A Novel Framework to Construct Quantum Circuits of S-Boxes: Application to 4-bit S-Boxes

Yongjin Jeon, Seungjun Baek, and Jongsung Kim*

Kookmin University, Seoul, Korea
{idealtop18, hellosj3, jskim}@kookmin.ac.kr

Abstract

In this paper, we present a new framework for constructing quantum circuits of S-boxes. We model the quantum circuits of S-boxes using two layers: Toffoli and linear layers. We generate vector spaces based on the values of qubits used in the linear layers and apply them to find quantum circuits. The proposed framework finds the quantum circuit by matching elements of vector spaces generated from the input and output of a given S-box using the meet-in-the-middle strategy. We developed a tool to apply this framework to 4-bit S-boxes. While the 4-bit S-box quantum circuit construction tool `LIGHTER-R` only finds circuits that can be implemented with 4 qubits, the proposed tool additionally achieves the circuits with 5 qubits. This tool can find quantum circuits of 4-bit odd permutations based on the CNOT, NOT, and Toffoli gates, whereas `LIGHTER-R` cannot perform this task in the same environment. We expect that this technique can become a critical step toward finding optimized S-box quantum circuits.

Keywords: Mobile internet security, Quantum Circuit, S-box, Toffoli-depth

1 Introduction

Quantum computers speed up many algorithms based on the superposition principle of quantum mechanics, presenting considerable influence on mobile internet security and cryptography. Shor's algorithm [1] exponentially reduces the complexity of attacking public-key schemes on quantum computers. Since 2016, the National Institute of Standards and Technology (NIST) has been conducting the post-quantum cryptography standardization process [2]. For symmetric-key schemes, Grover's algorithm [3] and Simon's algorithm [4] display significant performance to attack schemes, but these algorithms do not wholly compromise the security of such systems. However, in a quantum computing environment, symmetric-key cryptography may have weak properties not yet studied for each algorithm. Unknown quantum attacks using them may exist; this, in-depth research should be conducted.

The S-box is a crucial component that provides confusion in symmetric-key schemes. When implementing a cipher as a quantum circuit, the linear layer can be implemented with only NOT and controlled-NOT (CNOT) gates. However, highly structured nonlinear layers, such as the S-box, must employ relatively expensive Toffoli gates and numerous qubits. In quantum circuits for symmetric-key schemes, the S-box incurs the greatest cost.

The complexity of a quantum circuit is evaluated by the number of qubits and the Toffoli-depth defined by the number of nonparallelizable Toffoli gates. Optimizing these two parameters increases the implementation efficiency of quantum computers, improving the ability of attackers to perform quantum exhaustive searches and dedicated attacks using Grover's algorithm. Hence,

The 7th International Conference on Mobile Internet Security (MobiSec'23), Dec. 19-21, 2023, Okinawa, Japan, Article No.12

*Corresponding author: Department of Information Security Cryptography Mathematics, Kookmin University, Seoul, 02707, Republic of Korea, Tel: +82-02-910-5126 optimizing the quantum circuits of S-boxes is crucial for assessing the security of symmetric-key schemes against quantum computer-based attacks.

Extensive recent research has been conducted on finding efficient quantum circuits for the Advanced Encryption Standard (AES). Grassl et al. [5] initially proposed a quantum circuit for AES and introduced a zig-zag structure to reduce the number of qubits required for implementation. Subsequently, several studies have been constructed to reduce the number of qubits to implement AES [6, 7, 8, 9]. However, in NIST's post-quantum cryptography standardization process, the circuit's Toffoli-depth represents a crucial parameter. In response, Jaques et al. [10] attempted to construct an AES quantum circuit with a shallow Toffoli-depth. Recently, Huang and Sun [9] proposed an AES quantum circuit with the shallowest depth.

Contributions. This paper provides a new framework for constructing quantum circuits $\mathcal{C} : |\alpha\rangle \to |S(\alpha)\rangle$ of S-boxes. We treat the quantum circuits of S-boxes using only CNOT, NOT, and Toffoli gates, called CNT-circuits. Using the CNT-circuits, we provide a framework for finding the quantum circuits of the S-boxes with low Toffoli-depths according to the limited number of qubits by matching elements of vector spaces generated from the inputs and outputs of the S-boxes. The framework employs a meet-in-the-middle strategy. The key is to analyze the vector spaces spanned by the values before and after the Toffoli layers. The proposed framework provides a specialized search on Toffoli-depth by ignoring the detailed implementations of linear layers. To the best of our knowledge, no study has analyzed the Toffoli-depth and number of qubits through the vector space and the basis analysis between Toffoli layers in quantum circuits of S-boxes.

To verify the effectiveness of the proposed framework, we propose a technique and tool for applying the framework to a 4-bit S-box. These components are currently used as essential elements in many AEAD schemes and block ciphers [11, 12, 13, 14, 15]. Finding quantum circuits of S-boxes supports an accurate analysis of the quantum quantitative complexity of target ciphers. The technique involves two algorithms to apply to a 4-bit S-box, leading to one feature that is missing in existing algorithms for CNT-circuits. `LIGHTER-R` [16] provides Toffoli-depth optimized quantum circuits of 4-bit S-boxes with a 4-qubit restriction. However, this approach fails if the target 4-bit S-boxes are odd permutations, which occurs due to the theorem that odd permutation cannot be implemented with 4 qubits in CNT-circuits and requires at least 5 qubits [17]. The proposed algorithms offer a more comprehensive range of quantum circuits compared to `LIGHTER-R` in terms of the Toffoli-depth and number of qubits (up to 5). This improvement allows the algorithms to produce the quantum circuits of the 4-bit S-boxes, an odd permutation beyond the capability of `LIGHTER-R`. Given that half of the 4-bit S-boxes are odd permutations, this result allows researchers to implement quantum circuits for all 4-bit S-boxes.

Paper Organization. Section 2 describes the quantum computation and quantum circuits. Section 3 defines the CNT-circuit and describes its properties. Section 4 describes the proposed framework for finding quantum circuits of 4-bit S-boxes according to a limited number of qubits and presents an example. Section 5 presents the conclusions.

2 Quantum Computation and Quantum Circuits

A fundamental concept in classical computing involves the bit, characterized as either 0 or 1. Conversely, a qubit plays the role of a bit in quantum computing, holding 0 and 1 at the same time according to the superposition principle of quantum mechanics. The values $|0\rangle$ and $|1\rangle$ are orthonormal bases of a two-dimensional Hilbert space, also called the computational basis. The superposition state of a qubit can be represented as $\alpha|0\rangle+\beta|1\rangle$ $(\alpha, \beta \in \mathbb{C})$, and such α and β are the complex probability amplitude. The qubit's state is destroyed by measurement, after which one can observe $|0\rangle$ or $|1\rangle$, with the respective probabilities of $|\alpha|^2$ and $|\beta|^2$ (thus, $|\alpha|^2 + |\beta|^2 = 1$ holds). To describe n qubits, we require a 2^n-dimensional Hilbert space for which the orthonormal bases are $|00\cdots0\rangle$, $|00\cdots1\rangle$, ..., $|11\cdots1\rangle$, and total 2^n.

This work is primarily concerned with quantum circuits consisting of CNOT, NOT and Toffoli gates. A $CNOT$ gate is the two-qubit CNOT gate defined by $CNOT : |a\rangle|b\rangle \mapsto |a\rangle|b \oplus a\rangle$, and a NOT gate is the single-qubit gate defined by $NOT : |a\rangle \mapsto |a \oplus 1\rangle$. A $Toffoli$ gate is the three-qubit CNOT gate defined by $Toffoli : |a\rangle|b\rangle|c\rangle \mapsto |a\rangle|b\rangle|c \oplus ab\rangle$. A $Toffoli$ gate can manage the XOR and AND of classical gates at once. These quantum gates are presented in Figure 1.

A quantum circuit using only CNOT, NOT, and Toffoli gates is defined as a CNT-circuit. In the CNT-circuit, the NOT gates can be moved to the circuit's last operation without changing the Toffoli-depth using the properties of Figure 2. The NOT gates gathered in the last operation are equivalent to using an XORing with a constant value in the S-box. As all S-boxes can be implemented with CNT-circuits, CT-circuits (without NOT gates) can implement all S-boxes satisfying $0 \mapsto 0$ [17].

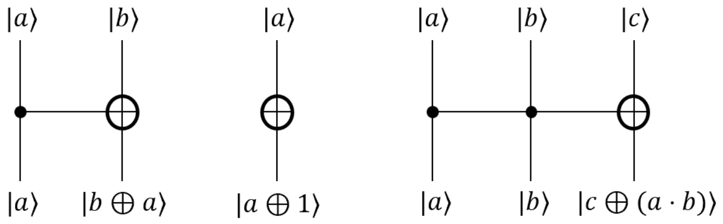

Figure 1: CNOT(left), NOT(middle), and Toffoli(right) gates

3 Modeling Quantum Circuits of S-Boxes

We consider the n-bit S-box defined by a vectorial Boolean function $\mathbb{F}^n \rightarrow \mathbb{F}^n$. In the S-box circuit, n Boolean coordinate functions are represented by various wires. Each wire connects to n input bits, and additional wires may be required depending on the circuit. These wires become qubits in a quantum circuit.

We modeled CT-circuits for $\mathcal{C} : |\alpha\rangle|0\rangle \rightarrow |S(\alpha)\rangle|0\rangle$ of S-boxes satisfying $0 \mapsto 0$. We let $\mathcal{C}$ use q qubits and have a Toffoli-depth of t. We define layers with only Toffoli gates as Toffoli layers and treat the layers between them as linear layers (including empty layers). In addition, $\mathcal{C}$ has $t+1$ linear layers, including two the outermost linear layers. We establish the indices of the layers as represented in Equation (1). The CNOT gates can be implemented without additional qubits [18], and their cost is exempt from the analysis model. Therefore, we omit the detailed implementation of the CNOT gates in the linear layer:

$$\mathcal{C} : L_t \circ T_t \circ L_{t-1} \circ T_{t-1} \circ \cdots \circ L_1 \circ T_1 \circ L_0. \tag{1}$$

To facilitate finding the circuit, we arrange Toffoli gates in order within the Toffoli layers. We assume that the control and target qubit positions of the Toffoli gates are fixed, and the

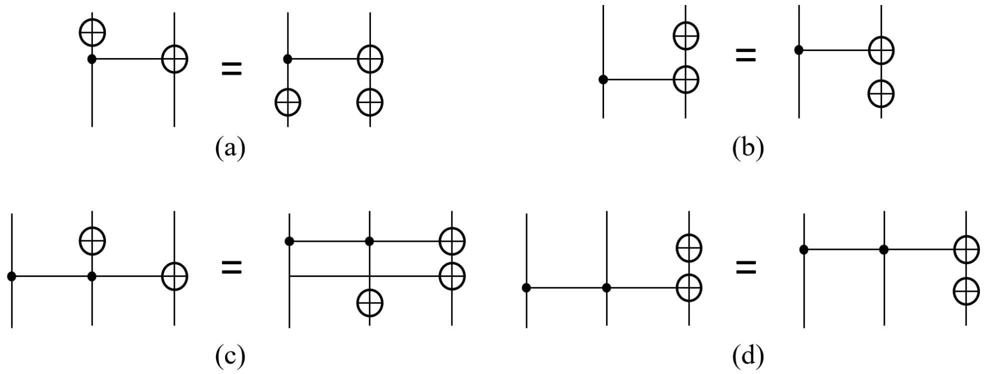
(a) (b)
(c) (d)

Figure 2: Properties of NOT gates

exchange of wires that occurs while fixing them is absorbed by the linear layers. In detail, the control qubits of the i-th Toffoli gate use the $(3i-2)$-th and $(3i-1)$-th qubits, and the $3i$-th qubit serves as the target qubit. Afterward, Toffoli gates are arranged consecutively in the Toffoli layers, and the layers can be implemented based on the number of Toffoli gates (See Figure 3).

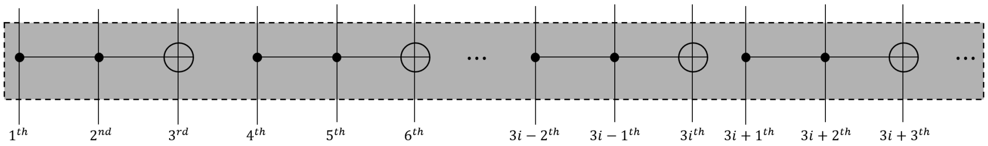

Figure 3: Toffoli layer in our model

Implementing linear layers is equivalent to knowing the input and output values of the linear layers. When Toffoli layers are implemented, the input values of the next linear layer can be determined through the output values of the previous linear layer. If the output values of t linear layers are known, the entire circuit can be implemented.

The input and output of the linear layer are represented by the qubit values at each point. We treat the qubit values as Boolean functions and consider the vector space they span. The vector spaces spanned at the input and output points of the linear layer are identical. We define the vector space generated by L_i as $\mathcal{X}_i$, and each $\mathcal{X}_i$ is transformed into $\mathcal{X}_{i+1}$ by T_{i+1} corresponding to the $(i+1)$-th Toffoli layer.

$$span(x_1, \cdots, x_n) = \mathcal{X}_0 \xrightarrow{T_1} \mathcal{X}_1 \xrightarrow{T_2} \cdots \xrightarrow{T_t} \mathcal{X}_t = span(y_1, \cdots, y_n).$$

4 Finding Quantum Circuits of 4-bit S-Boxes Using Up to 5 Qubits

The vector space $\mathcal{X}_i$ is spanned by the qubit values for the input and output of L_i. Of the three qubits included in one Toffoli gate, only the target qubit changes in value. The intersection of two consecutive spaces, $\mathcal{X}_i$ and $\mathcal{X}_{i+1}$, is nonempty because qubit values can be invariant. The intersections between spaces spanned in a circuit with the minimum Toffoli-depth increase as the indices of the spaces become closer. This logic is generalized in Theorem 1.

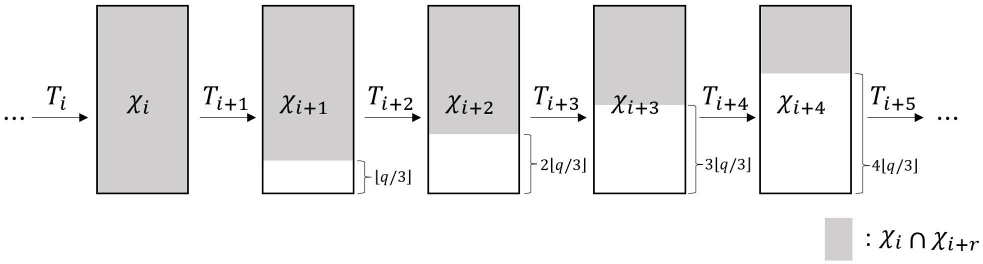

Figure 4: Description of Theorem 1

Theorem 1. *If the circuit uses q qubits, for any $i \leq t - r$,*

$$dim(\mathcal{X}_i) - dim(\mathcal{X}_i \cap \mathcal{X}_{i+r}) \leq r\lfloor q/3 \rfloor,$$
$$dim(\mathcal{X}_{i+r}) - dim(\mathcal{X}_i \cap \mathcal{X}_{i+r}) \leq r\lfloor q/3 \rfloor.$$

Proof. There are r Toffoli layers between the points of $\mathcal{X}_i$ and $\mathcal{X}_{i+r}$. A maximum of $\lfloor q/3 \rfloor$ Toffoli gates can be used in one Toffoli layer; thus, at most $r\lfloor q/3 \rfloor$ values are not in $\mathcal{X}_i$. Therefore, we obtain $dim(\mathcal{X}_i) - dim(\mathcal{X}_i \cap \mathcal{X}_{i+r}) \leq r\lfloor q/3 \rfloor$. The lower equation uses a similar process. □

According to the above theorem, we obtain the lower bound of the Toffoli-depth as follows for the quantum circuit $|\alpha\rangle \rightarrow |S(\alpha)\rangle$.

Corollary 2. *We let $\mathcal{X}_0$ be the set of all n-variable Boolean linear functions (including the zero function), and $\mathcal{X}_t$ be the set of all component functions of the S-box S (including the zero function). Then, the quantum circuit's Toffoli-depth is $\mathcal{C} : |\alpha\rangle \rightarrow |S(\alpha)\rangle$ of n-bit S using q qubits is $(n - dim(\mathcal{X}_0 \cap \mathcal{X}_t))/\lfloor q/3 \rfloor$ or greater.*

Proof. In this special case, $i = 0$ and $r = t$ in Theorem 1. The proof is as follows:

$$\begin{aligned} dim(\mathcal{X}_0) - dim(\mathcal{X}_0 \cap \mathcal{X}_t) &\leq t\lfloor q/3 \rfloor, \\ n - dim(\mathcal{X}_0 \cap \mathcal{X}_t) &\leq t\lfloor q/3 \rfloor, \\ (n - dim(\mathcal{X}_0 \cap \mathcal{X}_t))/\lfloor q/3 \rfloor &\leq t. \end{aligned}$$

□

4.1 Meet-in-the-middle Strategy

Quantum circuits comprise reversible gates; thus, the implementation of these circuits is found in both the forward and backward directions:

$$\textbf{forward} : \mathcal{X}_0 \rightarrow \mathcal{X}_1 \rightarrow \cdots \rightarrow \mathcal{X}_t,$$
$$\textbf{backward} : \mathcal{X}_t \rightarrow \mathcal{X}_{t-1} \rightarrow \cdots \rightarrow \mathcal{X}_0.$$

The proposed algorithms confirm how many values of the newly constructed vector space belong to the opposite vector space. For example, we consider that $\mathcal{X}_i$ and $\mathcal{X}_{t-j}$ are obtained by implementing up to the i-th Toffoli layer in the forward direction and the j-th Toffoli layer in the backward direction. The proposed algorithms select $\mathcal{X}_{i+1}$ to implement the $(i+1)$-th Toffoli layer in the forward direction, yielding the largest intersection with $\mathcal{X}_{t-j}$. In this case, $|\mathcal{X}_i \cap \mathcal{X}_{t-j}| < |\mathcal{X}_{i+1} \cap \mathcal{X}_{t-j}|$ holds, and if $\mathcal{X}_{i+1} = \mathcal{X}_{t-j}$, the circuit is completely implemented.

We describe the process of implementing the 4-bit S-box $S(x_1, x_2, x_3, x_4) = (y_1, y_2, y_3, y_4)$ using fewer than 5 qubits. Only one Toffoli gate exists in each Toffoli layer. We let $\mathfrak{P}_0 = \mathit{span}(x_1, x_2, x_3, x_4)$ and $\mathfrak{P}_t = \mathit{span}(y_1, y_2, y_3, y_4)$. If $\mathfrak{P}_0 = \mathfrak{P}_t$ holds, S is a linear function, so the Toffoli-depth is 0. The linear function can be implemented without ancilla qubits (i.e., with 4 qubits).

The proposed algorithms take $(\mathcal{X}_i, \mathcal{X}_{t-j})$ as input, and $(\mathcal{X}_{i+1}, \mathcal{Y})$ as output, which are closer spaces than $(\mathcal{X}_i, \mathcal{X}_{t-j})$. Additionally, $\mathcal{Y}$ can be either $\mathcal{X}_{t-j}$ or $\mathcal{X}_{t-j-1}$. The following implies that $(\mathcal{X}_{i+1}, \mathcal{Y})$ is closer than $(\mathcal{X}_i, \mathcal{X}_{t-j})$:

- $|\mathcal{X}_{i+1} = \mathcal{Y}|$ holds.
- If $|\mathcal{X}_{i+1} \neq \mathcal{Y}|$, then $|\mathcal{X}_{i+1} \cap \mathcal{Y}| > |\mathcal{X}_i \cap \mathcal{X}_{t-j}|$.

Algorithm 1 finds only the forward direction, and Algorithm 2 discovers both directions based on the meet-in-the-middle strategy. First, we describe Algorithm 1. Suppose we generate a set $F_{\rightarrow}$ that collects the target qubit values that the Toffoli gates in T_{i+1} can have. When $2^q = |\mathcal{X}_i|$, the set of all qubits' values at the input point of T_{i+1} becomes the basis of T_{i+1}, such that the elements in the set are independent of each other.

$$F_{\rightarrow} = \{ab \oplus c | a, b(\neq a), c \in \mathcal{X}_i, \text{if } 2^q = |\mathcal{X}_i|, \text{ then } a, b, \text{and } c \text{ are linearly independent}\}.$$

There can be several combinations of $a, b, c \in \mathcal{X}_i$ that satisfy $p = ab \oplus c$ for element p of $F_{\rightarrow}$. For each p, a, b, and c, we can construct a basis B_i of $\mathcal{X}_i$. In addition, B_i must be constructed so that a and b belong to it, and if c is linearly independent of $\{a, b\}$, we adjust c to belong to B_i as well. We let d and e be linearly independent of $\{a, b, c\}$ and $\{a, b, c, d\}$, respectively, and B_i is possible in the following cases:

1. When $|\mathcal{X}_i| = 32$, $B_i = \{a, b, c, d, e\}$.
2. When $|\mathcal{X}_i| = 16$ holds and a, b, and c are linearly dependent, $B_i = \{a, b, d, e\}$.
3. When $|\mathcal{X}_i| = 16$ holds and a, b, and c are linearly independent, $B_i = \{a, b, c, d\}$.

Algorithm 1: Forward finding for quantum circuits

```
input  : $\mathcal{X}_i$, $\mathcal{X}_{t-j}$, $n$, $q$
output: a set $\mathcal{D}$ of pairs $(B_i, \mathcal{X}_{i+1}, \mathcal{X}_{t-j})$
$\mathcal{D} \leftarrow \{\}$
$F_{\rightarrow} \leftarrow \{ab \oplus c | a, b(\neq a), c \in \mathcal{X}_i$, if $2^q = |\mathcal{X}_i|$, then $a, b, c$ are linearly independent$\}$
for $p \in \mathcal{F}_{\rightarrow}$ do
    for $a, b, c \in \mathcal{X}_i$ do
        for each case of $B_i$ and $B_{i+1}$ do
            $\mathcal{X}_{i+1} \leftarrow span(B_{i+1})$
            if $(\mathcal{X}_{i+1}, \mathcal{X}_{t-j})$ is closer than $(\mathcal{X}_i, \mathcal{X}_{t-j})$ then
                $\mathcal{D} \leftarrow \mathcal{D} \cup \{(B_{i+1}, \mathcal{X}_{i+1}, \mathcal{X}_{t-j})\}$
return $\mathcal{D}$
```

Algorithm 2: Meet-in-the-middle finding for quantum circuits

```
input  : $\mathcal{X}_i$, $\mathcal{X}_{t-j}$, $n$, $q$
output: a set $\mathcal{D}$ of pairs $(B_{i+1}, B_{t-j-1}, \mathcal{X}_{i+1}, \mathcal{X}_{t-j-1})$
$\mathcal{D} \leftarrow \{\}$
$F_{\rightarrow} \leftarrow \{ab \oplus c | a, b(\neq a), c \in \mathcal{X}_i$, if $2^q = |\mathcal{X}_i|$, then $a, b, c$ are linearly independent$\}$
$F_{\leftarrow} \leftarrow \{\alpha\beta \oplus \gamma | \alpha, \beta(\neq \alpha), \gamma \in \mathcal{X}_{t-j}$, if $2^q = |\mathcal{X}_{t-j}|$, then $\alpha, \beta, \gamma$ are linearly independent$\}$
for $p \in \mathcal{F}_{\rightarrow}$ do
    for $a, b, c \in \mathcal{X}_i$ do
        for each case of $B_i$ and $B_{i+1}$ do
            $\mathcal{X}_{i+1} \leftarrow span(B_{i+1})$
            for $\rho \in \mathcal{F}_{\leftarrow}$ do
                for $\alpha, \beta, \gamma \in \mathcal{X}_{t-j}$ do
                    for each case of $B_{t-j}$ and $B_{t-j-1}$ do
                        $\mathcal{X}_{t-j-1} \leftarrow span(B_{t-j-1})$
                        if $(\mathcal{X}_{i+1}, \mathcal{X}_{t-j-1})$ is closer than $(\mathcal{X}_i, \mathcal{X}_{t-j})$ then
                            $\mathcal{D} \leftarrow$ set $\cup \{(B_{i+1}, B_{t-j-1}, \mathcal{X}_{i+1}, \mathcal{X}_{t-j-1})\}$
return $\mathcal{D}$
```

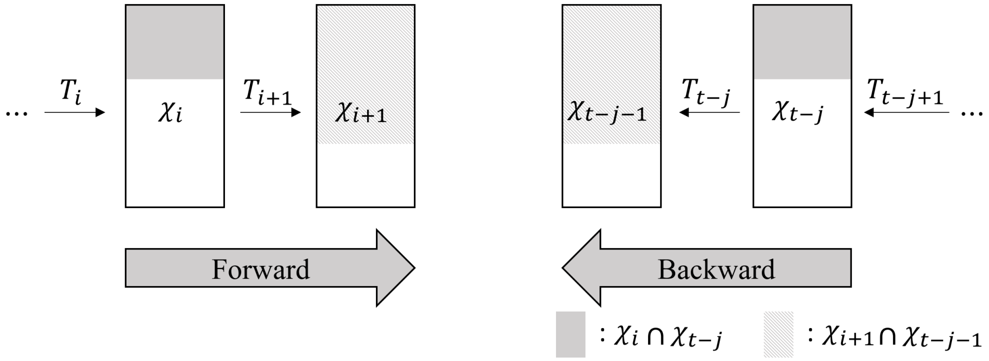

Figure 5: Description of the meet-in-the-middle strategy

We let B_{i+1} be the basis of $\mathcal{X}_{i+1}$ to be generated. In Case 1, c changes to p, resulting in $B_{i+1} = \{a, b, p, d, e\}$. In Case 2, c also changes to p, but because c did not become a basis, $B_{i+1} = \{a, b, p, d, e\}$ holds. In Case 3, c can change to p or p can be newly added, and B_{i+1} becomes $\{a, b, p, d\}$ or $\{a, b, p, c, d\}$. If $(\mathcal{X}_{i+1}, \mathcal{X}_{t-j})$ is closer than $(\mathcal{X}_i, \mathcal{X}_{t-j})$ for all cases, we adopt these spaces and store $(B_{i+1}, \mathcal{X}_{i+1}, \mathcal{X}_{t-j})$. After this process is performed for all p, the stored set of $(B_{i+1}, \mathcal{X}_{i+1}, \mathcal{X}_{t-j})$ becomes the output.

If Algorithm 1 outputs an empty set, we proceed with Algorithm 2. The latter generates $F_{\rightarrow}$ in the same way as Algorithm 1. Subsequently, we create a set $F_{\leftarrow}$ that collects all possible target qubit values for the Toffoli gates in T_{t-j-1}.

$$F_{\leftarrow} = \{\alpha\beta \oplus \gamma | \alpha, \beta(\neq \alpha), \gamma \in \mathcal{X}_{t-j}, \text{if } 2^q = |\mathcal{X}_{t-j}|, \text{then } \alpha, \beta, \text{and } \gamma \text{ are linearly independent}\}.$$

For each $p \in F_{\rightarrow}$ and $a, b, c \in \mathcal{X}_i$, we consider $\mathcal{X}_{i+1}$ and B_{i+1} for all cases constructed in the mentioned manner. For each $\rho \in F_{\leftarrow}$ and $\alpha, \beta, \gamma \in \mathcal{X}_{t-j}$, the same process can be taken again to obtain $\mathcal{X}_{t-j-1}$ and B_{t-j-1}. If $(\mathcal{X}_{i+1}, \mathcal{X}_{t-j-1})$ is closer than $(\mathcal{X}_i, \mathcal{X}_{t-j})$ for all cases, we adopt these spaces and store $(B_{i+1}, B_{t-j-1}, \mathcal{X}_{i+1}, \mathcal{X}_{t-j-1})$. After this process is performed for all p and ρ, the stored set $(B_{i+1}, B_{t-j-1}, \mathcal{X}_{i+1}, \mathcal{X}_{t-j-1})$ represents the output. We depict the change in the intersection due to the meet-in-the-middle strategy in Figure 5.

The process of constructing $F_{\rightarrow}$ and $F_{\leftarrow}$ is determined by the dimensions of $\mathcal{X}_i$ and $\mathcal{X}_{t-j}$, respectively. Thus, if the dimension is d, the computational cost is 2^{3d}. When employing Algorithm 2, the complexity of finding both $p \in F_{\rightarrow}$ and $\rho \in F_{\leftarrow}$ is about 2^{6d} (e.g., if $d = 5$, the complexity is 2^{30}). The memory complexity depends on how many close spaces are stored; thus, it depends on the S-box.

4.2 Results for Some 4-bit S-Boxes

We applied the proposed algorithms to various 4-bit S-boxes. First, we consider all 4-bit optimal S-boxes classified by Leander and Poschmann [19] to demonstrate the validity of the algorithms (see Table 1). In addition, `LIGHTER-R` could not find the circuits of odd permutations (i.e. $G_0, G_1, G_2, G_8, G_9, G_{13}$, and G_{14}), whereas the proposed algorithm could do so.

Table 1: Toffoli depths of optimal S-boxes using 5 qubits

Class	G_0	G_1	G_2	G_3	G_4	G_5	G_6	G_7
Toffoli-depth	4	4	4	7	5	5	7	5
Class	G_8	G_9	G_{10}	G_{11}	G_{12}	G_{13}	G_{14}	G_{15}
Toffoli-depth	4	7	7	7	7	5	7	7

Second, we consider the 4-bit S-boxes of GIFT [15], SKINNY [14], and Satrunin [20] (see Table 2). The proposed algorithms and `LIGHTER-R` output identical Toffoli-depths in the circuit implementation when using 4 qubits. We executed the proposed algorithm using 5 qubits but outputted the same Toffoli-depths. To compare the results with those of existing circuits, we checked the AND-depth, which closely relates to the Toffoli-depth. Quantum circuits with a Toffoli-depth that is the same value as the AND-depth always exist, allowing a comparison [21]. These values represented the same AND-depths of the classical implementation as claimed by the designers of GIFT and SKINNY. In the case of Saturnin, we found a more efficient circuit with an AND-depth of 5, rather than a circuit with an AND-depth of 6 that the designers found.

Figure 6 is the circuit of Saturnin's S-box σ_0 that we found. We omitted the expression of the CNOT gates in L_i. For wires going to the same output on L_i, their values are XORed.

Discussion of Results Based on the Proposed Algorithms. For each i, the algorithms take a pair $(\mathcal{X}_i, \mathcal{X}_{t-j})$ as input and select the closer pair $(\mathcal{X}_{i+1}, \mathcal{Y})$, where $\mathcal{Y}$ can be either $\mathcal{X}_{t-j}$ or $\mathcal{X}_{t-j-1}$. The value j is determined by the number of times Algorithm 2 repeats. In the process, a pair that is not closer to any i and j is never selected. This fact incurs a weakness in that the proposed algorithms sometimes fail to find circuits with a minimum

Table 2: Toffoli depths of special S-boxes using 4 qubits

cipher	Saturnin	SKINNY	GIFT
Toffoli-depth of the S-box	5	4	4

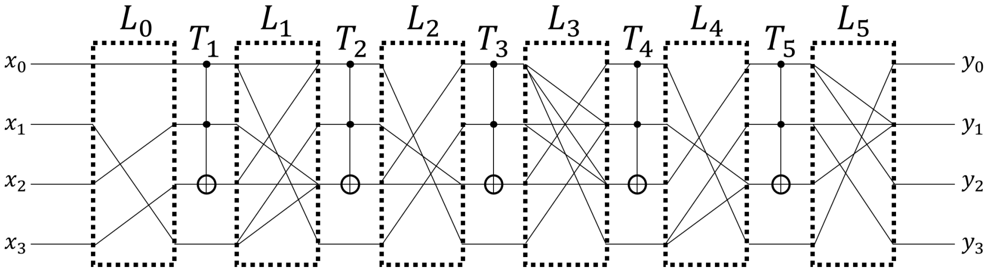

Figure 6: Circuit of Saturnin S-box σ_0

Toffoli-depth. However, we can determine the entire circuit's lower bound for the Toffoli-depth using Corollary 2. If the algorithms find a circuit with this lower bound, that implies the minimum Toffoli-depth. Furthermore, the algorithms offer the advantage of finding all circuits with such a lower bound, because if a circuit with that lower bound exists, forward finding discovers it (see Algorithm 1). If the outputted circuit does not have a lower bound, then the Toffoli-depth of the S-box is greater than the lower bound. We can confirm that the results of $G_0, G_1, G_2, G_4, G_5, G_7, G_8, G_{13}$, Saturnin, SKINNY, and GIFT provide the minimum Toffoli-depth.

5 Conclusions

This paper presents a new framework to construct quantum circuits of S-boxes according to a limited number of qubits. To construct such circuits, we analyzed the dimensions and bases before and after the Toffoli layer to find qubit values for which the equations match based on the meet-in-the-middle strategy. We employed the proposed tool to find the circuits of 4-bit S-boxes to verify its effectiveness in practice. Through the framework, we discovered all quantum circuits of odd permutations among all 4-bit optimal S-boxes classified by Leander and Poschmann. We also implemented quantum circuits of S-boxes for several well-known block ciphers, finding a more efficient quantum circuit of Saturnin's S-box. The proposed technique can be used to find circuits for S-boxes larger than 4 bits, which is left for future work. We believe that this technique contributes to the research field of finding the optimized quantum circuits of S-boxes.

6 Acknowledgements

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2023-RS-2022-00164800) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation

References

[1] Peter W Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science*, pages 124–134. IEEE, 1994.

[2] NIST. Post-quantum cryptography standardization, 2016. https://csrc.nist.gov/projects/post-quantum-cryptography.

[3] Lov K. Grover. A fast quantum mechanical algorithm for database search. In Gary L. Miller, editor, *STOC 1996*, pages 212–219. ACM, 1996. https://doi.org/10.1145/237814.237866.

[4] Daniel R. Simon. On the power of quantum computation. *SIAM J. Comput.*, 26(5):1474–1483, 1997.

[5] Markus Grassl, Brandon Langenberg, Martin Roetteler, and Rainer Steinwandt. Applying grover's algorithm to AES: quantum resource estimates. In Tsuyoshi Takagi, editor, *Post-Quantum Cryptography - 7th International Workshop, PQCrypto 2016, Fukuoka, Japan, February 24-26, 2016, Proceedings*, volume 9606 of *Lecture Notes in Computer Science*, pages 29–43, Cham, 2016. Springer.

[6] Mishal Almazrooie, Azman Samsudin, Rosni Abdullah, and Kussay N. Mutter. Quantum reversible circuit of AES-128. *Quantum Inf. Process.*, 17(5):112, 2018.

[7] Brandon Langenberg, Hai Pham, and Rainer Steinwandt. Reducing the cost of implementing AES as a quantum circuit. *IACR Cryptol. ePrint Arch.*, page 854, 2019.

[8] Jian Zou, Zihao Wei, Siwei Sun, Ximeng Liu, and Wenling Wu. Quantum circuit implementations of AES with fewer qubits. In Shiho Moriai and Huaxiong Wang, editors, *Advances in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7-11, 2020, Proceedings, Part II*, volume 12492 of *Lecture Notes in Computer Science*, pages 697–726. Springer, 2020.

[9] Zhenyu Huang and Siwei Sun. Synthesizing quantum circuits of AES with lower t-depth and less qubits. *IACR Cryptol. ePrint Arch.*, page 620, 2022.

[10] Samuel Jaques, Michael Naehrig, Martin Roetteler, and Fernando Virdia. Implementing grover oracles for quantum key search on AES and lowmc. In Anne Canteaut and Yuval Ishai, editors, *Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part II*, volume 12106 of *Lecture Notes in Computer Science*, pages 280–310. Springer, 2020.

[11] Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, Christof Paar, Axel Poschmann, Matthew J. B. Robshaw, Yannick Seurin, and C. Vikkelsoe. PRESENT: an ultra-lightweight block cipher. In Pascal Paillier and Ingrid Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop, Vienna, Austria, September 10-13, 2007, Proceedings*, volume 4727 of *Lecture Notes in Computer Science*, pages 450–466. Springer, 2007.

[12] Martin R. Albrecht, Benedikt Driessen, Elif Bilge Kavun, Gregor Leander, Christof Paar, and Tolga Yalçin. Block ciphers - focus on the linear layer (feat. PRIDE). In Juan A. Garay and Rosario Gennaro, editors, *Advances in Cryptology - CRYPTO 2014 - 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part I*, volume 8616 of *Lecture Notes in Computer Science*, pages 57–76. Springer, 2014.

[13] Wentao Zhang, Zhenzhen Bao, Dongdai Lin, Vincent Rijmen, Bohan Yang, and Ingrid Verbauwhede. RECTANGLE: a bit-slice lightweight block cipher suitable for multiple platforms. *Sci. China Inf. Sci.*, 58(12):1–15, 2015.

[14] Christof Beierle, Jérémy Jean, Stefan Kölbl, Gregor Leander, Amir Moradi, Thomas Peyrin, Yu Sasaki, Pascal Sasdrich, and Siang Meng Sim. The SKINNY family of block ciphers and its low-latency variant MANTIS. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II*, volume 9815 of *Lecture Notes in Computer Science*, pages 123–153. Springer, 2016.

[15] Subhadeep Banik, Sumit Kumar Pandey, Thomas Peyrin, Yu Sasaki, Siang Meng Sim, and Yosuke Todo. GIFT: A small present - towards reaching the limit of lightweight encryption. In Wieland Fischer and Naofumi Homma, editors, *Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings*, volume

10529 of *Lecture Notes in Computer Science*, pages 321–345. Springer, 2017.

[16] Vishnu Asutosh Dasu, Anubhab Baksi, Sumanta Sarkar, and Anupam Chattopadhyay. LIGHTER-R: optimized reversible circuit implementation for sboxes. In *32nd IEEE International System-on-Chip Conference, SOCC 2019, Singapore, September 3-6, 2019*, pages 260–265. IEEE, 2019.

[17] Vivek V. Shende, Aditya K. Prasad, Igor L. Markov, and John P. Hayes. Synthesis of reversible logic circuits. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.*, 22(6):710–722, 2003.

[18] Jiaqing Jiang, Xiaoming Sun, Shang-Hua Teng, Bujiao Wu, Kewen Wu, and Jialin Zhang. Optimal space-depth trade-off of CNOT circuits in quantum logic synthesis. In Shuchi Chawla, editor, *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pages 213–229. SIAM, 2020.

[19] Gregor Leander and Axel Poschmann. On the classification of 4 bit s-boxes. In Claude Carlet and Berk Sunar, editors, *Arithmetic of Finite Fields, First International Workshop, WAIFI 2007, Madrid, Spain, June 21-22, 2007, Proceedings*, volume 4547 of *Lecture Notes in Computer Science*, pages 159–176. Springer, 2007.

[20] Anne Canteaut, Sébastien Duval, Gaëtan Leurent, María Naya-Plasencia, Léo Perrin, Thomas Pornin, and André Schrottenloher. Saturnin: a suite of lightweight symmetric algorithms for post-quantum security. *IACR Trans. Symmetric Cryptol.*, 2020(S1):160–207, 2020.

[21] Zhenyu Huang and Siwei Sun. Synthesizing quantum circuits of AES with lower T-depth and less qubits. Cryptology ePrint Archive, Report 2022/620, 2022. https://eprint.iacr.org/2022/620.