni, G. Ito, X. Dong, and T. Iwata, "Quantum attacks against type-1 generalized feistel ciphers and applications to CAST-256," in *Progress in Cryptology – INDOCRYPT 2019*, Cham: Springer, 2019. DOI: 10.1007/978-3-030-35423-7_22.

[16] G. Ito, A. Hosoyamada, R. Matsumoto, Y. Sasaki, and T. Iwata, "Quantum chosen-ciphertext attacks against feistel ciphers," in *Topics in Cryptology - CT-RSA 2019 - the Cryptographers' Track at the RSA Conf. 2019, San Francisco, CA, USA, March 4-8, 2019, Proc.*, ser. Lecture Notes in Computer Science, vol. 11405, Springer, 2019. DOI: 10.1007/978-3-030-12612-4_20.

[17] X. Bonnetain, A. Hosoyamada, M. Naya-Plasencia, Y. Sasaki, and A. Schrottenloher, "Quantum attacks without superposition queries: The offline simon's algorithm," in *Advances in Cryptology - ASIACRYPT 2019 - 25th Intl. Conf. on the Theory and Application of Cryptology and Information Security, Kobe, Japan, December 8-12, 2019, Proc., Part I*, ser. Lecture Notes in Computer Science, vol. 11921, Springer, 2019. DOI: 10.1007/978-3-030-34578-5_20.

[18] D. R. Simon, "On the power of quantum computation," *SIAM J. Comput.*, vol. 26, no. 5, 1997. DOI: 10.1137/S0097539796298637.

[19] M. Almazrooie, A. Samsudin, R. Abdullah, and K. N. Mutter, "Quantum exhaustive key search with simplified-DES as a case study," *SpringerPlus*, vol. 5, no. 1, 2016. DOI: 10.1186/s40064-016-3159-4.

[20] D. V. Denisenko and M. V. Nikitenkova, "Application of Grover's Quantum Algorithm for SDES Key Searching," *J. Exp. Theor. Phys.*, vol. 128, no. 1, Jan. 1, 2019. DOI: 10.1134/S1063776118120142.

[21] S. Jaques and J. M. Schanck, "Quantum cryptanalysis in the RAM model: Claw-finding attacks on SIKE," in *Advances in Cryptology - CRYPTO 2019 - 39th Annual Intl. Cryptology Conf., Santa Barbara, CA, USA, August 18-22, 2019, Proc., Part I*, ser. Lecture Notes in Computer Science, vol. 11692, Springer, 2019. DOI: 10.1007/978-3-030-26948-7_2.

[22] S. Perriello, A. Barenghi, and G. Pelosi, "Improving the efficiency of quantum circuits for information set decoding," *ACM Trans. Quantum Comput.*, vol. 4, no. 4, 2023. DOI: 10.1145/3607256.

[23] S. Jaques and A. G. Rattew, *QRAM: A survey and critique*, 2023. arXiv: 2305.10310 [quant-ph].

[24] C. E. Shannon, "Communication theory of secrecy systems," *Bell System Technical Journal*, vol. 28, no. 4, 1949. DOI: 10.1002/J.1538-7305.1949.TB00928.X.

[25] V. A. Dasu, A. Baksi, S. Sarkar, and A. Chattopadhyay, "LIGHTER-R: Optimized reversible circuit implementation for SBoxes," in *32nd IEEE Intl. System-on-Chip Conf., SOCC 2019, Singapore, September 3-6, 2019*, IEEE, 2019. DOI: 10.1109/SOCC46988.2019.1570548320.

[26] Z. Bao, J. Guo, S. Ling, and Y. Sasaki, "Peigen–a platform for evaluation, implementation, and generation of s-boxes," *IACR Trans. on Symmetric Cryptol.*, pp. 330–394, 2019. DOI: 10.13154/tosc.v2019.i1.330-394.

[27] J. Jean, T. Peyrin, S. M. Sim, and J. Tourteaux, "Optimizing implementations of lightweight building blocks," *IACR Trans. on Symmetric Cryptol.*, pp. 130–168, 2017. DOI: 10.13154/tosc.v2017.i4.130-168.

[28] B. Bilgin, L. De Meyer, S. Duval, I. Levi, and F.-X. Standaert, "Low AND depth and efficient inverses: A guide on s-boxes for low-latency masking," *IACR Trans. on Symmetric Cryptol.*, vol. 2020, no. 1, 2020. DOI: 10.13154/tosc.v2020.i1.144-184.

[29] K. Stoffelen, "Optimizing S-box implementations for several criteria using SAT solvers," in *Fast Software Encryption*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2016.

[30] E. Biham, "A fast new DES implementation in software," in *Fast Software Encryption, 4th Intl. Workshop, FSE '97, Haifa, Israel, January 20-22, 1997, Proc.*, ser. Lecture Notes in Computer Science, vol. 1267, Springer, 1997. DOI: 10.1007/BFB0052352.

[31] M. Kwan, *Reducing the gate count of bitslice DES*, 2000. [Online]. Available: http://eprint.iacr.org/2000/051.

[32] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proc. of the Twenty-Eighth Annual {ACM} Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996*, ACM, 1996. DOI: 10.1145/237814.237866.

[33] G. Lancellotti, S. Perriello, A. Barenghi, and G. Pelosi, "Design of a quantum walk circuit to solve the subset-sum problem," in *61st ACM/IEEE Design Automation Conf., DAC 2024, San Francisco, CA, USA, July 23-27, 2024*, ACM, 2024. DOI: 10.1145/3649329.3657337.

[34] M. Boyer, G. Brassard, P. Høyer, and A. Tapp, "Tight Bounds on Quantum Searching," *Fortschritte der Physik: Progress of Physics*, vol. 46, no. 4-5, 1998. DOI: 10.1002/(SICI)1521-3978(199806)46:4/5<493::AID-PROP493>3.0.CO;2-P.

[35] P. Kim, D. Han, and K. C. Jeong, "Time-space complexity of quantum search algorithms in symmetric cryptanalysis: Applying to AES and SHA-2," *Quantum Inf. Process.*, vol. 17, no. 12, 2018. DOI: 10.1007/s11128-018-2107-3.

[36] S. Perriello, A. Barenghi, and G. Pelosi, "Quantum Circuit Design for the Lee-Brickell based Information Set Decoding," in *Applied Cryptography and Network Security Workshops - ACNS 2024 Satellite Workshops, ACNS*, ser. Lecture Notes in Computer Science, Abu Dhabi, UAE: Springer, 2024.

[37] C. Gidney, "Halving the cost of quantum addition," *Quantum*, vol. 2, Jun. 2018. DOI: 10.22331/q-2018-06-18-74.

[38] S. Jaques, M. Naehrig, M. Roetteler, and F. Virdia, *Implementing Grover oracles for quantum key search on AES and LowMC*, 2019. arXiv: 1910.01700 [quant-ph]. [Online]. Available: https://eprint.iacr.org/2019/1146.pdf.

[39] P. O. Boykin, T. Mor, M. Pulver, V. P. Roychowdhury, and F. Vatan, "A new universal and fault-tolerant quantum basis," *Inf. Process. Lett.*, vol. 75, no. 3, 2000. [Online]. Available: https://doi.org/10.1016/S0020-0190(00)00084-3.

[40] C. Jones, "Low-overhead constructions for the fault-tolerant Toffoli gate," *Phys. Rev. A*, vol. 87, no. 2, 2013.

[41] N. M. Linke, D. Maslov, M. Roetteler, *et al.*, "Experimental comparison of two quantum computing architectures," *Proc. Natl. Acad. Sci.*, vol. 114, no. 13, 2017. DOI: 10.1073/pnas.1618020114.

[42] M. Kaplan, G. Leurent, A. Leverrier, and M. Naya-Plasencia, "Quantum differential and linear cryptanalysis," *IACR Trans. Symmetric Cryptol.*, vol. 2016, no. 1, 2016. DOI: 10.13154/TOSC.V2016.I1.71-94.