Received: 23 January 2022 | Revised: 23 May 2022 | Accepted: 6 June 2022

IET Information Security

DOI: 10.1049/ise2.12074

The Institution of Engineering and Technology WILEY

ORIGINAL RESEARCH

T-depth reduction method for efficient SHA-256 quantum circuit construction

Jongheon Lee[1,2] | Sokjoon Lee[3] | You-Seok Lee[2] | Dooho Choi[4]

[1]Department of Information Security Engineering, University of Science and Technology (UST), Daejeon, Korea

[2]Cryptographic Engineering Research Section, Electronics and Telecommunications Research Institute (ETRI), Daejeon, Korea

[3]Department of Computer Engineering (Smart Security), Gachon University, Seongnam, Korea

[4]Department of AI Cyber Security, College of Science & Technology, Korea University Sejong, Sejong, Korea

Correspondence

Dooho Choi, Department of AI Cyber Security, College of Science & Technology, Korea University Sejong, Sejong 30019, Korea.
Email: doohochoi@korea.ac.kr

Funding information

Korea Government, Grant/Award Number: 2019-0-00033; Korea University

Abstract

To perform a quantum brute force attack on a cryptosystem based on Grover's algorithm, it is necessary to implement a quantum circuit of the cryptographic algorithm. Therefore, an efficient quantum circuit design of a given cryptographic algorithm is essential, especially in terms of quantum security analysis, and it is well known that T-depth should be reduced for time complexity efficiency. In this paper, the authors propose a novel technique to reduce T-depth (and T-count) when some quantum circuits located in between two Toffoli-gates are interchangeable with a controlled phase gate (CP gate), and the authors apply this technique to five types of quantum adders, reducing T-depth by more than 33%. The authors also present new SHA-256 quantum circuits which have a critical path with only three quantum adders while the critical paths of quantum circuits in the previous studies consist of seven or 10 quantum adders, and the authors also apply our technique to the proposed SHA-256 quantum circuits. Four versions of SHA-256 quantum circuit are presented. Among the previous results, T-depth of the circuit with the smallest Width (the number of qubits) 801 was approximately 109,104. On the other hand, T-depth of the proposed SHA-256 quantum circuit with the Width 797 is 16,055, which is remarkably reduced by about 85%. Another proposed quantum circuit only requires 768 qubits, which is the smallest Width compared to the previous results to the best of our knowledge. Furthermore, one other version is the most time-efficient circuit with an overall Toffoli-depth (and T-depth) that is less than 5000.

1 | INTRODUCTION

Hash algorithms were created to provide secure data transmission and data integrity in information and communication protocols. Hash algorithms can be used for digital signatures, keyed message authentication codes, random number generation, key derivation functions etc. [1]. In particular, the SHA-2 hash family, published in 2002, is designated as a hash function standard [2].

If a collision attack is performed on a hash algorithm, it is known that the attack succeeds if the operation is performed several times corresponding to half the length of the bit string constituting the output value due to the birthday paradox. SHA-256 provides 128-bit security strength against classical collision attack [1, 2].

There have been many theoretical studies on hash functions such as MD5, SHA-1, and SHA-512, and in a quantum environment, it is known that the security strength of existing hash functions is halved when using Grover's algorithm [3]. Previous studies dealt with several quantum attack algorithms and methods of implementing quantum circuits of the cryptographic algorithms that can perform pre-image attack, second pre-image attack, and collision attack [4–6]. However, the circuits presented in the previous studies have few details on the implementation of the hash algorithm, and they are forms made intuitively rather than created by efficient quantum circuits, such as reducing Depth or Width (the number of qubits) of the circuits. That is, the circuit is inefficient to use the existing circuit as it is to verify whether the security strength is reduced by half.

Although there are various standardised hash families such as MD5, SHA-2, and SHA-3, we focus on quantum design for SHA-2, since it is currently the most widely commercialised, such as SSL digital authentication and IEEE 1609.2-based wireless V2X communication. There are 6 algorithms totally in the SHA-2 hash family, and the structures of these algorithms are the same except for some constants and the number of rounds. Hence, we concentrate on SHA-256 and designed quantum circuits in this paper. Before presenting the SHA-2 quantum circuit, we present the quantum adder circuits to be used in the SHA-2 quantum circuit. Since many additions are performed in SHA-2, it is essential to use an efficient quantum adder circuit. In general, in quantum circuits, as T-depth (Depth formed by T and $T^{\dagger}$ gates in the critical path) and Width increase, the computational complexity of quantum computing exponentially increases, which degrades circuit performance. We will present T-depth (and T-count) reduction methods that can be used in quantum adders and SHA-256 quantum circuits. In this paper, our contribution is threefold:

- We introduce a novel technique to be able to reduce T-depth. We show that if some quantum circuits located in between two Toffoli-gates are interchangeable with a controlled phase gate, then T-depth six is reduced to four or five.
- We apply the above trick to five quantum adders—CDKM adder [7], VBE adder [8, 9], TK (Y. Takahashi and N. Kunihiro) adder [10], HRS (T. Häner, M. Roetteler and K. M. Svore) adder [11] and QCLA [12]—and so our improved adders have the effect of reducing the T-depth by more than 33% compared to the previous adders. Additionally, a modified version of the TK adder is presented by applying the Toffoli-count reduction rules. This adder has a new ripple-carry form and Toffoli-depth is the same as the CDKM adder's Toffoli-depth.
- We propose a new SHA-256 quantum circuit design that has a critical path with only three quantum adders, while the critical paths of quantum circuits in the previous studies consist of seven or 10 quantum adders [5, 6]. We use our improved adders in this new SHA-256 quantum circuit, and also this T-depth reduction method is applied in the function blocks such as Maj, and Ch in the SHA-256 quantum circuit. One of our resulting circuits requires only 797 qubits and has 12,023 Toffoli-depth which is a huge reduction of 67% compared to the previous work [6] with the smallest Width 801. Another proposed SHA-256 quantum circuit only requires 768 qubits which is the smallest Width of all the previous results to the best of our knowledge. Furthermore, another proposed circuit is the most time-efficient SHA-256 quantum circuit with Toffoli-depth (and T-depth) less than 5000. Additionally, the design of the Ch function block, one of the SHA-256 function blocks, is newly proposed.

This paper is an improved extension version of Ref. [13] with the following contents added. Section 2 introduces the pruning procedure [14]. The reason for introducing this procedure is that it will be used as a pre-processing step of the presented T-depth reduction method in Section 3. We use a modified version of the existing pruning procedure. This procedure clarifies the area we cover. In Section 3, we added a second T-depth reduction process as well as the pruning procedure. That is, a situation in which T-depth can be further reduced is presented. In this process, it is very difficult to further reduce T-count. The exchangeability determination algorithm is revised and supplemented, and specific examples of success and failure are presented. In Section 4, a modified TK adder is presented. Toffoli-depth of the suggested TK adder is consistent with that of the CDKM adder. When designing the SHA-256 quantum circuit, we replaced CDKM adders with these adders. QCLA is further covered. That is, the proposed T-depth reduction technique is applied to QCLA. For the VBE adder and HRS adder, the reason why resources vary according to the LSB value is explained. In Section 5, Table 1, which presents resources for function blocks that do not use Toffoli-gates at all, is added. These can be implemented by arranging CNOT gates in the reverse order of PLU decomposition. Table 2 presents the improved T-depth

TABLE 1 Σ_0, Σ_1, σ_0, σ_1 function block resources. These function blocks can be implemented using only controlled-NOT gate (CNOT) gates and do not use work qubits at all. These circuits are implemented by arranging CNOT gates in the reverse order of PLU decomposition [4]

	Count	Depth	Swapping
Σ_0	166	55	17
Σ_1	166	44	22
σ_0	193	50	20
σ_1	142	40	23

TABLE 2 Function block resources in the SHA-256 quantum circuit. These T-depth values are the results after our T-depth reduction technique is applied. All quantum adders are performed in modular 2^{32}. HRS and VBE adders are only used when adding constant K_t. In the HRS adder, 8 work qubits are borrowed dirty, that is, they are in arbitrary states. Toffoli-depths and T-depths of VBE and HRS adders depend on the value of LSB of constant K_t. That is, if the value of LSB is 0, the lower values are taken. TK-v1, TK-v2, and TK-v3 adders in the table are modified versions we made. They have the same Toffoli-depth as the CDKM adder. The CDKM adder is not used when designing the SHA-256 quantum circuit, but it is included for comparison with the resources of other adders

	Width	#Ancilla	T-depth	Toffoli-depth
Maj	3	0	2	1
Ch	3	0	2	1
Σ_0, Σ_1, σ_0, σ_1	32	0	0	0
VBE adder	61	29	60 or 62	57 or 59
TK-v1 adder	64	0	122	61
TK-v2 adder	65	1	62	61
TK-v3 adder	67	3	61	61
HRS adder	40	8	424 or 432	384 or 392
QCLA	117	53	24	22
CDKM adder	65	1	64	61

Abbreviation: QCLA, quantum carry-look ahead adder.

(Toffoli-depth) of each adder circuit. In Figure 27, the arrangement of gates in the message schedule algorithm is changed. A total of four circuits were presented by adding two new circuits. They were named SHA-Z1, SHA-Z2, SHA-Z3, and SHA-Z4, respectively.

The rest of the paper is constructed as follows: Section 2 introduces some background for the Toffoli-gate and related work for the SHA-2 hash family and its quantum circuits. In Section 3, we propose a T-depth and T-count reduction method by gathering ideas from several papers. In Section 4, we apply this reduction method to quantum adder circuits. Section 5 presents the SHA-256 quantum circuit made using these improved quantum adder circuits. We present a new design structure and apply the reduction method to the function blocks used in SHA-256. Finally, this section compares the number of quantum resources with other existing quantum circuits. Section 6 mentions the conclusions and topics for further research in the future.

2 | BACKGROUND AND RELATED WORK

2.1 | Toffoli-gate

In classical circuits, NAND gates and Fanout gates form a universal set of gates for classical computation. In a quantum circuit, the Clifford + T set forms the standard universal fault-tolerant gate set [15]. {H, CNOT, T} is a minimal generating set of the Clifford + T set. Hadamard gate (H gate), NOT gate (X gate), T gate, P gate (= T^2 gate), Z gate (= T^4 gate), and Controlled-NOT gate (CNOT gate) belonging to Clifford + T set are widely used and performed as follows. $\oplus$ stands for modulo-2 addition.

$$\begin{aligned} &H:|x_1\rangle \to \frac{|0\rangle + (-1)^{x_1}|1\rangle}{\sqrt{2}} && X:|x_1\rangle \to |x_1 \oplus 1\rangle \\ &CNOT:|x_1x_2\rangle \to |x_1(x_1 \oplus x_2)\rangle && T:|x_1\rangle \to e^{\frac{\pi i}{4}x_1}|x_1\rangle \\ &P:|x_1\rangle \to e^{\frac{\pi i}{2}x_1}|x_1\rangle && Z:|x_1\rangle \to (-1)^{x_1}|x_1\rangle \end{aligned} \tag{1}$$

In the next section, we will deal with the circuit composed of H gate, X gate, Z-rotation gate (R_z gate (T, P, Z gate)), and CNOT gate. As the counterpart of AND gate in the classical circuit, there is a Toffoli-gate in the quantum circuit. A Toffoli-gate is a doubly controlled-NOT gate (C^2NOT gate), and it causes a change in the remaining one value according to two input values among three input values. This gate can be decomposed into two H gates and one doubly controlled Z gate (C^2Z gate, Figure 1). A Toffoli-gate can be implemented in various ways by properly arranging two H gates, T/$T^\dagger$ gates, and CNOT gates. The reason Toffoli-gates need to be implemented in various ways is that with proper conversion, Toffoli-gates can share some T-depth with Toffoli-gates next to them. We will call this T-depth sharing. In Section 4, when making quantum circuits, we will show that T-depth can be reduced by using T-depth sharing and various implementation methods such as the second implementation in Figure 1.

$$\begin{aligned} C^2NOT:|x_1x_2x_3\rangle &\to |x_1x_2(x_3 \oplus (x_1x_2))\rangle \\ C^2Z:|x_1x_2x_3\rangle &\to (-1)^{x_1x_2x_3}|x_1x_2x_3\rangle \end{aligned} \tag{2}$$

The phase of the output value from the C^2Z gate can be expressed exactly as $e^{\frac{\pi i}{4}4x_1x_2x_3}$. For $x_1, x_2, x_3 \in \{0,1\}$, the following expression holds [17].

$$\begin{aligned} 4x_1x_2x_3 = {} & x_1 + x_2 + x_3 - x_1 \oplus x_2 - x_2 \oplus x_3 \\ & - x_1 \oplus x_3 + x_1 \oplus x_2 \oplus x_3 \end{aligned} \tag{3}$$

It can be seen that $4x_1x_2x_3$ consists of seven operands. That is, C^2Z gate consists of 4 T gates and 3 $T^\dagger$ gates. T gates make phases $e^{\frac{\pi i}{4}x_1}$, $e^{\frac{\pi i}{4}x_2}$, $e^{\frac{\pi i}{4}x_3}$, and $e^{\frac{\pi i}{4}x_1 \oplus x_2 \oplus x_3}$. $T^\dagger$ gates create phases $e^{-\frac{\pi i}{4}x_1 \oplus x_2}$, $e^{-\frac{\pi i}{4}x_2 \oplus x_3}$, and $e^{-\frac{\pi i}{4}x_1 \oplus x_3}$.

Meanwhile, the controlled|-$P^\dagger$ gate (C$P^\dagger$ gate) makes phase $e^{-\frac{\pi i}{4}2x_1x_2}$ $(= (-i)^{x_1x_2})$. Since $2x_1x_2 = x_1 + x_2 - x_1 \oplus x_2$, it can be seen that C$P^\dagger$ gate consists of one T gate and two $T^\dagger$ gates. If C$P^\dagger$ gates exist on the two control lines of the Toffoli-gate, these two gates become a C^2(−iX) gate composed of two T gates and two $T^\dagger$ gates [17] (Figure 2).

$$\begin{aligned} & x_1 + x_2 + x_3 - x_1 \oplus x_2 - x_2 \oplus x_3 - x_1 \oplus x_3 + x_1 \oplus x_2 \oplus x_3 \\ & - (x_1 + x_2 - x_1 \oplus x_2) \\ & = x_3 - x_2 \oplus x_3 - x_1 \oplus x_3 + x_1 \oplus x_2 \oplus x_3 \end{aligned} \tag{4}$$

2.2 | Pruning procedure: Identifying a subcircuit consisting only of {X, CNOT, R_z} [14]

In the fault-tolerant model, it is known that implementing the T gate among the gates of the Clifford + T set is much more difficult than all other gates and that the T-depth determines

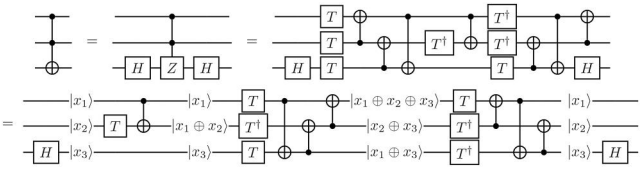

FIGURE 1 A Toffoli-gate with T-depth 3 and T-count 7 [16]. Toffoli-gate can be designed in various ways as shown in the figure, and T-depth is at least 3 unless there is a work qubit (ancilla qubit)

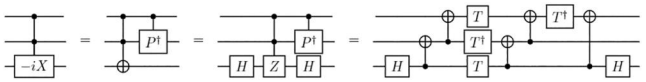

FIGURE 2 A doubly controlled (−iX) gate [17]. When there is no work qubit, at least 5 controlled-NOT gate (CNOT) gates are needed to make a C^2(−iX) gate. The case of using 6 CNOT gates can be seen in Figure 11

the circuit runtime [18–20]. In a quantum circuit, if there are two T gates consecutively, it can be converted to a P gate, and the design cost of the quantum circuit can be lowered, and the performance can be increased. In Ref. [14], the researchers studied how to achieve merging when there are multiple R_z gates in a quantum circuit in general. That is, they studied how to merge one R_z gate with another R_z gate without changing the overall operation. They formed a subcircuit composed of NOT gates, R_z gates, and CNOT gates in the entire circuit and attempted to merge R_z gate within it. The pruning procedure was suggested as a method of forming a subcircuit. That is, they confirmed whether R_z gates can be merged after clearly defining the subcircuit including R_z gates through this procedure. Only gates that can be involved in merging are selected to form a subcircuit.

We present the pruning procedure that can identify the subcircuits presented in the previous study. This procedure is governed by the following rules.

- The pruning procedure starts by designating one CNOT gate as the start point. We decide whether to include X, R_z, CNOT, and H gates that meet while traversing back and forth of this CNOT gate in the subcircuit.
- The border constituting the subcircuit is called the termination border, and the termination border gets wider through traversing. The basic termination border consists of lines with the CNOT gate, which is the start point and is the area just before reaching the termination points.
- If we encounter an X or R_z gate while traversing, we can pass unconditionally and continue traversing. If the H gate is met, it is designated as a termination point, and traversing in that direction is stopped. Alternatively, if the end of the circuit is reached, the end is designated as a termination point and traversing is stopped.
- When we encounter the CNOT gate while crossing, we must carefully consider whether we can pass or not. If this CNOT gate shares both lines with the CNOT gate, which is the start point, we can pass through this gate and continue to cross. To be more precise, if the CNOT gate is included in the existing termination border, we can pass the control part or the target part of the CNOT gate. If a target part of a new CNOT gate is encountered while traversing, the target part is designated as a termination point and traversing is stopped in principle. However, when a target part is encountered, if the corresponding control part is contained within the termination border, we can pass through the target part and continue traversing. When a control part is encountered, if only the control part is included in the termination border, that is, if the target part is located outside the termination border, this CNOT gate can be passed because it creates new connectivity with a new line. We designate the target part of this CNOT gate as the anchor point. The termination border (or subcircuit) can be widened through this anchor point. If the control part of the CNOT gate that cannot create new connectivity is encountered, the CNOT gate is excluded. That is, this CNOT gate is skipped, and traversing continues.
- After traversing at the two lines with the CNOT gate as the start point, traverse at the anchor points. All gates passed while traversing constitute a subcircuit. Gates that have been skipped or not encountered while crossing are excluded.

Let us look at the example in Figure 3. There are a total of 6 R_z gates in the whole circuit, and it is composed of a total of 3 lines. We designate the top left CNOT gate in the circuit as the start point. This CNOT gate exists on the first and second lines. If we cross from the CNOT gate, which is the start point, to the left in the first line, we see the R_z gate first, and then the H gate. The R_z gate can be passed through and the H gate is assigned as a termination point and the traversal is terminated. Conversely, if we cross to the right, we can meet 3 R_z gates and 2 CNOT gates. The first CNOT gate can be passed because it shares both wires with the start point. R_z gates can be passed unconditionally, and in the case of the second CNOT gate, we encounter a control part that creates new connectivity. Therefore, this CNOT gate can be passed. The target part of this CNOT gate is designated as an anchor point. After passing through the $R_z(\theta_4)$ gate, we reach the end of the circuit, so the end becomes the termination point. Let us look at the second line this time. We start from the CNOT gate, which is the start point. If we cross to the left, we meet the control part of one CNOT gate. However, since this CNOT gate does not form new connectivity, it is skipped and continues traversing. We reach the end of the circuit. If we turn to the right, we meet 3 CNOT gates and 1 H gate. The first CNOT gate is also an exception because it cannot create new connectivity. The second CNOT gate is passed because it is the gate that has already been determined to be in the termination border in the first line. Then, the target part of the third CNOT gate is met. Originally, this target part should be designated as a termination point and traversal should be completed. However, due to the target part of the other CNOT gate designated as the anchor point made in the first line, both the control part and the target part of this third CNOT gate are placed within the

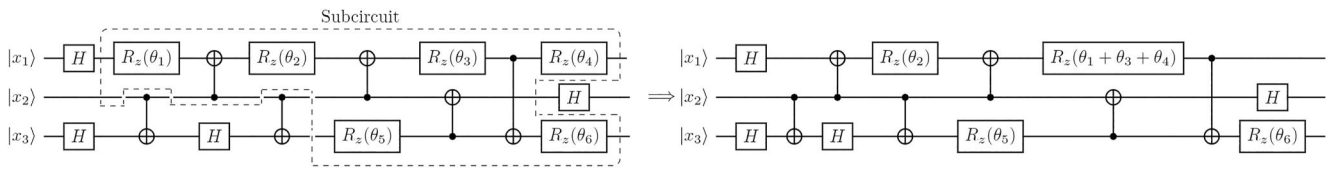

FIGURE 3 Example: A subcircuit made by the pruning procedure [14]. It is used in rotation (R_z gates) merging. They identified a subcircuit consisting of X, controlled-NOT gate (CNOT), and R_z gates through the pruning procedure, and checked whether R_z gates can be merged within the subcircuit in the previous work

termination border. Therefore, we can pass through this CNOT gate and continue traversing. Finally, we meet the H gate at the far right and stop the crossing. Now let us look at the third line. We start traversing at the target part we previously marked as the anchor point. Crossing to the left, we meet and pass through the CNOT gate just mentioned. The $R_z(\theta_5)$ gate is encountered, and then the CNOT gate, which was previously excluded, is encountered. We skip this CNOT gate and meet the H gate. We end the crossing. If we cross to the right, there is the $R_z(\theta_6)$ gate. The end of the circuit becomes the termination point. By summing all the gates passed during traversing, one subcircuit is composed (Figure 3). As a result, the subcircuit consists of 4 CNOT gates and 6 R_z gates. Three R_z gates are merged into 1 R_z gate by judging the merging possibility in this subcircuit and using the rules in Ref. [14]. More details can be found in Ref. [14]. We will modify this pruning procedure as a pre-processing step in the proposed T-depth reduction method.

2.3 | Quantum adder circuit

There are five types of quantum adder circuits covered in this paper: CDKM adder, VBE adder, TK adder, HRS adder, and Basic QCLA (Quantum carry-look ahead adder) [7, 8, 10–12]. The first four adders follow the ripple-carry form (QRCA, quantum ripple-carry adder), especially HRS adder is in-place constant-adder. Quantum resources used by each adder can be found in Table 2. In SHA-256, all adders are used in modular 2^{32}.

In the proposed SHA-256 quantum circuit, the VBE adder and HRS adder are used only when adding the constant K_t [9, 11]. It is decided whether some gates in the VBE adder and HRS adder are included through bit-values of constant. Also, it is possible to obtain the sum of 32-bit operands in modular 2^{32} with only 61 and 40 qubits in total, respectively.

The CDKM adder [7] is a typical QRCA circuit that requires at least one work qubit (ancilla qubit) regardless of whether modular addition is performed. However, the TK adder is a QRCA adder that does not use work qubits at all. Since the TK adder uses fewer qubits than the CDKM adder, Toffoli-depth is larger than that of the CDKM adder. However, by using the Toffoli-count reduction rules which are not covered in detail in this paper, we will change the circuit to have the same Toffoli-depth as the CDKM adder. Because Toffoli-gates in the modified version of the TK adder are executed sequentially, Toffoli-count is consistent with Toffoli-depth. Therefore, if we use a total of 4 work qubits, T-depth can be consistent with these values by using the Matroid Partitioning concept [21]. In fact, in the modified version of the TK adder, T-depth can be lowered to the Toffoli-depth with only 3 working qubits. Toffoli-depth is a measure used in Ref. [6] and is a concept used instead of T-depth. In the previous work, there are two main reasons for analysing circuit resources with Toffoli-depth. One is that T gates are mostly used within Toffoli-gates, so even if the circuit performance is expressed with relatively inaccurate Toffoli-depth, that is logically justified. The other is that they do not decompose quantum circuits down to Clifford + T gate sets to form basic-level circuits. That is, they omitted the T-depth and T-count reduction process. In this work, we will refer to both Toffoli-depth and more accurate T-depth.

A quantum carry-look ahead adder (QCLA) is a quantum version of the classic carry look-ahead adder. For bit-length n of operands, Toffoli-depths of the preceding quantum adders are $\mathcal{O}(n)$, whereas Toffoli-depth of QCLA is $\mathcal{O}(\log n)$. But it uses a lot of work qubits. Note that one QCLA uses 53 work qubits for 32-bit module addition [12].

In the meantime, many studies have been conducted for efficient QCLA design [22–25]. As mentioned earlier, we consider using a circuit based on the Clifford + T gate set. Also, we want to deal with a quantum circuit that is not based on MBQC (measurement-based quantum computation). Because MBQC has an intermediate measurement process that may affect time complexity as much as T-depth. Among QCLAs that satisfy these two conditions, we chose Basic QCLA with the smallest Toffoli-depth, and we will simply call it QCLA from now on.

In the next section, we will provide a logic process that can reduce the T-count and T-depth of these five adder circuits. This process does not change Toffoli-count or Toffoli-depth.

2.4 | Secure hash Algorithm-2

2.4.1 | Pre-processing step

The SHA-2 hash algorithm consists of two main steps: a pre-processing step and a hash computation step [2]. In the first step, message padding and parsing are performed, and bits are added so that the length of the padded message is a multiple of 512 bits. The original message must be less than 2^{64} in length. That is, the length of the message must be able to be expressed in 64 bits. In this paper, it is assumed that the number N of 512-bit message blocks after the pre-processing step is 1. There is a hash value update process after the main round function algorithm is executed 64 times in SHA-256. In this process, if N is two or more, an operation to copy the value is required. However, this cannot be carried out by the No-cloning theorem in quantum environment [26]. Therefore, it is necessary to assume that N is 1, and that is why the hash value update operation was omitted in previous studies [5, 6]. The maximum length of the original message that can be handled by the SHA-256 quantum circuit is 447 because the minimum length of the padding is 65 in one message block.

2.4.2 | Hash computation step

In the hash computation step, a hash value (message digest) is created. Depending on the number of message blocks of the padded message, the entire algorithm iteration occurs. The main hash computation algorithm has been repeated a total of 64 times each time this iteration is performed. In the main hash computation algorithm, all additions are performed in modular

2^{32} and are largely divided into the message schedule algorithm and round function algorithm.

In the message schedule algorithm, 48 W_t (t = 16, …, 63) are created by using W_t (t = 0, …, 15), which is the existing 16 32-bit words composed of padded message values. Also, two logical functions $\sigma_0(x)$ and $\sigma_1(x)$ are used.

The initial hash value $H^{(0)}$ is a 256-bit constant value and is assigned to eight 32-bit internal variables a, b, c, d, e, f, g, and h used in the round function algorithm. Four logical functions Maj, Ch, $\Sigma_0(x)$, and $\Sigma_1(x)$ are used in the round function Algorithm. A total of 64 32-bit words K_0, K_1, …, and K_{63} are added sequentially for each round. The quantum circuit implementation for each internal function is introduced in section 5.

$$\begin{aligned} W_t &= M_t^{(1)} \quad 0 \le t \le 15 \\ &= \sigma_1(W_{t-2}) + W_{t-7} + \sigma_0(W_{t-15}) + W_{t-16} \quad 16 \le t \le 63 \\ where \quad \sigma_0(x) &= ROTR^7(x) \oplus ROTR^{18}(x) \oplus SHR^3(x), \\ \sigma_1(x) &= ROTR^{17}(x) \oplus ROTR^{19}(x) \oplus SHR^{10}(x) \end{aligned} \tag{5}$$

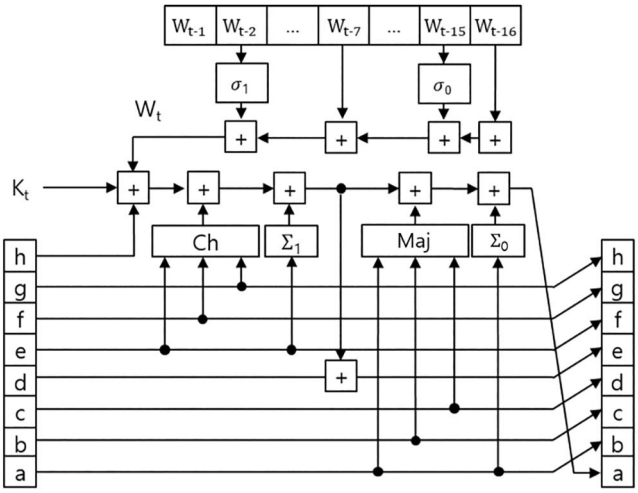

FIGURE 4 Round function and message schedule algorithm in SHA-256 [6]

$$\begin{aligned} &h = g, g = f, f = e, e = d + T_1, \\ &d = c, c = b, b = a, a = T_1 + T_2 \\ &where \quad T_1 = h + \Sigma_1(e) + Ch(e,f,g) + K_t + W_t, \\ &\qquad T_2 = \Sigma_0(a) + Maj(a,b,c), \\ &Maj(x,y,z) = (x \wedge y) \oplus (x \wedge z) \oplus (y \wedge z), \\ &Ch(x,y,z) = (x \wedge y) \oplus (\neg x \wedge z), \\ &\Sigma_0(x) = ROTR^2(x) \oplus ROTR^{13}(x) \oplus ROTR^{22}(x), \\ &\Sigma_1(x) = ROTR^6(x) \oplus ROTR^{11}(x) \oplus ROTR^{25}(x) \end{aligned} \tag{6}$$

After the 64th round iteration is completed, the intermediate hash value $H^{(i-1)}$ is added to the values of a, …, h and updated to the value $H^{(i)}$. Since we assumed N = 1 earlier, i = 1. Finally, we get the 256-bit hash value $H^{(1)} = H_0^{(1)} \| H_1^{(1)} \| \ldots \| H_7^{(1)}$. In the proposed quantum circuit, as in previous studies, we did not include the operation process of adding internal variables to $H^{(0)}$. The reason is that it is a meaningless part to implement the quantum cryptosystem circuit required in Grover's algorithm. If you look at Figure 4, you can see the procedure of the SHA-256 algorithm.

$$\begin{aligned} &H_0^{(1)} = H_0^{(0)} + a, H_1^{(1)} = H_1^{(0)} + b, H_2^{(1)} = H_2^{(0)} + c, \\ &H_3^{(1)} = H_3^{(0)} + d, H_4^{(1)} = H_4^{(0)} + e, H_5^{(1)} = H_5^{(0)} + f, \\ &H_6^{(1)} = H_6^{(0)} + g, H_7^{(1)} = H_7^{(0)} + h \end{aligned} \tag{7}$$

In the classic SHA-256 circuit, the critical path is a section in which 7 operands are added using six adders to produce the output value $T_1 + T_2$ [27]. Through repeated execution, $T_1 + T_2$ is updated 64 times and is continuously allocated to the internal variable A. Studies to reduce the time taken by this critical path in classical circuits have been conducted in several papers [27, 28].

2.4.3 | SHA-256 quantum circuit

Ref. [4–6] presented quantum cryptosystem circuit implementations to investigate the security strength of the various cryptosystems. Among them, the resources used for the SHA-256 quantum circuit can be seen in Table 3. We implemented the circuit based on the Clifford + T set, as in previous studies.

TABLE 3 SHA-256 quantum circuit resources comparison. Our four proposed circuits are named SHA-Z1, SHA-Z2, SHA-Z3 and SHA-Z4, respectively. When constructing our circuits, we used adder circuits with reduced T-depth to which the above T-depth reduction technique was applied

	Width	T-depth	Toffoli-depth	Used quantum adder
SHA-256 [5]	2402	70,400	-	CDKM [7]
SHA-C1 [6]	801	-	36,368	CDKM [7]
SHA-C2 & SHA-C3 [6]	853	-	13,280	QCLA [12]
SHA-C4 [6]	834	-	27,584	CDKM [7]
SHA-C5 & SHA-C6 [6]	938	-	10,112	QCLA [12]
SHA-Z1	768	43,510	32,895	HRS, TK-v1
SHA-Z2	797	16,055	12,023	VBE, TK-v1, TK-v3
SHA-Z3	927	7304	6914	VBE, TK-v3, QCLA
SHA-Z4	962	4936	4418	QCLA, TK-v2

Abbreviation: QCLA, quantum carry-look ahead adder.

In Ref. [5], the CDKM adder [7] was used throughout the SHA-256 quantum circuit, and the quantum circuit implementation of each internal function was briefly presented. The round function algorithm and message schedule algorithm were processed in parallel, and T-depth and T-count were optimised by performing T-par for the entire circuit [21].

In Ref. [6], SHA-256 quantum circuit implementation was presented in four versions, and the CDKM adder and QCLA were used as adders [7, 12]. As mentioned earlier, in this previous study, only Width and Toffoli-depth for each version were mentioned without performing T-depth and T-count reduction (optimisation) work.

In the quantum circuit we present, the round function and message schedule algorithm are processed in parallel. Unlike previous work, it is not implemented with only one adder, but as a hybrid version in which several types of adders are placed in appropriate positions. For accurate comparison with previous papers, both T-depth and Toffoli-depth are written in Table 3.

3 | T-DEPTH AND T-COUNT REDUCTION METHOD

Many studies have been conducted on T-depth and T-count reduction to reduce and optimise quantum circuit construction costs [14, 17, 21, 29]. We will use the Matroid Partitioning concept in Ref. [21] and modify the optimisation subroutines in Ref. [14]. We will deal with a quantum circuit with a subcircuit (an intermediate region) between two Toffoli-gates. Our method is applicable even if the Toffoli-gate has off-control parts.

If there are two Toffoli-gates in the quantum circuit, their relative positions exist in 10 cases [30] (Figure 5). Our method can be applied to all 10 cases. Four of these cases will be dealt with in detail in this paper. Three cases we will cover are when both the control lines and the target line are shared, only the control lines are shared, and only the target line is shared. The fourth case is the fourth sub-figure in Figure 5. One control line is shared, but the rest of the parts share the lines by crossing each other. That is, control parts and target parts share lines. The remaining six cases will be briefly mentioned. Because for the remaining six cases, our method can be applied similarly to the four cases we will cover in detail. Our method has a total of two processes and is called the first T-depth reduction process and the second T-depth reduction process. After performing the first T-depth reduction process, if a specific condition is satisfied, the second process can be performed. Both of these processes are performed with the same logic and algorithm.

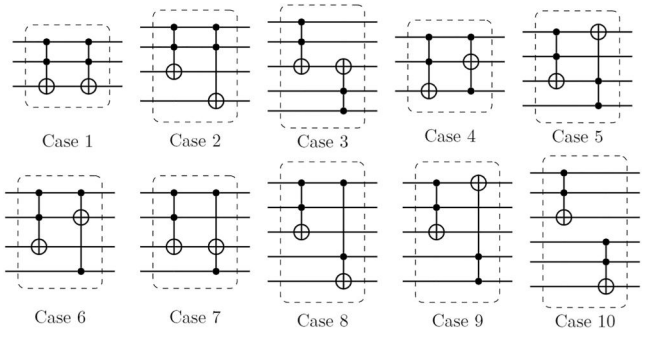

FIGURE 5 Ten cases of relative positions for two Toffoli-gates [30]. Of these, the first four cases are dealt with in detail. For the remaining six cases, our method can be applied similarly to the four cases

3.1 | First T-depth reduction process

3.1.1 | Case 1: Both control lines and a target line are shared

Assume that two Toffoli-gates share control lines and a target line and there is a subcircuit named A between these two gates as shown in Figure 6. Now we create $CP^{\dagger}$ gate and controlled-P gate (CP gate) on the right side of Toffoli-gate on the left side of the circuit. Since these two gates have an inverse operation relationship with each other, it is self-evident to constitute an identity circuit. Now consider the CP gate and the existing subcircuit A. If these two partial circuits are commutative, the CP gate can be moved to the left of the Toffoli-gate on the right side of the circuit. That is, if this exchange is possible as shown in Figure 6, the total T-depth can be reduced from six to 4, and the total T-count can be reduced from 14 to 8. Two Toffoli-gates become $C^2(-iX)$ gates and $C^2(iX)$ gates, respectively.

3.1.2 | Case 2: Control lines are shared, but a target line is not.

Figure 7 shows the case where the control lines are shared, but the target line is not. T-depth and T-count can be reduced

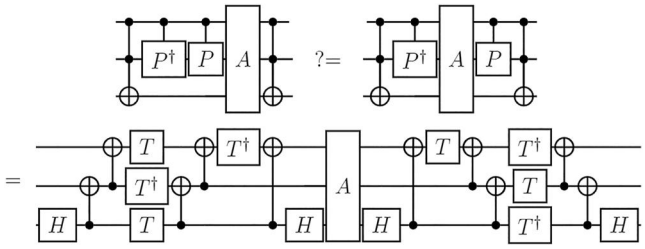

FIGURE 6 Case 1: both control lines and a target line are shared. If controlled-P gate (CP gate) and subcircuit A are interchangeable, T-depth can be reduced from 6 to 4

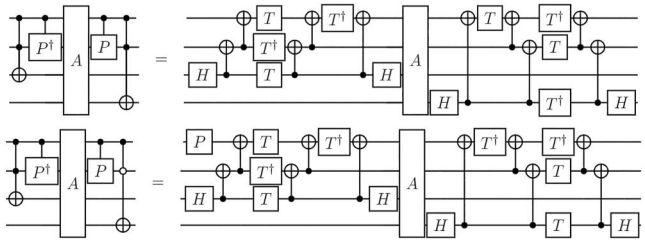

FIGURE 7 Case 2: control lines are shared, but a target line is not. Even if Toffoli-gate has an off-control part, T-depth can be reduced from 6 to 4

through the same logic. If there is a Toffoli-gate with an off-control part, as in the figure below of Figure 6, a P gate or $P^\dagger$ gate could be created. Let the basis variables corresponding to each line be x_1, x_2, x_3, and x_4 [21]. As we saw in the previous section, these basis variables are the components of phase in the $R_z(\theta)$ gate. It can be seen that T-count can be reduced to eight through Equations (8) and (9).

$$\begin{aligned}(-1)^{x_1x_2x_3}(-i)^{x_1x_2}(i)^{x_1x_2}(-1)^{x_1x_2x_4} &= \left(e^{i\pi/4}\right)^{x_3-x_2\oplus x_3-x_1\oplus x_3+x_1\oplus x_2\oplus x_3-x_4+x_2\oplus x_4+x_1\oplus x_4-x_1\oplus x_2\oplus x_4}\\ &\times (x_1+x_2+x_3-x_1\oplus x_2-x_2\oplus x_3-x_1\oplus x_3+x_1\oplus x_2\oplus x_3)+(-x_1-x_2+x_1\oplus x_2)\\ &+(x_1+x_2-x_1\oplus x_2)+(-x_1-x_2-x_4+x_1\oplus x_2+x_2\oplus x_4+x_1\oplus x_4-x_1\oplus x_2\oplus x_4)\\ &=(x_3-x_2\oplus x_3-x_1\oplus x_3+x_1\oplus x_2\oplus x_3)+(-x_4+x_2\oplus x_4+x_1\oplus x_4-x_1\oplus x_2\oplus x_4)\end{aligned} \tag{8}$$

$$\begin{aligned}(-1)^{x_1x_2x_3}(-i)^{x_1x_2}(i)^{x_1x_2}(-1)^{x_1\overline{x_2}x_4} &= \left(e^{i\pi/4}\right)^{x_3-x_2\oplus x_3-x_1\oplus x_3+x_1\oplus x_2\oplus x_3+2x_1+x_4+x_2\oplus x_4-x_1\oplus x_4-x_1\oplus x_2\oplus x_4}\\ &\times (x_1+x_2+x_3-x_1\oplus x_2-x_2\oplus x_3-x_1\oplus x_3+x_1\oplus x_2\oplus x_3)+(-x_1-x_2+x_1\oplus x_2)\\ &+(x_1+x_2-x_1\oplus x_2)+(x_1+\overline{x_2}+x_4-x_1\oplus\overline{x_2}-\overline{x_2}\oplus x_4-x_1\oplus x_4+x_1\oplus\overline{x_2}\oplus x_4)\\ &=(x_3-x_2\oplus x_3-x_1\oplus x_3+x_1\oplus x_2\oplus x_3)+(x_1+x_2-x_1\oplus x_2)+(x_1+(1-x_2)+x_4\\ &-(1-x_1\oplus x_2)-(1-x_2\oplus x_4)-x_1\oplus x_4+(1-x_1\oplus x_2\oplus x_4))=(x_3-x_2\oplus x_3\\ &-x_1\oplus x_3+x_1\oplus x_2\oplus x_3)+(2x_1+x_4+x_2\oplus x_4-x_1\oplus x_4-x_1\oplus x_2\oplus x_4)\\ &\times\left(For\ a,b\in\{0,1\},\ \overline{a}=a\oplus 1=1-a,\ and\ \overline{a}\oplus b=\overline{a\oplus b}.\right)\end{aligned} \tag{9}$$

3.1.3 | Case 3: A target line is shared, but control lines are not.

In this case, T-count does not change, but T and $T^\dagger$ gates composing different Toffoli-gates share T-depth and consequently reduce it. We drew Figure 8 using three Toffoli-gates. Unlike the previous two cases, this third case is a method applicable even when three or more Toffoli-gates exist. Because the control lines are not shared among Toffoli-gates, T-depth cannot be reduced with the same logic as in the previous two cases. Looking at Figure 8, the rightmost Toffoli-gate maintains T-depth 3 and T-count 7 without any conversion. If CP gates and subcircuits A_1 and A_2 are interchangeable, T-depth of the rightmost Toffoli-gate could be shared. That is, T and $T^\dagger$ gates of these three gates are placed on the same timeline to make T-depth 3.

It is possible to reduce the T-depth from 3n to $2n+1$ for n (≥1) Toffoli-gates in this case. If a CP gate is made on control lines of the rightmost Toffoli-gate and this gate is interchangeable with subcircuit A_2, T-depth can be reduced by one more. In all cases, if one work qubit is added, T-depth can be further reduced [17]. In particular, in the third case, the T-depth $2n+1$ can be reduced to $n+1$.

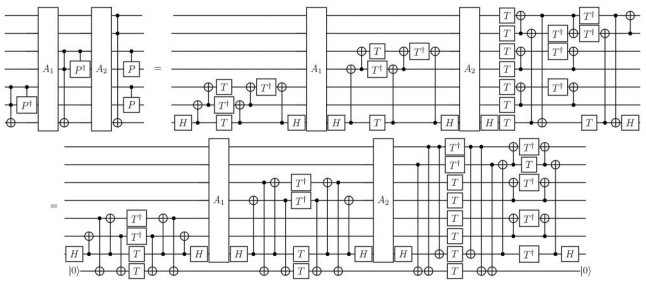

FIGURE 8 Case 3: a target line is shared, but control lines are not. For n (≥1) Toffoli-gates, T-depth can be reduced from 3n to $2n+1$. If one work qubit is added, T-depth from $2n+1$ could be reduced to $n+1$

3.1.4 | Case 4: One control line is shared, but the rest of the parts share the lines by crossing each other

In the fourth case in Figure 5, we should apply our method a little differently from other cases. That is, T and $T^\dagger$ gate should be used instead of CP and $CP^\dagger$ gates (Figure 9). The CP gate cannot be used due to the existence of H gates constituting Toffoli-gates. The T gate and H gate are not commutative [17, 21]. So, the CP gate composed of T gates cannot be exchanged with the H gate. To avoid wires with H gates, we have to use

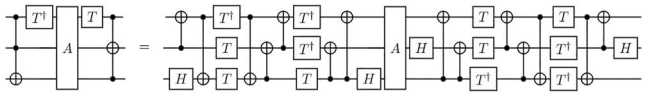

FIGURE 9 The fourth case in Figure 5. T-depth is reduced from 6 to 4 like the first three cases. But T-count cannot be reduced to 8

the T and $T^\dagger$ gate instead. In these cases, T-counts cannot be reduced to 8.

3.1.5 | The other six cases

For the remaining six cases, we can reduce T-depth by applying our method similar to the four cases mentioned above. In the fifth case, as in the fourth case, T and $T^\dagger$ gates should be used. Two lines can be considered at this time (Fig. 10). If subcircuit A and T (or $T^\dagger$) gates are interchangeable on both wires, T-depth can be reduced to 4. If an exchange is not possible on one line, it can be reduced to 5. In the sixth case, the Toffoli-gate on the right becomes a $C^2(-iX)$ gate, and the CP gate that passes through subcircuit A from right to left shares T-Depth with the Toffoli-gate on the left (Figure 10). We cannot reduce T-count to 8 at this time. In the seventh case, since the target line is shared as in the first and third cases, the possibility of performing the second reduction process mentioned below while reducing T-count is higher than in other cases (Figure 5). The eighth case is similar to case 6. In the ninth and tenth cases, a process of sharing T-depth should be used as in case 3 because the control lines are not shared at all.

3.2 | Second T-depth reduction process

In the previous section, we showed that the $C^2(iZ)$ gate or the $C^2(-iZ)$ gate is made of five CNOT gates [17]. On the other hand, in Ref. [31], 6 CNOT gates were used to make the $C^2(iZ)$ gate. The advantage of adding one CNOT gate is that it can prevent one T or $T^\dagger$ gate from being surrounded by CNOT gates (Figure 11). The basis variable corresponding to this T (or $T^\dagger$) gate is x_3 (or $-x_3$).

Now, let $\omega = (-1)^{1/4} = e^{i\pi/4}$ and consider the first case in Figure 5. As shown in Figure 12, let us combine subcircuit A and two H gates to form a large subcircuit B. If B is interchangeable with T gate, T-depth can be reduced from four to two. As a result, the $C^2(-i\omega^{-1}X)$ gate and $C^2(i\omega X)$ gate will exist on both sides of subcircuit A, respectively. The second reduction in this way can be applied similarly to all cases, except for the fourth and fifth cases. In the fourth and fifth cases, as it can be seen from Figures 9 and 10, even if a pair of T and $T^\dagger$ gates are deleted, T-depth cannot be reduced. That is, unlike all other eight cases, one T gate or $T^\dagger$ gate does not form one T-depth after finishing the previous first T-depth reduction process. Therefore, if T and $T^\dagger$ gates were used in the first reduction process, the second reduction techniques would become meaningless. If we want to perform this second reduction, we should perform the decision algorithm mentioned below in subcircuit B instead of in A.

Certainly, from the overall circuit point of view, the T-count remains unchanged and the overall T-depth can be lowered through T-depth sharing. This is because, in general, various Toffoli-gates appear in different cases in the entire circuit. However, it is not likely that this second T-depth reduction technique while reducing T-count is performed. As mentioned earlier, since the H gate and T gate are not interchangeable, the H gate reduction process mentioned below must succeed in two H gates on both sides of subcircuit A. Also, as mentioned in the next subsection, the state of the basis variable corresponding to the third line in Figure 12 should not be changed. The simplest situation that holds these two conditions is that there is no gate in subcircuit A on the third line, where two H gates exist. Then, since two H gates are inversely related to each other, they disappear obviously, and this second T-depth reduction process while reducing T-count can be performed.

A typical example of performing this second T-depth reduction process is the k-controlled Toffoli-gate (C^kNOT gate, $k \geq 3$). This gate can be decomposed into $4(k-2)$ Toffoli-gates if $k-2$ dirty borrowed qubits (qubits in arbitrary states) exist by Lemma 7.2 in Ref. [32]. For instance, we can see the process of decreasing T-depth of C^4NOT gate in Figure 13, and Figure 4 in Ref. [31]. They deleted some H gates and placed CP and $CP^\dagger$ gates appropriately so that they could be self-evidently exchanged with subcircuit A. In other words, the first T-depth reduction process that we will be discussing

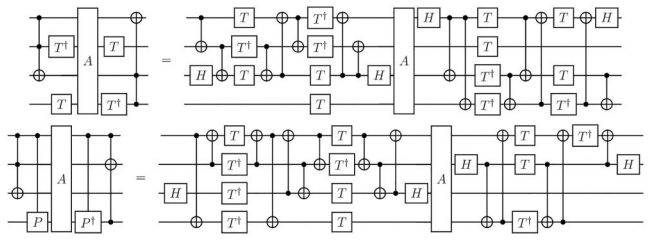

FIGURE 10 The fifth and sixth cases in Figure 5. In the fifth case, the CP and $CP^\dagger$ gate cannot be used. In the sixth case, a pair of CP and $CP^\dagger$ gates can be used, but the T-count cannot be reduced to 8

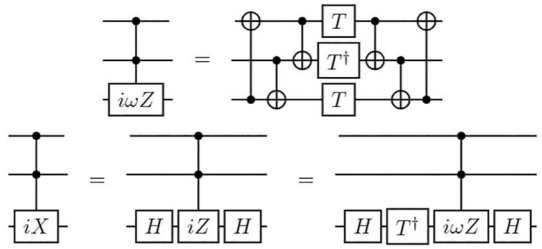

FIGURE 11 A doubly controlled $i\omega X$ gate [31]. A doubly controlled iX gate can be decomposed into two H gates, one $T^\dagger$ gate and a doubly controlled $i\omega Z$ gate

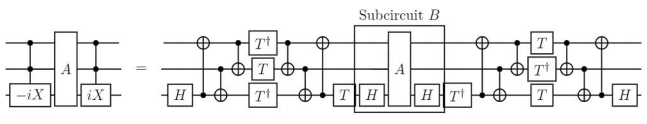

FIGURE 12 Second T-depth reduction process. If we want to perform this second reduction, we could perform the decision process mentioned below in subcircuit B instead of in subcircuit A. That is, it is necessary to check whether subcircuit B and T or $T^\dagger$ gate are interchangeable

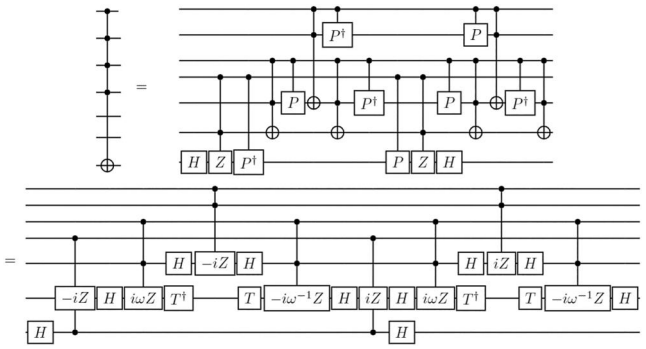

FIGURE 13 T-depth reduction process in the C^4NOT gate [31]. The positions of CP and $CP^\dagger$ gates in this figure are slightly different from Figure 4 in Ref. [31]. As a result, the same T-depth 12 is obtained

can be performed easily. The second T-depth reduction process can also be performed. At this time, they explained that the reason that the outermost extra phase gates (a pair of T and $T^\dagger$ gates) can be erased is that the state of dirty borrowed qubits is restored in Figure 4 in Ref. [31]. We can state in our way why these gates can be deleted. That is, the quantum subcircuit between these two extra phase gates is interchangeable with T or $T^\dagger$ gates. After the exchange, T and $T^\dagger$ gates become adjacent to each other and are erased. In summary, for every k $\geq$ 3, if there are $k-2$ qubits in arbitrary states, then the C^kNOT gate can be implemented as a circuit with T-depth $4(k-1)$ [31, 33].

In practice, it is very difficult to find a circuit implemented with only two Toffoli-gates. Pairs of Toffoli-gates can be selected in various ways in the quantum circuit, and T-depth can be further reduced without changing T-count through T-depth sharing as in case 3. The process of applying these first and second reduction techniques will be looked at in quantum adder circuits in the next section.

3.3 | Exchangeability determination algorithm

We consider the characteristics of subcircuit A interchangeable with the CP gate before presenting the process of determining exchangeability. The CP gate works as follows.

$$Controlled-P:|x_1x_2\rangle \rightarrow e^{\frac{\pi i}{4}2x_1x_2}|x_1x_2\rangle \qquad (10)$$

It can be seen that x_1 and x_2, which are basis variables corresponding to each wire, are maintained as they are, and $i^{x_1x_2}$ is generated. For the subcircuit A to be exchangeable with the CP gate, the above operation must be possible as it is. That is, the global phase $i^{x_1x_2}$ should be generated even after passing through the subcircuit A first.

Now, we present a method to determine the exchangeability between subcircuit A and CP gate. More precisely, without changing the overall operation of the entire circuit, we determine whether a part of subcircuit A can be converted into a circuit that can be exchanged with the CP gate. After confirming the possibility of conversion, some kind of restoration work is required so that the entire operation does not change. It was mentioned earlier that a circuit consisting of the H gate, X gate, R_z gate, and CNOT gate would be considered, and our technique is applicable even if some control parts of the Toffoli-gate are off. We will use variants of the optimisation subroutines in Ref. [14]. The following subroutines are executed sequentially. The order of subroutines has been carefully chosen.

3.3.1 | Step 0: Pruning procedure to identify RoI (Region of Interest) in subcircuit A between the CP gate and Toffoli-gate

We present a modified pruning procedure as a pre-processing step. This procedure identifies RoI (Region of Interest) in subcircuit A. After completing this procedure, we will see if RoI can be converted into a circuit that can be exchanged with a CP gate. We will try to modify the pruning procedure presented in the previous section. Again, our first concern is whether the CP gate can move to the left of the Toffoli-gate. That is, the values of the basis variables corresponding to the two lines with the CP gate should be maintained after passing through subcircuit A first. Gates that do not have any effect on the changes in the values of these basis variables can be, for example, CNOT gates that do not share a line with a CP gate. Conversely, all single-qubit gates and two-qubit gates located on the two lines with CP gates are gates that can be affected. Our main lines of interest are the two lines with CP gates. This modified procedure complies with the following rules.

- We designate the CP gate as the starting point. In the situation where the CP gate, the subcircuit A (the intermediate region), and the Toffoli-gate are sequential, our pruning procedure will only traverse to the right. The Toffoli-gate is the endpoint.
- The basic termination border is determined by two lines with CP gates between the timeline with the CP gate and the timeline with the Toffoli-gate.
- Unlike the original pruning procedure, even if an H gate is encountered while traversing, this gate is passed. When the R_z or X gates are met, it is passed through the same as the original pruning procedure.
- It should be carefully considered when meeting with CNOT gates. If one of the two lines with a CP gate shares only the control part, we make an exception and continue crossing. However, if the target part also belongs to the termination border, it can be passed. The reason for the exception of the CNOT gate with a target part outside the termination border is clear. This is because it does not affect the phase generation of the CP gate at all. If only the target part of the CNOT gate is shared with one of the two lines with the CP gate, it is passed, and the control part is designated as the anchor point. Through these anchor points, the termination border is widened. Even if new connectivity cannot be created, it is passed differently from the original pruning procedure.

- When traversing is completed on two lines with CP gates, traversal is performed with the same logic at anchor points. Anchor points also traverse only to the right and traverse to the timeline of Toffoli-gate. If we encounter the CNOT gate that we set as an exception, we skip it and continue traversing. All gates passed during traversing constitute RoI (or RoI0), which is a part of subcircuit A.

Our pruning procedure is mostly the opposite of the original pruning procedure. Again, the reason why it is different from the original pruning procedure is that the reason for doing this procedure is different. The original pruning procedure aims to designate the subcircuit as wide as possible for merging between R_z gates in the subcircuit created through the procedure. On the other hand, our pruning procedure is used as a pre-processing step to unambiguously form (identify) RoI between the CP gate and the Toffoli-gate. That is, since it is used to exclude gates that do not cause any change in the basis variables and do not affect them at all, we try to designate RoI as small as possible.

We can put the carry function block in the HRS adder as an example to show the sure effect of this pruning procedure (Figures 23 and 3 in [11]). The presented circuit consists of a total of ten Toffoli-gates. We want to specify exactly RoI between the first Toffoli-gate and the last Toffoli-gate. The CP gate is placed on two lines with the two control parts of the first Toffoli-gate. Since subcircuit A consists of many Toffoli-gates, it may be thought that it is very difficult to determine whether the T-depth is reduced, but it is very simple. Through the pruning procedure presented by us, it can be easily confirmed that the CNOT gate at the bottom of the centre is an exception. As a result, RoI created by excluding this CNOT gate can become an identity circuit. This is because gates placed in inverse relationships with each other can be sequentially deleted. Therefore, it can be seen that the total T-depth composed of the first and last Toffoli-gates can be reduced from 6 to 4 without performing the main process after the pruning procedure.

If you look at the example Figures 16 and 17, RoI (RoI0) made through the pruning procedure is indicated by a dotted line.

3.3.2 | Step 1: X gate propagation

In the first step, move all X gates to the right using X gate propagation [14, 26] (Figure 14). If an X gate has reached the left of the Toffoli-gate through X gate propagation, it can be moved to the right of the Toffoli-gate. At this time, an on-control part of the Toffoli-gate that meets the X gate is changed to an off-control part. Through this process, it is possible to ensure that there are no X gates in RoI. As mentioned earlier, our technique is applicable even if some control parts of the Toffoli-gate are off. RoI and the control parts of Toffoli-gate on the right may be slightly changed in this step. Of course, the overall operation does not change. RoI after this step is completed is called RoI1. Now, only CNOT, $R_z(\theta)$, and H gates exist in RoI1.

3.3.3 | Step 2: CNOT and R_z gate propagation

Let us consider the right side of RoI1. By using CNOT and R_z gate propagation, some CNOT gates can be moved to the right of the Toffoli-gate (Figure 15). That is, we take as many CNOT and R_z gates as possible out in RoI1. In the case of R_z gates, if they are adjacent to one of the two control parts of the Toffoli-gate, they can move to the right of the control part. Conversely, if the R_z gate is adjacent to the target part, it cannot move. There are a total of eight cases of relative positions between the Toffoli-gate and CNOT gate. CNOT and R_z gate propagation is performed sequentially from the CNOT gate or R_z gate adjacent to the Toffoli-gate. If the relative position during execution is one of the fourth, fifth, and sixth cases, this step is finished.

The reason for performing this step is to make the calculation in steps three and four easier. The smaller the number of CNOT gates in steps three and four, the faster we can move to the next step.

Let RoI after this step be called RoI2. Of course, after completing this step, if there are no gates on the two lines with CP gates, that is, if it becomes an identity circuit, it can be said that subcircuit A can be converted into an interchangeable circuit. If any gates remain in RoI2, proceed to the next step.

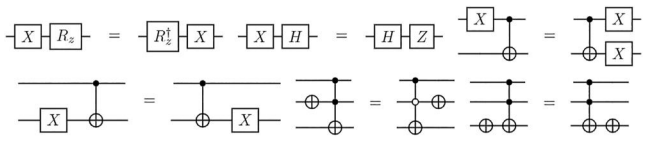

FIGURE 14 Commutation rules in X gate propagation [14]. If the X gate reaches the right end of subcircuit A, it is also exchanged with the Toffoli-gate on the right. At this time, if the X gate is adjacent to the control part of the Toffoli-gate, the control part is flipped

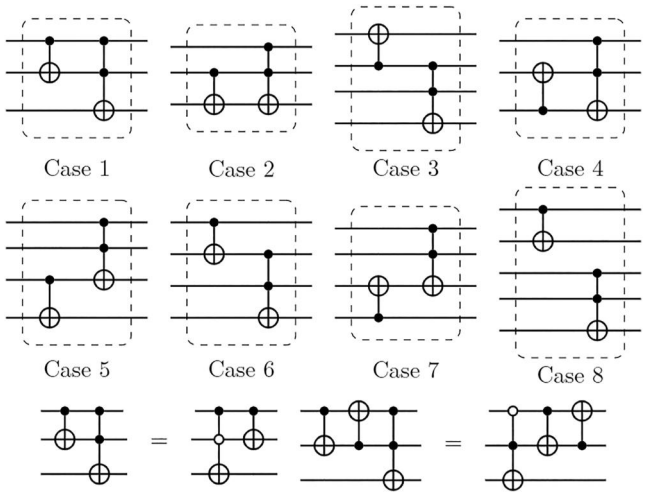

FIGURE 15 Eight cases of relative positions for the controlled-NOT gate (CNOT) gate and Toffoli-gate. In cases 4, 5, and 6, the CNOT propagation process cannot be performed to the right. That is, the CNOT gate cannot be exchanged with the Toffoli-gate. (Exchange is possible only by adding one more Toffoli-gate.) Of the remaining five cases, only in the first case, one of the control parts of the Toffoli-gate is flipped [30]

3.3.4 | Step 3: Hadamard gate reduction

The CP gate consists of two CNOT gates, two T gates, and one $T^\dagger$ gate. As mentioned earlier, the H gate is not commutative with the T gate. Therefore, H gates on lines with the CP gate in A must be reduced and erased using the circuit identities shown in Figure 4 in Ref. [14] and Figure 8 in Ref. [34]. At this time, if X gates occur while performing H gate reduction, X gate propagation is performed as in the first step. Also, using Figure 5 in Ref. [14] and the rules of Ref. [35], the positions of R_z gates and CNOT gates are changed to further perform H gate reduction. If two adjacent gates are placed in an inverse relationship with each other during position movement, they are deleted. The movement of all gates is finally performed for H gate reduction on lines with the CP gate in this step. Let us call the region after this step RoI3. If at least one H gate remains in the two lines with the CP gate after this step, it can be said that subcircuit A cannot be converted into a circuit that is interchangeable with the CP gate. If the H gates in both lines disappear, go to the next step.

3.3.5 | Step 4: Z-rotation gate ($R_z(\theta)$ gate) cancelation

The overall operation did not change through the previous three steps. Now, after completing step 4, the overall operation may be different. If it is different, it is necessary to restore RoI3 after this step or step 5 is finished. Now, only R_z gates and CNOT gates exist in RoI3. We are interested in whether the basis variables x_1 and x_2 corresponding to the two lines on which the CP gate lies do not change. Therefore, $R_z(\theta)$ gates in RoI3 that do not affect the change of these two basis variables can be ignored. After erasing all R_z gates, we can reduce the number of remaining CNOT gates using rules in Ref. [35].

In this step, the R_z gate cancelation operation can change the entire operation for the first time. If RoI3 becomes an identity circuit RoI4 after this step is over, it can be said that subcircuit A can be converted into a circuit that is interchangeable with the CP gate. If it is determined that the exchange is possible, restore gates that are erased in this step. They are restored to the 'original position'. If at least one CNOT gate remains in RoI4, go to the next step.

3.3.6 | Step 5: Discrimination between the remaining CNOT gates

In this last step, we check whether the basis variables x_1 and x_2 have changed after passing through RoI4 where only CNOT gates are left, that is, whether we can make the desired phase. The change of basis variables is examined, considering the basis variables corresponding to all lines in subcircuit RoI4. After passing through RoI4, if x_1 and x_2 do not exist in the two lines with the CP gate, we cannot create the desired phase. That is, subcircuit A cannot be converted into a circuit that is interchangeable with the CP gate. If the states are maintained even if x_1 and x_2 pass through RoI4, then RoI4 is a circuit that can swap positions with the CP gate.

3.3.7 | Examples

Let us take a concrete example with Figure 16, and the basis variables corresponding to each line be x_1, x_2, x_3, and x_4. Through the pruning procedure, the two CNOT gates in the middle are not included in RoI. Through X gate propagation, one X gate is moved to the right of the Toffoli-gate. Then, 3 R_z gates and 1 CNOT gate are moved to the right of the Toffoli-gate through CNOT and R_z gate propagation. Then, the H gate reduction is performed. We can clear all H gates on the first and second lines. Now, only $R_z(\theta)$ gates and CNOT gates remain in RoI3. After erasing all R_z gates (T gate), reduce the number of CNOT gates using the rules in Ref. [35]. As a result, only 3 CNOT gates remain and x_2 is changed to $x_2 \oplus x_4$. Since the basis variables x_1 and x_2 do not appear in the first and second lines, this subcircuit A cannot be converted into a circuit that can be exchanged with the CP gate. The CP gate cannot create the phase $i^{x_1 x_2}$ after exchanging with RoI4.

Although RoI4 is not interchangeable with the CP gate, it is not that the T-depth formed by two Toffoli-gates cannot be reduced. We can see that there is no gate on the first line of subcircuit RoI4. That is, subcircuit A can be converted into a subcircuit that can be exchanged with the T gate if the T gate exists on the first line. When using T and $T^\dagger$ gate, T-depth can be reduced to 4 as when using the CP gate, but T-count cannot be reduced to 8.

Let us look at another example where the exchange succeeds this time. Figure 17 is very similar to Figure 16, but one

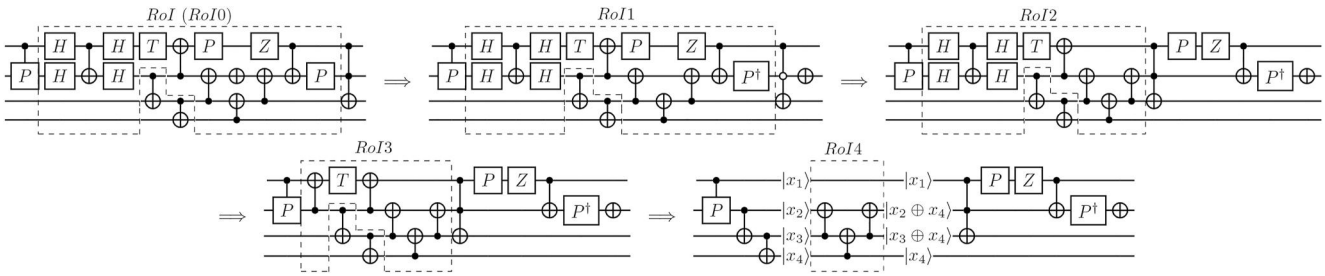

FIGURE 16 Example: subcircuit A (the intermediate region) that cannot be converted into a circuit interchangeable with the controlled-P gate (CP gate). However, this subcircuit A is interchangeable with the T gate if the T gate exists on the first line

CNOT gate is deleted in the middle. RoI can become identity circuit RoI4 through the above process. Therefore, R_z gates and CNOT gates deleted in step 4 are restored to their original positions. Then, the positions of CP gate and restored RoI3 are exchanged. Through this process, T-depth is reduced, but the overall operation does not change.

3.4 | Remark and caution

3.4.1 | Remark: An unpaired Toffoli-gate can only attempt T-depth sharing

Some readers may have recalled relative phase decomposition in NISQ (noisy intermediate-scale quantum computing) circuits while reading this paper [36]. Or one might wonder why it does not cover the case where a single Toffoli-gate is alone. As mentioned in the introduction, we aim to create an efficient quantum cryptosystem circuit to verify security strength. A typical quantum attack algorithm used in this case is Grover's algorithm [3]. In this case, Grover's algorithm uses the phase kick-back technique. If there is only one Toffoli-gate, and it is combined with the $CP^{\dagger}$ gate, phase -i occurs, and it cannot be removed. A quantum circuit that does not maintain the input phases $+\frac{1}{\sqrt{2^n}}$ for the number n of bits of the superposed input value cannot be used in Grover's algorithm. Therefore, when modifying the circuit to reduce T-depth and T-count, we must take care that the phase $+\frac{1}{\sqrt{2^n}}$ of each state is maintained after passing through all gates in the quantum crypto-system circuit. So relative phase decomposition cannot be used.

Of course, even if the Toffoli-gate is alone, the T-depth of the entire circuit can be reduced through T-depth sharing. In other words, T-depth made by other Toffoli-gates can be shared. In this case, the T-count does not decrease like in case 3.

3.4.2 | Caution: There are cases where two or more Toffoli-gates can be converted into one Toffoli-gate.

If subcircuit A has a special shape, then Toffoli-gates on both sides can be reduced to one. For example, assume that two Toffoli-gates share both two control lines and one target line as the second case in Figure 18. At this time, suppose that subcircuit A is a CNOT gate that uses one of two control lines of the Toffoli-gates as a target line and the fourth line as a control line. Then two Toffoli-gates can be reduced to one Toffoli-gate having a different control line (Figure 18). That is, there is a situation in which the Toffoli-count can be decreased. In this case, T-depth can be reduced from six to three instead of four. There are various other situations in which the Toffoli-count can be reduced (Figure 18, [30, 37, 38]).

4 | APPLICATION 1: APPLYING T-DEPTH AND T-COUNT REDUCTION TECHNIQUE TO QUANTUM ADDERS

Now we apply the technique presented in the above section to some quantum adders. In all five quantum adders, T-depth reduction rate is more than 33%. These changed T-depth values can be seen in Table 2.

4.1 | CDKM adder

First, let us take a look at the CDKM adder. As can be seen from Figure 6 in Ref. [7], all Toffoli-gates, except for one, can be paired by two and share two control lines and one target line. In the central part of the circuit, RoI is composed of X gates and CNOT gates. The situation corresponds to case 1 in Figure 5. T-depth 1 can be further reduced through T-depth sharing near the centre. As a result, in modulo 2^{32}, Toffoli-depth is 61 and T-depth is 64. The T-depth reduction rate is about 65%. Unfortunately, this CDKM adder is not used when we construct the SHA-256 quantum circuit. The reason is that it is an inferior circuit to the TK-v2 adder below. The TK-v2 adder uses one work qubit like the CDKM adder, and Toffoli-depth is also the same, but T-depth is smaller.

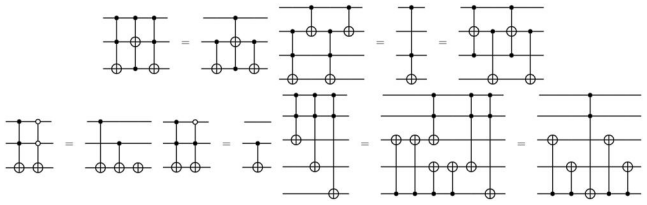

FIGURE 18 Toffoli-count reduction rules [30, 37, 38]

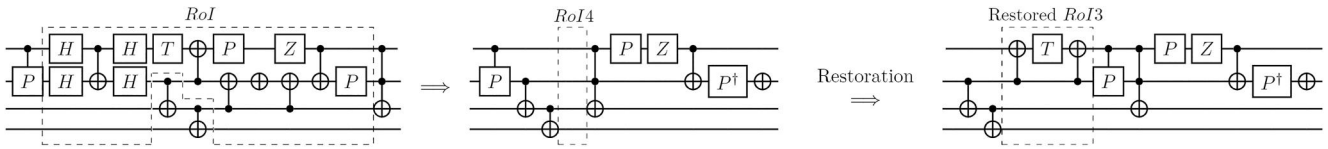

FIGURE 17 Example: the subcircuit A that can be converted into a commutative circuit with the controlled-P gate (CP gate). Contrary to the previous example (Figure 16), one controlled-NOT gate (CNOT) gate has been removed in subcircuit A (RoI). After step 4, RoI4 is an identity circuit, so T-depth reduction is possible. We restore $R_z(\theta)$ gates, and CNOT gates deleted in step 4 to the 'original positions' in RoI4 so that the entire operation is maintained. The restored RoI3 swaps positions with the CP gate, and as a result, the CP gate is located on the left side of the Toffoli-gate

4.2 | VBE adder

The VBE adder consists of carry function blocks and sum function blocks, and you can see the circuit in Figure 1 in Ref. [9] and Figure 19 when one operand is a constant. In the SHA-256 quantum circuit to be mentioned in the next section, the VBE adder is used in a situation where one of the two operands is a constant. In the VBE adder, most Toffoli-gates constitute case 1 such as the CDKM adder. Since RoIs are identity circuits, T-depth reduction is of course possible. For this VBE constant adder, Toffoli-depth varies according to the LSB (Least-Significant bit) value of the constant. Toffoli-depth is 57 or 59 and T-depth is 60 or 62 in modulo 2^{32}. In SHA-256, among 64 constants K_t ($t = 0, \ldots, 63$), there are 33 K_ts with LSB of 1 and 31 K_ts with LSB of 0. Consequently, the T-depth reduction rate is about 65%.

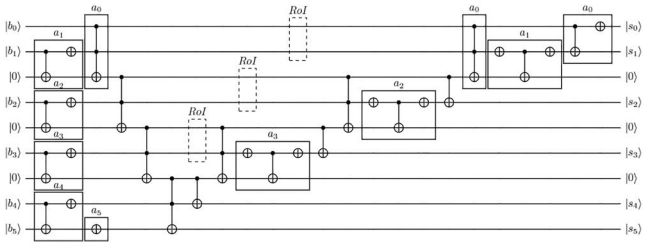

FIGURE 19 VBE constant adder circuit in modulo 2^6 [9]. Whether gates are included is determined according to bit values of A, which is a constant. In particular, if the LSB (Least-Significant bit) value of a is 0, two Toffoli-gates at the top of the circuit are not included

4.3 | TK adder

The TK adder is shown in Figure 5, and Figure 7 in Ref. [10] (Figure 20). We will not use this adder circuit as it is, but use a modified version by reducing the Toffoli-depth (Toffoli-count) using the Toffoli-count reduction rules (Figure 18). Let us explain the configuration of the modified circuit (Figure 21). First, to add the operands A and B, which are n-bit numbers, these values are stored in arrays A and B, respectively. The modified TK adder proceeds as (11).

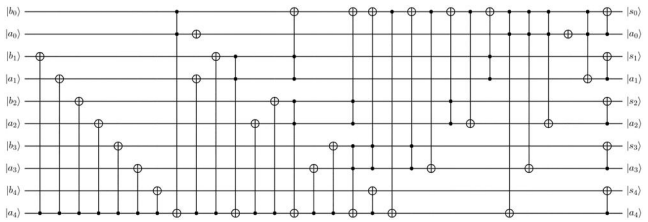

FIGURE 20 Original TK adder circuit in modulo 2^5 [10]

4.4 | HRS adder

The design of HRS (constant) adder is much more complicated than those of the three adders above [11] (Figure 22). Unlike

1. $For \quad 1 \le i \le n-2, \quad B[i] \oplus = A[n-1], \quad and \quad A[i] \oplus = A[n-1]. \quad And \quad then \quad B[n-1] \oplus = A[n-1].$
2. $For \quad 1 \le i \le n-2, \quad A[n-1] \oplus = A[i-1]B[i-1], \quad A[i] \oplus = A[n-1], \quad and \quad B[i] \oplus = A[n-1].$
3. $B[n-1] \oplus = A[n-2]B[n-2], \quad B[n-1] \oplus = A[n-1], \quad and \quad A[n-2] \oplus = A[n-1].$
4. $For \quad 2 \le i \le n-2 \quad in \quad reverse \quad order, \quad A[n-1] \oplus = A[i-1]B[i-1], \quad and \quad A[i-1] \oplus = A[n-1].$ (11)
   $And \quad then \quad A[n-1] \oplus = A[0]B[0]$
5. $For \quad 2 \le i \le n-1, \quad A[i-1] \oplus = A[n-1].$
6. $For \quad 0 \le i \le n-1, \quad B[i] \oplus = A[i].$

From now on, this modified TK adder will be called TK-v1 or TK-v2 or TK-v3 adder. The TK-v1 adder does not use work qubits at all like the original TK adder. The TK-v2 adder uses one work qubit, so after T-depth reduction operation is finished, T-depth of all $C^2(-iX)$ and $C^2(iX)$ gates can become one. The TK-v3 adder uses three work qubits so Toffoli-depth and T-depth can be the same by the Matroid Partitioning concept [21]. For the bit length n of the operand, Toffoli-depth of this adder is $2n - 3$. That is, when performing addition in modulo 2^n, these modified versions have the same Toffoli-depth as the CDKM adder [7].

Like the CDKM and VBE adder, most Toffoli-gates constitute case 1, and one Toffoli-gate in the centre cannot participate in the first T-depth reduction process. However, it can share one T-depth with Toffoli-gate (or $C^2(iX)$ gate) on the right. The T-depth reduction rate is about 33% in the TK-v1 adder.

VBE and TK adders, there is no need to use clean work qubits. Instead, we can use borrowed dirty qubits so that the states of dirty borrowed qubits are the same after the operation is finished. As module 2^{32} addition is performed, it consists of a total of five layers. Each layer consists of two carry function blocks (Figure 23) and two controlled-incrementer function blocks [39] (Figure 24). In the last layer, there is no Toffoli-gate.

In carry function blocks, we can perform a first T-depth reduction process and a second T-depth reduction process (Figure 23). All Toffoli-gates participated in our T-depth reduction work. Both processes are executed in case 1. However, in each process, the partners of Toffoli-gates constituting case 1 are different. In these function blocks, there are generally four Toffoli-gates that share two control lines and one target line. At this time, when performing the first T-depth reduction process, the first and fourth Toffoli-gates are paired,

and the second and third Toffoli-gates are paired. In the next second T-depth reduction process, the first and second Toffoli-gates are paired, and the third and fourth Toffoli-gates are paired. Also, in general, the top two Toffoli-gates in this function block cannot perform the second T-depth reduction process. Toffoli-count reduction was not performed on carry function blocks.

In controlled-incrementer function blocks, all Toffoli-gates constitute case 1. In addition, T-depth can be further reduced in the remaining Toffoli-gates except for the top two Toffoli-gates and the bottom two ones.

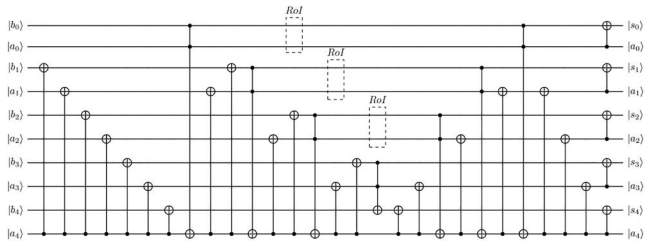

FIGURE 21 A modified TK adder circuit in modulo 2^5. We will use this better version of the TK adder in modulo 2^{32} instead of the original

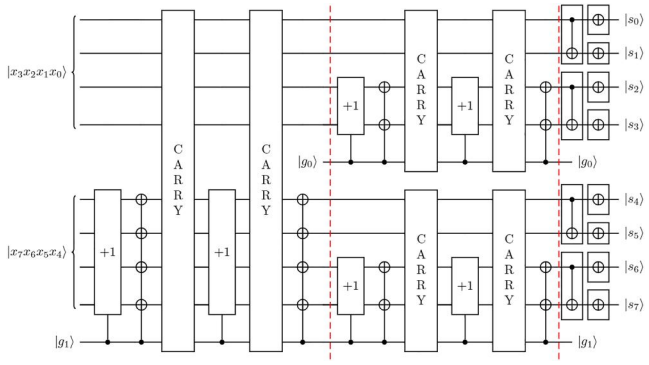

FIGURE 22 The HRS adder for $n = 8$. The operand X and the constant a are added to calculate the sum s in mod 2^8. This circuit has three layers. In the last layer, it consists of controlled-NOT gate (CNOT) gates and NOT gates, and execution is decided according to the value of a. For example, if the value of LSB a_0 is 0, the CNOT gate and NOT gate located at the top are not executed

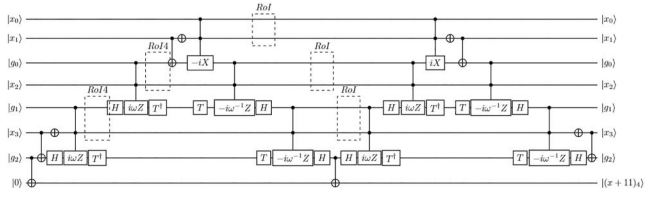

FIGURE 23 A carry function block in the HRS adder [11]. For bit-length $n = 4$, the operand X is added with the constant 1011_2. The output value is the Most-significant bit (MSB) of sum. When performing the first T-depth reduction process and second T-depth reduction process, two Toffoli-gates constituting Case 1 are selected differently. RoIs(RoI0s) and RoI4s shown in the figure are used in the first T-depth reduction process. g_0, g_1, and g_2 are dirty borrowed qubits

Since the circuit is created by the divide-and-conquer technique, operations are performed in parallel within the entire adder circuit. In mod 2^{32}, the HRS adder does not perform parallel execution only in the first layer. In the first layer, the process of generating and adding the 16th carry, c_{16}, is performed. At this time, $a_{15}\ldots a_0$ is used for the 32-bit constant A ($a_{31}\ldots a_0$). Therefore, like the VBE constant adder, the change in Toffoli-depth (T-depth) according to the LSB value of the constant a can be considered. If the LSB value of the constant a is 0, Toffoli-depth is decreased by two in the carry function blocks in the first layer. If we want to account for the change in Toffoli-depth in the second layer, the value of the 17th bit a_{16} as well as LSB must be 0. For convenience, only the first layer was considered to change Toffoli-depth.

As a result, in modulo 2^{32}, Toffoli-depth is 384 or 392 and T-depth is 424 or 432. Therefore, the T-depth reduction rate is about 63%.

4.5 | QCLA

The structure of QCLA will be briefly introduced first, and then the process of applying the T-depth reduction algorithm will be explained. QCLA is largely divided into an addition step and an uncomputation step (Figure 5 in [12]). In the addition step, it goes through P, G, C, P^{-1} rounds in sequence, and the order of rounds is reversed in the uncomputation step. Each P round and P^{-1} round consists of 3 layers in modulo 2^{32}. Except for the first layer in P round, the remaining two layers are processed in parallel with G round. In P^{-1} round, except for the last layer, the first two layers are processed in parallel with C round. Each G round and C round consist of 4 layers, respectively.

In the addition step, P and P^{-1} rounds constitute case 1. The same goes for the uncomputation step. We can use CP and $CP^{\dagger}$ gates to decrease T-count as well. Each G round in these steps constitutes Case 3 (Figure 8). Therefore, the T-depth reduction process can be performed in G rounds. Then, the Toffoli-gate, which forms the last layer in round G, can share T-depth with the first and second layers in round C. So T-depth reduction can be performed once more. Two C

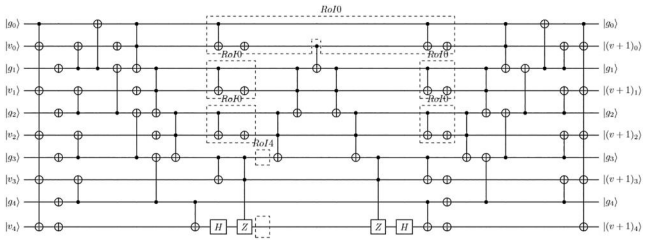

FIGURE 24 An incrementer for $n = 5$. $|g_i\rangle$ $(i = 0, \ldots, 4)$ are dirty borrowed qubits, and it shows the process in which the operand v expressed in 5 bits becomes $v + 1$. The controlled-incrementer for n is equivalent to one X gate added to the incrementer for $n + 1$ [39]. RoI0s and RoI4s shown in the figure are used in the first T-depth reduction process

rounds in each step constitute Case 1, so T-depth is reduced by using a pair of T and $T^\dagger$ gates. After that, T-depth can be shared among the layers as in the G round.

After doing all of the above, in the addition step, $C^2(-iX)$ gates constituting the first layer of P round and $C^2(-iX)$ gates forming the first layer of G round can share one T-depth. Similarly, $C^2(iX)$ gates constituting the last layer of P^{-1} round and $C^2(i\omega X)$ gates (or $C^2((-i\omega^{-1}X))$ gates) constituting the last layer of C round can share one T-depth. T-depth can be shared in the uncomputation step as well. Some Toffoli-gates do not belong to the addition step and uncomputation step in the front part and the back part of QCLA. These constitute case 1, and T-depth reduction can be performed by using T and $T^\dagger$ gates. As a result, the total T-depth becomes 24. Toffoli-depth is 22, so the T-depth reduction rate is about 64%.

4.6 | Relationship between Toffoli-count and T-count

What we realised while doing the process of reducing T-depth of the above five adder circuits is that the optimised T-depth can be obtained by reducing T-count as much as possible and then selecting the appropriate location of T and $T^\dagger$ gates. Based on the above method, CDKM, VBE, TK adder, and QCLA with reduced T-depth are circuits with T-depth optimised by the Matroid Partitioning concept [21]. However, it cannot be said that the HRS adder has become a circuit optimised for T-depth (and T-count). Because the HRS adder did not perform T-depth reduction for the entire circuit. It was only executed within the function blocks. Also, the HRS adder is a circuit in which the Toffoli-count can be reduced (Figure 18). The reason that the Toffoli-count operation is not performed is that if T-depth reduction operation is performed after the Toffoli-count operation is executed, T-depth is the same, but T-count comes out with a larger value. In other words, it does not seem that Toffoli-count reduction necessarily leads to T-count reduction. Investigating the correlation between Toffoli-count reduction work and T-depth and T-count may be a future research task.

5 | APPLICATION 2: USING REDUCED QUANTUM ADDERS IN OUR NEW QUANTUM SHA-256 CIRCUIT

5.1 | Function blocks in the SHA-256 quantum circuit

Now, we configure the SHA-256 quantum circuit using the above adders with reduced T-depth. In Figure 25, you can see the internal function blocks composing the SHA-256 quantum circuit we present. Work qubits are not shown in the figure. In the ADD function block, of course, the adder circuits discussed in the previous section are used. In the case of the Maj function block, the circuit of Ref. [7] is used as it is. On the other hand, the Ch function block is our newly created function block, which consists of one CNOT gate and one Toffoli-gate. Both function blocks do not use work qubits and have Toffoli-depth 1. These function blocks can be seen in more detail in Figure 26.

Σ_0, Σ_1, σ_0, and σ_1 function blocks receive a 32-bit string as an input value and output a 32-bit string, respectively (Table 1). Since these quantum circuits can be constructed using only CNOT gates in the reverse direction of the PLU decomposition [4], T-depths are all zero. The output values of these four function blocks are all used as the operands of addition and then restored to their original input values through the inverse operation in quantum circuit. Since they consist only of CNOT gates, they do not significantly affect the performance speed of the circuit. For example, in the case of the σ_0 function block, this quantum circuit with Depth 50 can be made by using a total of 193 CNOT gates. Although 20 swapping occurs, we do not need to change the swapping process to three CNOT gates because we only need to change the positions of the lines. If it is converted to three CNOT gates, a total of 253 CNOT gates are required.

5.2 | SHA-256 quantum circuit implementation

There are three main ideas introduced when designing the SHA-256 quantum circuit. We introduced a path balancing technique that makes some operations run in the next round. And instead of making T_1 (or $T_1 + T_2$) first, we made $d + T_1$

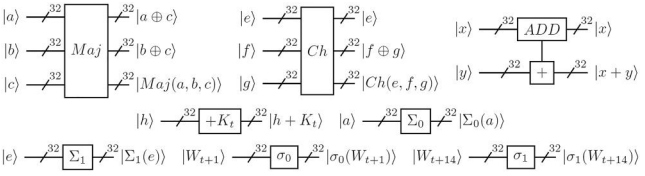

FIGURE 25 Function blocks in SHA-256 quantum circuit. In Maj and Ch function blocks, phases are not indicated ($(-i)^{(a\oplus c)\wedge(b\oplus c)}$ and $(-i)^{e\wedge(f\oplus g)}$)

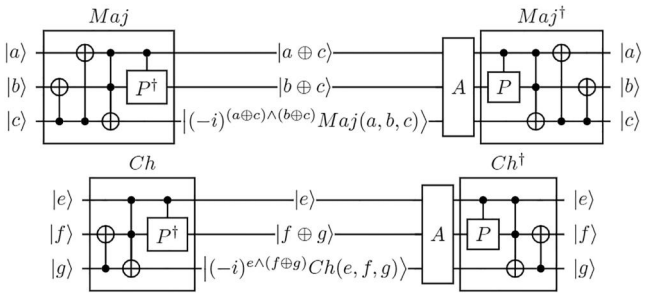

FIGURE 26 Maj function block and Ch function block in the SHA-256 quantum circuit. In the SHA-256 circuit, the qubits' states of each wire are maintained after passing through subcircuit A (RoI). Thus, our T-depth reduction process is possible in these blocks

first to reduce the length of the critical path of the entire circuit. In addition, T-depth is reduced as much as possible by providing enough work qubits to all adders performing in parallel.

In our proposed quantum circuits, we perform a total of 11 additions and 1 subtraction per round. This subtraction is used to restore the values of the internal variables e, f, and g in the circuit. When performing subtraction, we will use the subtractor version of TK-v1 or TK-v3 adder or QCLA, so T-depth and T-count are not different from one of these adders. The adder circuits used in the proposed circuit are VBE, TK-v1, TK-v2, TK-v3, HRS adder and QCLA mentioned in the previous section (Table 2). In the previous studies, only one adder circuit was used to construct each SHA-256 quantum circuit. We present circuits with reduced Width or T-depth by arranging several types of adder circuits in appropriate positions.

A path balancing technique is introduced when constructing the proposed circuits. This technique repositions some operations so that they can be performed in the next round. As a result, the round function algorithm consists of 65 rounds instead of 64 rounds in our proposed circuits. The difference between the classic circuit and the proposed quantum circuits is that the classical circuit repeats 64 times for one message block, whereas the proposed quantum circuits repeat a total of 65 times. In Figure 27, the function blocks painted in red are added in round 2, black in round 4, green in round 5 and yellow in round 6. The upper area is the round function algorithm and the lower area is the message schedule algorithm. Function blocks painted in red at the top of the circuit run from round 2 to round 65. In the final round 65, only the function blocks painted in red at the top are executed. For example, the internal variables $a = T_1 + T_2$, and $b =$ a used in round t are created just before the Maj function block in the upper right of the circuit is executed. In other words, they are created almost at the end of round t. After each round, the position of qubits is adjusted through swapping according to the hash algorithm.

In each round, the critical path can be composed of only three adders. That is, we design circuits to execute in parallel three out of nine adders for each time slice in the round function algorithm. The critical paths of the quantum circuits created in previous studies [5, 6] consist of seven or 10 adders. Function blocks painted in blue are function blocks constituting a critical path with T-depth that determines the performance of a quantum circuit.

In the classical circuit, T_1, which is commonly required for $d + T_1$ and $T_1 + T_2$, is made first, but in our quantum circuits, $d + T_1$ is made as quickly as possible using W_t twice. If you look at Figure 27, you can see that W_t is used twice as an operand. In the round function algorithm, 8 adder circuits and 1 subtractor circuit are used, and in the message schedule algorithm, 3 adder circuits are used. Both sub-algorithms are processed in parallel. In round t, the adder circuit with a long vertical line located to the centre of Figure 27 is to add W_{t-1}, so it is one of the adders constituting the round function algorithm.

As mentioned above, W_t is used twice as an operand in round $t + 1$ and round $t + 2$. It should be noted that $\sigma_0(W_{t+1})$ is first added to W_t at round $t + 4$, not round $t + 3$. If it is added in round $t + 3$, it is serially processed with the part using operands in the round function algorithm, so T-depth may increase. We add W_{t+9} when round $t + 5$ and $\sigma_1(W_{t+14})$ when round $t + 6$. When adding these two operands to W_t, the order of these additions does not matter. As a result, at round $t + 6$, W_t becomes W_{t+16}. W_{16} is made in round six and W_{63} is made in round 53. The black parts no longer exist from round 52, the green parts from round 53, and the yellow parts from round 54. That is, the message schedule starts at round four and ends at round 53.

5.3 | Our proposed circuits

There are four circuits presented by us. Our four proposed circuits are named SHA-Z1, SHA-Z2, SHA-Z3, and SHA-Z4, respectively (Table 3). SHA-Z1 consists of a total of 768 qubits, and one HRS adder and eleven TK-v1 adders are used. If you look at Figure 27, constant K_t is added using the HRS adder at the front, and five TK-v1 adders are located in the same time slice. The remaining six TK-v1 adders are grouped by three for parallel processing. Of the 768 qubits, 256 qubits are used to represent the internal variables a, …, h in the round function algorithm, and 512 qubits are used in the message schedule algorithm. When we use the HRS adder, we can use idle qubits in the message schedule algorithm as the borrowed dirty qubits. The HRS adder and TK-v1 adder do not use clean work qubits at all.

SHA-Z2 is a circuit with a total of 797 qubits, which consists of one VBE adder and five TK-v1 adders and six TK-v3 adders. Compared with the previous version, the HRS adder is replaced by the VBE adder. Five TK-v1 adders are still in the same time slice. The remaining six TK-v3 adders using 3 work qubits are located in the second and third time slices (Our quantum circuits consist of three time slices in one round, and the reason is that the critical path that determines the T-depth consists of three adders.). In this circuit design, there are a total of 29 work qubits because the VBE adder needs 29 work qubits and TK-v1 adders do not use work qubits at all. The remaining six TK-v3 adders can be used in a form in which the T-depth is reduced as much as possible by using 3 (or 4) work qubits each. Since up to three TK-v3 adders operate at the same time, the number of work qubits required in second and third time slice is up to 9, respectively. In fact, it is not necessary to use the method presented in the previous section for T-depth reduction in the second and third time slices in SHA-Z2. This is because work qubits can be sufficiently provided. In this time, our method only reduces T-count. In the final round 65, there are 31 work qubits in our second proposed circuit, so all three additions can use the TK-v3 adder.

SHA-Z3 uses 927 qubits, of which 159 are work qubits. Comparing with SHA-Z2, five TK-v1 adders are replaced by TK-v3 adders and six TK-v3 adders are replaced by QCLAs.

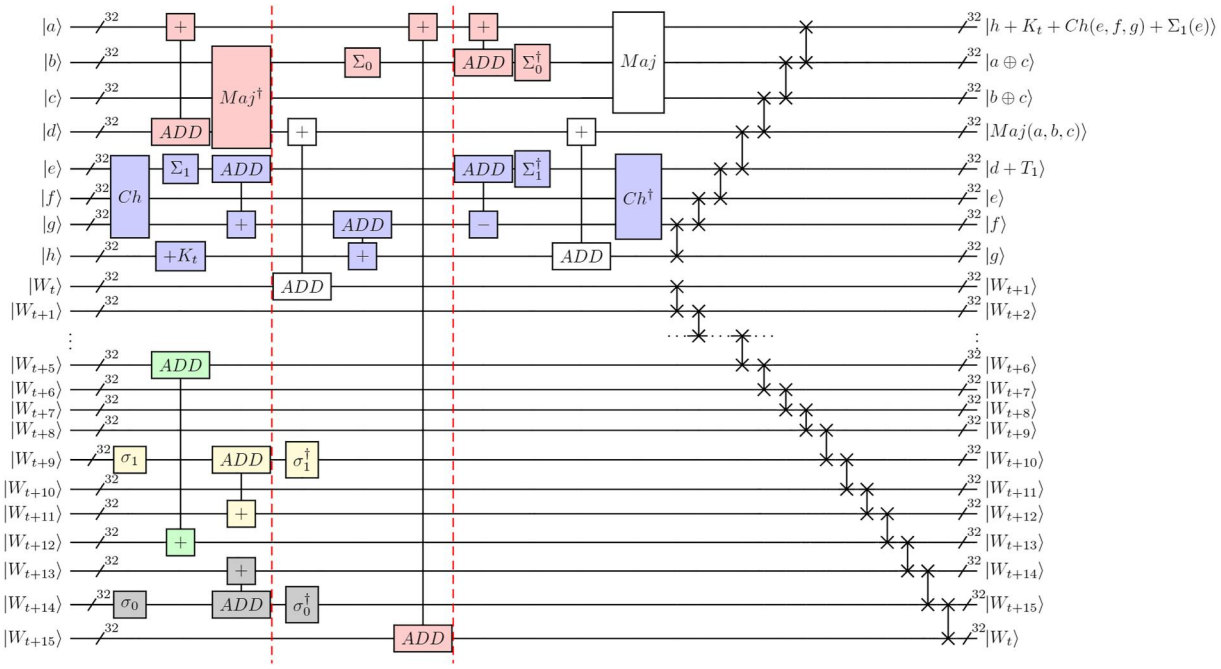

FIGURE 27 SHA-256 quantum circuit. The function blocks painted in red are added in round 2, black in round 4, green in round 5 and yellow in round 6. Function blocks coloured in blue constitute a critical path in one round. For each round, the critical path consists of only 3 quantum adder circuits. Two-qubit gates at the end of each round are SWAP gates

SHA-Z4 uses three TK-v2 adders and nine QCLAs and uses a total of 194 work qubits. In the round function algorithm, all additions and subtractions are performed only with QCLA, and the message schedule algorithm is performed only with the TK-v2 adder. Since QCLA is used when adding constants, 32 additional work qubits are required to hold constants.

SHA-Z1's critical path consists of one HRS adder, two TK-v1 adders, Ch function block's inverse and Σ_1 function block's inverse. On the other hand, the critical paths of the remaining three circuits consist of Ch and its inverse, Σ_1 and its inverse, and three adders. T-depth of these critical paths are 670 (or 678), 248, 113, and 76, respectively. Recall that T-depths are not optimised for the HRS adder to which our T-depth reduction technique is applied. Also, we did not perform our T-depth reduction technique on the entire circuits. Therefore, T-depth of these SHA-256 circuits may be further reduced.

5.4 | Quantum circuit resources comparison

SHA-Z1 has been implemented with fewer qubits than any previous circuits. SHA-Z2 has a smaller Toffoli-depth than SHA-C2 and SHA-C3 implemented by QCLA with Toffoli-depth $O(\log\ n)$. SHA-Z3 uses fewer quantum resources than SHA-C5 and SHA-C6, so it can be said that it is a superior circuit. SHA-Z4 is the only circuit with T-depth less than 5000, so it is the most time-efficient SHA-256 circuit. If more work qubits were available, T-depth 4936 could be reduced to 4418 which is Toffoli-depth.

6 | CONCLUSION AND FUTURE WORK

In the fault-tolerant model (QECC circuit), T-depth is an important factor that determines the running time of a quantum circuit. We have proposed a novel method that reduces T-depth (T-count) when there are two Toffoli-gates and a subcircuit between them. Our method is largely divided into two processes, and for each process, our presented algorithm is applied once. Each process was named the first T-depth reduction process and second T-depth reduction process, respectively, and the algorithm was named the exchangeability determination algorithm. Our main idea is to check whether the subcircuit and the CP gate (or T gate) are interchangeable. More precisely, without changing the overall operation of the entire circuit, we determine whether a part of subcircuit A can be converted into a circuit that can be exchanged with CP gate. In our process, T-depth can be further reduced without reducing T-count through T-depth sharing.

By using our method, CDKM adder, VBE adder, TK adder, HRS adder, and QCLA became circuits in which T-depth was reduced by more than 33%. In most adders, two Toffoli-gates are paired to share both control lines and a target line, so the T-depth can be greatly reduced. As a side note, we made a modified TK adder. This adder is an adder that cannot use any work qubits, and Toffoli-depth is the same as that of the CDKM adder. These adders are placed appropriately for use in our SHA-256 circuits.

In addition, we present a new construction of SHA-256 quantum circuit so that the critical path consists of only three adders. A total of four versions are presented with a new

design. There are two main ideas used in the circuit design: 1) some operations are performed in the next round through the path balancing technique, and 2) $d + T_1$ is created first among the values T_1, $d + T_1$, and $T_1 + T_2$. In previous works, critical paths consisted of seven or 10 adders. However, by introducing these two ideas, the critical path can be made to consist of only three adders. This circuit's construction and performance are much better than SHA-256 circuits in previous work. During design, we made a new Ch (Choice) function block. This function block is made of one CNOT gate and one Toffoli-gate.

Figure 5 presents 10 cases in which two Toffoli-gates can appear in a quantum circuit. When there are many Toffoli-gates in a quantum circuit, it is necessary to find a suitable pair to apply our method. When choosing one Toffoli-gate and looking for another Toffoli-gate to mate, we have to determine the order for these 10 cases. Determining the order of these cases may be a future research task. Case 1 in Figure 5 may be the first case to consider because this case can reduce T-count in the first T-depth reduction process and can also reduce in the second T-depth reduction process.

The Toffoli-count reduction (or optimisation) technique is not covered in detail in this paper (Figure 18). As mentioned earlier, Toffoli-count reduction does not seem to necessarily lead to T-count reduction. Algorithm research to optimise Toffoli-count or Toffoli-depth seems to be necessary. If this algorithm that optimises the Toffoli-count or Toffoli-depth is made, then the correlation study with the T-depth or T-count optimisation algorithm will be an interesting research topic. There are previous studies that provided hints for implementing this algorithm [30, 37, 38].

We also do not know if our proposed circuit design is optimised in a quantum environment. Are proposed circuits' Toffoli-depth (or T-depth) optimised? Can we further reduce the length of the critical path? There may be more efficient circuit designs available for SHA-256. Finding or implementing more efficient adder circuits that can be used for SHA-256 also could be a future research task. One candidate would be to build a multi-operand adder circuit specialised for the SHA-256 quantum circuit. Of course, the performance of this adder should be better than that of the three adders composing our proposed circuit's critical path. That is, T-depth of this adder must be less than T-depth of three consecutive adders that make up the critical path in our proposed circuit.

ACKNOWLEDGEMENT

This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (⟨Q| Crypton⟩, No.2019-0-00033, Study on Quantum Security Evaluation of Cryptography based on Computational Quantum Complexity) and also partially supported by a Korea University Grant.

CONFLICT OF INTEREST

The author declares no conflict of interest.

PERMISSION TO REPRODUCE MATERIALS FROM OTHER SOURCES

None.

DATA AVAILABILITY STATEMENT

Data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Jongheon Lee https://orcid.org/0000-0002-3493-7278
Dooho Choi https://orcid.org/0000-0001-5625-4067

REFERENCES

1. Forouzan, B.A., Mukhopadhyay, D.: Cryptography and Network Security. Mc Graw Hill Education (India) Private Limited New York (2015)
2. N. I. of Standards and Technology: Fips Pub 180-4: Secure Hash Standard (Shs) (2012)
3. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing, pp. 212–219 (1996)
4. Jaques, S., et al.: Implementing Grover oracles for quantum key search on aes and lowmc. Adv. Cryptol–EUROCRYPT. 12106, 280 (2020)
5. Amy, M., et al.: Estimating the cost of generic quantum pre-image attacks on SHA-2 and SHA-3. In: International Conference on Selected Areas in Cryptography, pp. 317–337. Springer (2016)
6. Kim, P., Han, D., Jeong, K.C.: Time–space complexity of quantum search algorithms in symmetric cryptanalysis: applying to aes and SHA-2. Quant. Inf. Process. 17(12), 1–39 (2018). https://doi.org/10.1007/s11128-018-2107-3
7. Cuccaro, S.A., et al.: A New Quantum Ripple-Carry Addition Circuit (2004). arXiv preprint quant-ph/0410184
8. Vedral, V., Barenco, A., Ekert, A.: Quantum networks for elementary arithmetic operations. Phys. Rev. 54(1), 147–153 (1996). https://doi.org/10.1103/physreva.54.147
9. Beauregard, S., Brassard, G., Fernandez, J.M.: Quantum Arithmetic on Galois Fields (2003). arXiv preprint quant-ph/0301163
10. Takahashi, Y., Kunihiro, N.: A linear-size quantum circuit for addition with no ancillary qubits. Quant. Inf. Comput. 5(6), 440–448 (2005). https://doi.org/10.26421/qic5.6-2
11. Häner, T., Roetteler, M., Svore, K.M.: Factoring Using 2n+ 2 Qubits with Toffoli Based Modular Multiplication (2016). arXiv preprint arXiv:1611.07995
12. Draper, T.G., et al.: A Logarithmic-Depth Quantum Carry-Lookahead Adder (2004). arXiv preprint quant-ph/0406142
13. Lee, J., et al.: T-depth reduction method for efficient SHA-256 quantum circuit construction. In: International Conference on Information Security and Cryptology (ICISC), vol. 1, pp. 368–391 (2021)
14. Nam, Y., et al.: Automated optimization of large quantum circuits with continuous parameters. npj Quant. Inf. 4(1), 1–12 (2018). https://doi.org/10.1038/s41534-018-0072-4
15. Lidar, D.A., Brun, T.A.: Quantum Error Correction. Cambridge university press (2013)
16. Amy, M., et al.: A meet-in-the-middle algorithm for fast synthesis of depth-optimal quantum circuits. IEEE Trans. Comput. Aided Des. Integrated Circ. Syst. 32(6), 818–830 (2013). https://doi.org/10.1109/tcad.2013.2244643
17. Selinger, P.: Quantum circuits of t-depth one. Phys. Rev. 87(4), 042302 (2013). https://doi.org/10.1103/physreva.87.042302
18. Buhrman, H., et al.: New limits on fault-tolerant quantum computation. In: 2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06), pp. 411–419. IEEE (2006)
19. Fowler, A.G., Stephens, A.M., Groszkowski, P.: High-threshold universal quantum computation on the surface code. Phys. Rev. 80(5), 052312 (2009). https://doi.org/10.1103/physreva.80.052312

20. Fowler, A.G.: Time-optimal Quantum Computation (2012). arXiv preprint arXiv:1210.4626
21. Amy, M., Maslov, D., Mosca, M.: Polynomial-time t-depth optimization of clifford+ t circuits via matroid partitioning. IEEE Trans. Comput. Aided Des. Integrated Circ. Syst. 33(10), 1476–1489 (2014). https://doi.org/10.1109/tcad.2014.2341953
22. Mogensen, T.Æ.: Reversible in-place carry-look ahead addition with few ancillae. In: International Conference on Reversible Computation, pp. 224–237. Springer (2019)
23. Takahashi, Y., Kunihiro, N.: A fast quantum circuit for addition with few qubits. Quant. Inf. Comput. 8(6), 636–649 (2008). https://doi.org/10.26421/qic8.6-7-5
24. Thapliyal, H., et al.: Progress in reversible processor design: a novel methodology for reversible carry look-ahead adder. In: Transactions on Computational Science XVII, pp. 73–97. Springer (2013)
25. Thapliyal, H., Muñoz-Coreas, E., Khalus, V.: Quantum circuit designs of carry look ahead adder optimized for t-count t-depth and qubits. Sustain. Comput: Inf. Syst. 29, 100457 (2021). https://doi.org/10.1016/j.suscom.2020.100457
26. Nielsen, M.A., Chuang, I.: Quantum Computation and Quantum Information (2002)
27. Sun, W., et al.: Design and optimized implementation of the SHA-2 (256, 384, 512) hash algorithms. In: 2007 7th International Conference on ASIC, pp. 858–861. IEEE (2007)
28. Ahmad, I., Das, A.S.: Hardware implementation analysis of SHA-256 and SHA-512 algorithms on fpgas. Comput. Electr. Eng. 31(6), 345–360 (2005). https://doi.org/10.1016/j.compeleceng.2005.07.001
29. Gidney, C.: Halving the cost of quantum addition. Quantum. 2, 74 (2018). https://doi.org/10.22331/q-2018-06-18-74
30. Rahman, M.Z., Rice, J.E.: Templates for positive and negative control toffoli networks. In: International Conference on Reversible Computation, pp. 125–136. Springer (2014)
31. Abdessaied, N., et al.: Technology mapping of reversible circuits to clifford+ t quantum circuits. In: 2016 IEEE 46th International Symposium on Multiple-Valued Logic (ISMVL), pp. 150–155. IEEE (2016)
32. Barenco, A., et al.: Elementary gates for quantum computation. Phys. Rev. 52(5), 3457–3467 (1995). https://doi.org/10.1103/physreva.52.3457
33. Niemann, P., Gupta, A., Drechsler, R.: T-depth optimization for fault-tolerant quantum circuits. In: 2019 IEEE 49th International Symposium on Multiple-Valued Logic (ISMVL), pp. 108–113. IEEE (2019)
34. Abdessaied, N., Soeken, M., Drechsler, R.: Quantum circuit optimization by hadamard gate reduction. In: International Conference on Reversible Computation, pp. 149–162. Springer (2014)
35. Garcia-Escartin, J.C., Chamorro-Posada, P.: Equivalent Quantum Circuits (2011). arXiv preprint arXiv:1110.2998
36. Paler, A., Oumarou, O., Basmadjian, R.: On the realistic worst case analysis of quantum arithmetic circuits. IEEE Trans. Quant. Eng. 3, 1–11 (2022). https://doi.org/10.1109/tqe.2022.3163624
37. Maslov, D., Dueck, G.W., Miller, D.M.: Simplification of Toffoli networks via templates. In: 16th Symposium on Integrated Circuits and Systems Design, 2003. SBCCI 2003. Proceedings, pp. 53–58. IEEE (2003)
38. Miller, D.M., Maslov, D., Dueck, G.W.: A transformation based algorithm for reversible logic synthesis. In: Proceedings 2003. Design Automation Conference (Ieee Cat. No. 03ch37451), pp. 318–323. IEEE (2003)
39. Gidney, C.: Algorithmic assertions: constructing large increment gates. URL: https://algassert.com/circuits/2015/06/12/Constructing-Large-Increment-Gates.html. (2015)

How to cite this article: Lee, J., et al.: T-depth reduction method for efficient SHA-256 quantum circuit construction. IET Inf. Secur. 1–20 (2022). https://doi.org/10.1049/ise2.12074