Quantum Circuits for High-degree and half-Multiplication For Post-Quantum Analysis

Rini Wisnu Wardhani[1][0000−0003−0565−6458] ⋆, Dedy Septono Catur Putranto[2,3][0000−0002−1246−7877], and Howon Kim[1][0000−0001−8475−7294]

[1] School of Computer Science and Engineering, Pusan National University, Busan 609735, South Korea
[2] IoT Research Center, Pusan National University, Busan 609735, South Korea
[3] Blockchain Platform Research Center, Pusan National University, Busan 609735, South Korea
{rini.wisnu,dedy.septono,howonkim}@pusan.ac.kr

Abstract. Along with the possibility of accelerated polynomial multiplication, the Toom-Cook $k-$way multiplication technique has drawn significant interest in the field of post-quantum cryptography due to its ability to serve as a part of the lattice-based algorithm. In contrast, the growing likelihood of attacks based on multiplication, specifically correlation power analysis attacks, has heightened vulnerability and emphasized the need to examine the feasibility of employing the polynomial multiplication method as a potential alternative in the era of post-quantum. This study examines thoroughly an elaborate mathematical procedure designated as high-degree and half-multiplication, focusing on the design of an efficient multiplication technique. The proposed polynomial multiplication is intended to be enhanced in terms of asymptotic performance analysis and quantum resource utilization. Through the utilization of the Toom-Cook 8.5-way method, we reach the lowest asymptotic performance and quantum resources usage for multiplication operation in comparison to the existing Toom-Cook-based multiplication designs with $186n^{\log_9 17} - 202n$ Toffoli count and $n(\frac{17}{9})^{1-\frac{\log 17}{(2\log 17-\log 9)}\log_9 n} \approx n^{1.053}$ Toffoli depth. The designed multiplication yields a qubit count of $n(\frac{17}{9})^{\frac{\log 17}{(2\log 17-\log 9)}\log_9 n}$, or approximately $n^{1.236}$. We further compare its asymptotic performance and quantum resource efficiency to other Toom-Cook-based multiplications to determine its efficacy.

Keywords: High-degree and half-multiplication · Toom-Cook · Post-Quantum Cryptography · Correlation Power Analysis · Quantum

1 Introduction

The Toom-Cook, a method based on [34], [11], is widely acknowledged as an effective approach for solving large number multiplication algorithms. The approach being referred to is a mathematical method employed for the efficient multiplication of polynomials. This method involves breaking down the multiplication process into smaller multiplications (sub-multiplications) and

⋆ This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the Convergence Security Core Talent Training Business (Pusan National University), support program (IITP-2023-2022-0-01201) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation, 50%) and by the Institute for Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2019-0-00033, Study on Quantum Security Evaluation of Cryptography based on Computational Quantum Complexity, 50%)

additions, thereby minimizing the overall computing complexity. The use of this technique is prevalent throughout diverse domains, including computer algebra systems, cryptography, and signal processing, with the aim of enhancing the efficiency of polynomial multiplication processes.

Besides the number theoretic transform (NTT)-based polynomial multiplication, the Toom-Cook-based or Karatsuba-based polynomial multiplication algorithms have experienced a resurgence in popularity after the commencement of the National Institute of Standards and Technology's (NIST) post-quantum standardization program [26], [23]. Several studies (i.e., [14], [23], and [26]) have put forth a new approach to Toom-Cook multiplication, taking into account the NIST adoption of the module learning with errors (MLWE) algorithm, which forms the basis of many lattice-based cryptography schemes, as the forthcoming standard.

In terms of Toom-Cook multiplication implementation, to optimize performance and reduce implementation costs, Putranto et al. [32] propose employing a Toom-Cook-based multiplier based on several Toom-Cook calculation strategies, including [7], [35], [13], [21]. The analysis of the asymptotic performance of multiplication algorithms and the corresponding costs associated with their quantum implementation offers effectiveness in multiplication operations and valuable perspectives on the importance of multiplication algorithms within the realm of post-quantum cryptography (PQC) and mitigating the risk of side-channel attacks (SCA). Meanwhile, Mera et al. [26], provide a proposition consisting of two innovative strategies aimed at enhancing the efficiency of polynomial multiplications based on the Toom-Cook algorithm. These techniques are then implemented within the Saber post-quantum key encapsulation mechanism.

Recently, the present study [23] investigates the vulnerabilities of the Toom-Cook algorithm in the reference implementation of the Saber cryptographic scheme. It introduces a novel approach by conducting a single-trace attack on Toom-Cook, utilizing the soft-analytical side-channel attack technique. In accordance with this, Mujdei et al. [28] undertook a comparative examination of the complexity associated with attacking various multiplication schemes, multiplication algorithms, and parameter selections. This study utilized the correlation power analysis (CPA) technique, which was first introduced by Brier et al. in their influential paper released in 2004 [10], to prove the existing Toom-Cook vulnerability, particularly the Toom-Cook 4-way PQC algorithm, against the attacks.

The examination of the feasibility of polynomial multiplication as a prospective alternative within the context of PQC holds significant importance. Lattice-based cryptographic systems commonly employ either the NTT with time complexity of $(\mathcal{O}(n \log n))$ [30] or the Toom-Cook/Karatsuba algorithm with time complexity of $(\mathcal{O}(n^{1+\epsilon})$, where $0 < \epsilon < 1$), [34], [11], [17], to achieve efficient polynomial multiplication involving n coefficients [28]. In this paper, we will explore the utilization of a new and advantageous multiplication operation derived from Toom's approach, considering that Toom-Cook-based multiplication, especially degrees up to 4, is part of the lattice-based post-quantum algorithm approach, which is also associated with attacks. Further, the proposed multiplication is intended to be integrated into a quantum cryptanalysis circuit with the aim of facilitating an evaluation of post-quantum security.

In this study, we refer to Bodrato's research on high-degree Toom'n'half balanced and unbalanced multiplication [8] to elucidate the functioning of Toom's method for polynomials. To the best of our knowledge, this study is the first to utilize high-degree and half-multiplication compounds in quantum circuits, specifically Toom-Cook-based multiplication exceeding 8 degrees. The primary objective in the design of high-degree and half-multiplication quantum circuits is to reach lower asymptotic performance analyses and minimize the utilization of quantum resources during the execution of multiplication operations. The contributions of this paper can be succinctly summarized as follows:

1. We elaborate a comprehensive analysis of multiplication strategies (i.e., [35], [22], and [32]), with a specific emphasis on the high-degree and half-multiplication technique, the Toom-Cook 8.5-way method. Referring to [8], we conduct computation steps like splitting, evaluation, recursive multiplication, interpolation, and recomposition in a certain order, to reach the goal of yielding the best asymptotic performance analysis and the lowest amount of quantum resource use.
2. We design the Toom-Cook 8.5-way multiplier in a quantum environment, yielding the lowest asymptotic performance analysis for the multiplier and the minimum quantum resource utilization with qubit count $n(\frac{17}{9})^{\frac{\log 17}{(2\log 17 - \log 9)}\log_9 n} \approx n^{1.236}$, $186n^{\log_9 17} - 202n$ Toffoli count, and $n(\frac{17}{9})^{1-\frac{\log 17}{(2\log 17 - \log 9)}\log_9 n} \approx n^{1.053}$ Toffoli depth.
3. We then investigate the asymptotic performance and quantum resource use of various multiplication algorithms, namely the naïve schoolbook method, the Karatsuba algorithm, and existing Toom-Cook-based multiplication up to 8.5 degrees. Additionally, we provide a thorough analysis and evaluation of various factors, including qubit count, Toffoli count, and Toffoli depth, for the purpose of assessing the space-time complexity and drawing up a comprehensive comparison metric to the multiplication operation.

The organization of the paper is as follows: Section 1 provides an overview of the background insights relevant to our work. Section 2 provides a brief overview of high-degree and half-multiplication, particularly in the context of Toom-Cook-based multiplication. Section 3 outlines a detailed procedure for designing the proposed high-degree and half-multiplication, the Toom-Cook 8.5-way. In Section 4, we provide a concise insight into the utilization and underlying principles of multiplication-based attacks with CPA and address multiplication usage in cryptanalysis circuits that led to a post-quantum security evaluation. In Section 5, we analyze and compare the computational complexity in terms of space and time for designs involving proposed multiplication. Future work discussion and conclusions are formulated in Section 6 and Section 7.

2 High-degree and half-Multiplication

The Schoolbook Multiplication algorithm, which has a time complexity of $\mathcal{O}(n^2)$, is considered the most basic and straightforward approach for multiplying polynomials of degree n, which is equivalent to a variant of the Toom-Cook 1-way algorithm. Meanwhile, the Karatsuba algorithm can be considered a variant of the Toom-Cook 2-way algorithm, in which the original number is divided into two smaller sub-numbers. The reduction of four multiplications to three results in the Karatsuba method yield efficiency compared to naive with a complexity value of $(n^{log(3)/log(2)}) \equiv \mathcal{O}(n^{1.58})$.

The Toom-Cook algorithm, specifically the Toom-Cook k-way algorithm for multiplication, is a divide-and-conquer approach that bears resemblance to Karatsuba multiplication. However, unlike Karatsuba multiplication which divides each polynomial into two equal parts during each recursive step, the Toom-Cook k-way multiplication divides two large integers f and g into k smaller parts, each with a length of l. In general, the time complexity of the Toom-Cook $k-$way algorithm can be expressed as $\mathcal{O}(c(k)n^e)$, where e is calculated as the logarithm of $(2k-1)$ divided by the logarithm of k. The term n^e represents the time spent on sub-multiplications, while c denotes the time spent on additions and multiplication by small constants.

The computational procedures encompass many steps such as splitting, evaluation, recursive multiplication, interpolation, and recomposition, which have already received extensive study in other works ([8, 13, 21, 32, 35]). This study concentrates its attention on effective multiplication, specifically exploring its complexity before delving into the realm of quantum circuits for high-degree and half-multiplication in quantum architecture.

In the first step in Toom's splitting step, in order to divide a given quantity into k segments using Toom's $k-$way algorithm, it is necessary to choose a base $B = b^i$ that satisfies the condition where the number of integer digits both m and n when expressed in base B does not exceed k. A commonly selected option for the variable i is provided by Equation 1, then, the variables m and n are partitioned into their respective base B digits, denoted as m_i and n_i.

$$i = max\left\{\left\lfloor\frac{\lceil\log_b m\rceil}{k_m}\right\rfloor, \left\lfloor\frac{\lceil\log_b n\rceil}{k_n}\right\rfloor\right\}+1 \tag{1}$$

Subsequently, the aforementioned digits are employed as coefficients in polynomials p and q of degree $(k-1)$, satisfying the condition that $p(B)$ equals m and $q(B)$ equals n. The rationale for the defining of these polynomials lies in the fact that by calculating their product, denoted as $r(x) = p(x)q(x)$, the resulting value $r(B)$ will correspond to the multiplication of m x n.

In the case where the multiplicands have different magnitudes, it is advantageous to employ different values of k for m and n, denoted as k_m and k_n. An instance in this condition is the high-degree and half-multiplication Toom-Cook $k-$way ; for example (using terminology, high-degree and half-multiplication), Toom-Cook 8.5-way corresponds to the Toom-Cook algorithm with the specific values of $k_m = 9$ and $k_n = 8$. In this particular scenario, the selection of the variable i in the equation $B = b^i$ is commonly determined by Equation 1.

3 Quantum Toom-Cook 8.5-way Multiplier Design

Zanoni et al. [35] introduce a conventional computational implementation of a balanced Toom-Cook 8-way algorithm for the purpose of integer multiplication and squaring. The authors successfully achieved a degree of 7 in their Toom-Cook-based multiplication version. In their comprehensive study, Dutta et al. [13] provide an in-depth elucidation of the Toom-Cook 2.5-way technique employed in the realm of quantum computing. The authors primarily concentrate on the identification of the maximum count of Toffoli gates and qubits attainable by means of a rigorous examination of the recursive tree inherent to the algorithm.

The research undertaken by Larasati et al. [21] shows findings that demonstrate the possibility of the $k-$way Toom-Cook method, which employs higher-order polynomial interpolation, to exhibit lower asymptotic complexity in comparison to alternative approaches such as Toom-Cook 2.5-way. In their study, Larasati et al. [21] expound upon the Toom-Cook 3-way algorithm by incorporating the division gate. They augment their analysis by drawing upon the research conducted by Bodrato et al. [7], resulting in a singular instance of accurate division by three circuits in every iteration. Moreover, the cost related to the remaining division was reduced by the usage of the circuit's unique properties. The aforementioned accomplishment was attained through the use of a circuit that employs a constant multiplication by reciprocal technique, complemented with the requisite swap operations [21].

Referring to [32], the following part provides a detailed description of the sequential procedure for implementing our quantum Toom-Cook 8.5-way Multiplication algorithm, while also highlighting the distinctions between this approach and the Toom-Cook 8-way multiplication method for the purpose of clarification. The comparison between the recursion tree structures of Toom-Cook 8-way and Toom-Cook 8.5-way is depicted in Figure 1. In the present context, Figure 2 draws a

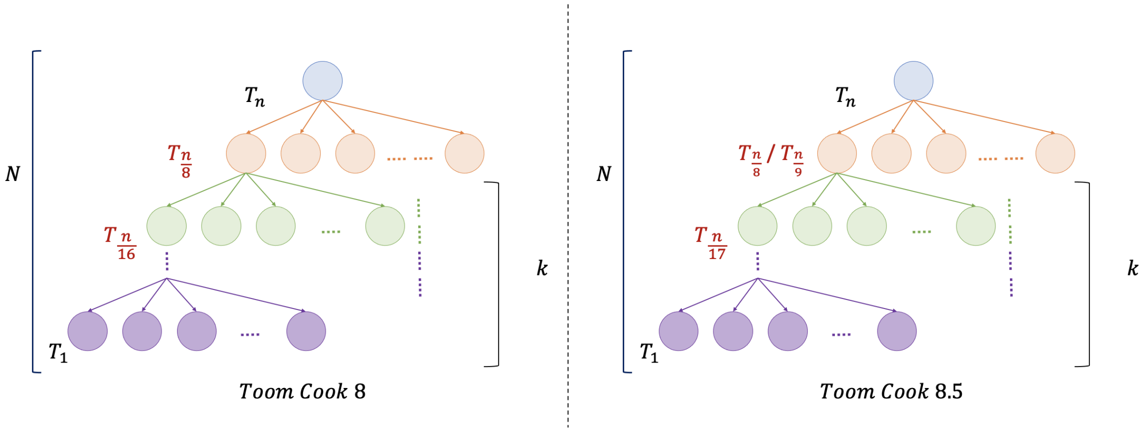


Fig. 1: The Toom-Cook 8-way and 8.5-way Multiplication Recursion Tree Structure, where T represents the Toom-Cook $k-$way Multiplication and n and N represent the bit length for each level and the overall depth of the tree, respectively.

comparative analysis of quantum circuits pertaining to the multiplications of Toom-Cook 8-way and Toom-Cook 8.5-way.

3.1 Computation Steps

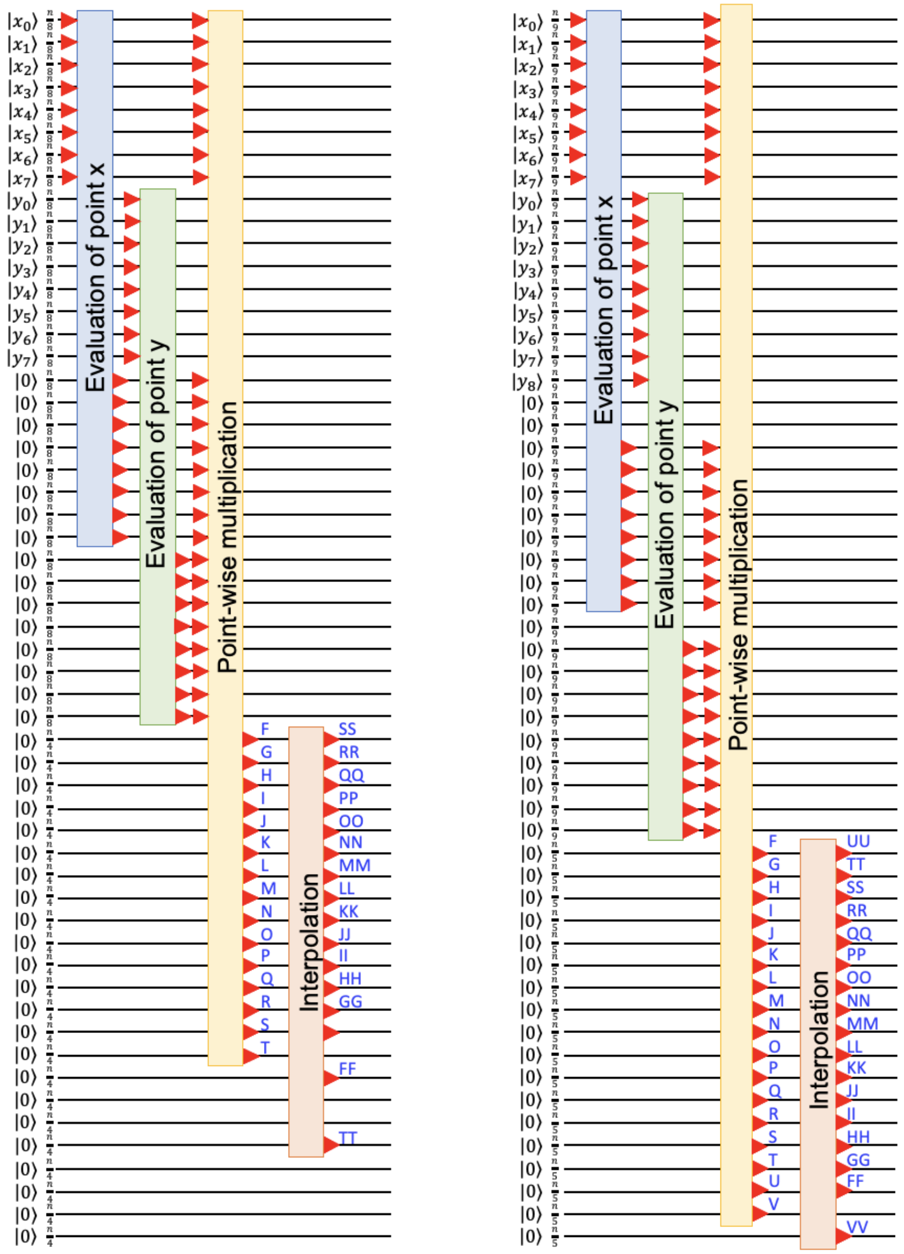
Focusing on the Toom-Cook 8.5-way strategy design, this work explains and undertakes a thorough investigation of high-degree and half-multiplication methods based on the Toom-Cook algorithm within the context of polynomial multiplication. We incorporate several prior research findings, including [32], and [8]. The processes of computation include splitting, evaluation, recursive multiplication, interpolation, and recomposition, as discussed in previous studies [35], [8], [22], [32]. To offer a succinct explanation of the approach, the quantities to be multiplied, referred to as the input operands, are represented by the variables x and y. The variable x is used to represent the complete numerical input. The subscripts $x_0, x_1, x_{-1}, x_{-2}, \ldots$ are used to signify the individual components of the input. On the other hand, the notations $x(0), x(1), x(-1), x(-2), \ldots$ are employed to indicate the results obtained by evaluating the variable x at certain places.

Splitting. As shown by Equations 2 and 3, the specified inputs, denoted as x and y, are divided into eight smaller pieces of length $\frac{n}{8}$. The radix j in the equations can be determined in advance through the calculation of Equation 4.

$$x = x_7 s^{7j} + x_6 s^{6j} + x_5 s^{5j} + x_4 s^{4j} + x_3 s^{3j} + x_2 s^{2j} + x_1 s^{j} + x_0 \tag{2}$$

$$y = y_8 s^{8j} + y_7 s^{7j} + y_6 s^{6j} + y_5 s^{5j} + y_4 s^{4j} + y_3 s^{3j} + y_2 s^{2j} + y_1 s^{j} + y_0 \tag{3}$$

$$j = max\left\{\left\lfloor\frac{\lceil\log_2 x\rceil}{9}\right\rfloor, \left\lfloor\frac{\lceil\log_2 y\rceil}{8}\right\rfloor\right\}+1 \tag{4}$$


$|x_0\rangle$ … $|x_7\rangle$
$|y_0\rangle$ … $|y_7\rangle$
$|0\rangle$
$\frac{n}{8}$
$\frac{n}{4}$
Evaluation of point x
Evaluation of point y
Point-wise multiplication
Interpolation
F G H I J K L M N O P Q R S T
SS RR QQ PP OO NN MM LL KK JJ II HH GG FF TT
$|x_0\rangle$ … $|x_7\rangle$
$|y_0\rangle$ … $|y_8\rangle$
$|0\rangle$
$\frac{n}{9}$
$\frac{n}{5}$
Evaluation of point x
Evaluation of point y
Point-wise multiplication
Interpolation
F G H I J K L M N O P Q R S T U V
UU TT SS RR QQ PP OO NN MM LL KK JJ II HH GG FF VV

Toom Cook 8   *Toom Cook* 8.5

Fig. 2: Quantum Circuits Comparison for the Toom-Cook 8-way and Toom-Cook 8.5-way Multiplication Algorithms. The function block boxes serve as representations of the individual steps involved in constructing the Toom-Cook quantum circuit. The quantum circuit utilized in the multiplication algorithm uses red triangles to denote the input and output of each respective operation within the function blocks. A notation symbol is employed to denote the quantum state of the input, with each line representing a required register in the quantum circuit. The presence of triangles positioned on the left side of a block serves to highlight the location of its input entry point. The output location on the right side is symbolized by triangles. To maintain simplicity, the ancilla registers are omitted from the display.

$$
\begin{array}{l}
F = x_0 y_0 \\
G = (x_7 + x_6 + x_5 + x_4 + x_3 + x_2 + x_1 + x_0)(y_8 + y_7 + y_6 + y_5 + y_4 + y_3 + y_2 + y_1 + y_0) \\
H = (-x_7 + x_6 - x_5 + x_4 - x_3 + x_2 - x_1 + x_0)(y_8 + -y_7 + y_6 - y_5 + y_4 - y_3 + y_2 - y_1 + y_0) \\
I = (128x_7 + 64x_6 + 32x_5 + 16x_4 + 8x_3 + 4x_2 + 2x_1 + x_0)(256y_8 + 128y_7 + 64y_6 + 32y_5 + 16y_4 + 8y_3 + 4y_2 + 2y_1 + y_0) \\
J = (-128x_7 + 64x_6 - 32x_5 + 16x_4 - 8x_3 + 4x_2 - 2x_1 + x_0)(256y_8 + -128y_7 + 64y_6 - 32y_5 + 16y_4 - 8y_3 + 4y_2 - 2y_1 + y_0) \\
K = (16384x_7 + 4096x_6 + 1024x_5 + 256x_4 + 64x_3 + 16x_2 + 4x_1 + x_0) \\
(65536y_8 + 16384y_7 + 4096y_6 + 1024y_5 + 256y_4 + 64y_3 + 16y_2 + 4y_1 + x_0) \\
L = (-16384x_7 + 4096x_6 - 1024x_5 + 256x_4 - 64x_3 + 16x_2 - 4x_1 + x_0) \\
(65536y_8 - 16384y_7 + 4096y_6 - 1024y_5 + 256y_4 - 64y_3 + 16y_2 - 4y_1 + x_0) \\
M = (2097152x_7 + 262144x_6 + 32768x_5 + 4096x_4 + 512x_3 + 64x_2 + 8x_1 + x_0) \\
(16777216y_8 + 2097152y_7 + 262144y_6 + 32768y_5 + 4096y_4 + 512y_3 + 64y_2 + 8y_1 + y_0) \\
N = (-2097152x_7 + 262144x_6 - 32768x_5 + 4096x_4 - 512x_3 + 64x_2 - 8x_1 + x_0) \\
(16777216y_8 + -2097152y_7 + 262144y_6 - 32768y_5 + 4096y_4 - 512y_3 + 64y_2 - 8y_1 + y_0) \\
O = (268435456x_7 + 16777216x_6 + 1048576x_5 + 65536x_4 + 4096x_3 + 256x_2 + 16x_1 + x_0) \\
(4294967296y_8 + 268435456y_7 + 16777216y_6 + 1048576y_5 + 65536y_4 + 4096y_3 + 256y_2 + 16y_1 + y_0) \\
P = (-268435456x_7 + 16777216x_6 - 1048576x_5 + 65536x_4 - 4096x_3 + 256x_2 - 16x_1 + x_0) \\
(4294967296y_8 - 268435456y_7 + 16777216y_6 - 1048576y_5 + 65536y_4 - 4096y_3 + 256y_2 - 16y_1 + y_0) \\
Q = (0.0078125x_7 + 0.015625x_6 + 0.03125x_5 + 0.0625x_4 + 0.125x_3 + 0.25x_2 + 0.5x_1 + x_0) \\
(0.00390625y_8 + 0.0078125y_7 + 0.015625y_6 + 0.03125y_5 + 0.0625y_4 + 0.125y_3 + 0.25y_2 + 0.5y_1 + y_0) \\
R = (-0.0078125x_7 + 0.015625x_6 - 0.03125x_5 + 0.0625x_4 - 0.125x_3 + 0.25x_2 - 0.5x_1 + x_0) \\
(0.00390625y_8 - 0.0078125y_7 + 0.015625y_6 - 0.03125y_5 + 0.0625y_4 - 0.125y_3 + 0.25y_2 - 0.5y_1 + y_0) \\
S = (0.00006103515625x_7 + 0.000244140625x_6 + 0.0009765625x_5 + 0.00390625x_4 + 0.015625x_3 + 0.0625x_2 + 0.25x_1 + x_0) \\
(0.0000152587890625y_8 + 0.00006103515625y_7 + 0.000244140625y_6 + 0.0009765625y_5 + 0.00390625y_4 + 0.015625y_3 + 0.0625y_2 \\
\ + 0.25y_1 + y_0) \\
T = (-0.00006103515625x_7 + 0.000244140625x_6 - 0.0009765625x_5 + 0.00390625x_4 - 0.015625x_3 + 0.0625x_2 - 0.25x_1 + x_0) \\
(0.0000152587890625y_8 - 0.00006103515625y_7 + 0.000244140625y_6 - 0.0009765625y_5 + 0.00390625y_4 - 0.015625y_3 + 0.0625y_2 \\
\ - 0.25y_1 + y_0) \\
U = (0.000000476837158203125x_7 + 0.000003814697265625x_6 + 0.000030517578125x_5 + 0.000244140625x_4 + 0.001953125x_3 \\
\ + 0.015625x_2 + 0.125x_1 + x_0)(0.0000000596046447753906y_8 + 0.000000476837158203125y_7 + 0.000003814697265625y_6 \\
\ + 0.000030517578125y_5 + 0.000244140625y_4 + 0.001953125y_3 + 0.015625y_2 + 0.125y_1 + y_0) \\
V = (-0.000000476837158203125x_7 + 0.000003814697265625x_6 - 0.000030517578125x_5 + 0.000244140625x_4 - 0.001953125x_3 \\
\ + 0.015625x_2 - 0.125x_1 + x_0)(0.0000000596046447753906y_8 - 0.000000476837158203125y_7 + 0.000003814697265625y6 \\
\ - 0.000030517578125y_5 + 0.000244140625y_4 - 0.001953125y_3 + 0.015625y_2 - 0.125y_1 + y_0)
\end{array}
\tag{5}
$$

Evaluation. We employ $x_1 = 0$, $x_2 = 1$, $x_3 = -1$, $x_4 = 2$, $x_5 = -2$, $x_6 = 4$, $x_7 = -4$, $x_8 = 8$, $x_9 = -8$, $x_{10} = 16$, $x_{11} = -16$, $x_{12} = 0.5$, $x_{13} = -0.5$, $x_{14} = 0.25$, $x_{15} = -0.25$, $x_{16} = -0.125$, and

$x_{17} = -0.125$ to obtain $x(0)$, $x(1)$, $x(-1)$, $x(2)$, $x(-2)$, $x(4)$, $x(-4)$, $x(8)$, $x(-8)$, $x(16)$, $x(-16)$, $x(0.5)$, $x(-0.5)$, $x(0.25)$, $x(-0.25)$, $x(0.125)$ and $x(-0.125)$ for the evaluating points x and y, each of the 17 predefined evaluation points. Figure 3 and Figure 4 illustrate the evaluation points x and y for the evaluation stage in the Toom-Cook 8.5-way multiplications design. The exact equation for the evaluation points $x(0)$, $x(1)$, $x(-1)$, $x(2)$, $x(-2)$, $x(4)$, $x(-4)$, $x(8)$, $x(-8)$, $x(16)$, $x(-16)$, $x(0.5)$, $x(-0.5)$, $x(0.25)$, $x(-0.25)$, $x(0.125)$ and $x(-0.125)$ is not included in this work. However, it can be inferred from the evaluation multiplication equation, Equation 5.

Recursive Multiplication. A single iteration of non-recursive point-wise multiplication for Toom-Cook 8.5-way multiplication utilizes a total of 17 multiplications, each with smaller bit lengths. To multiply each component of $x(0)$, $x(1)$, $x(-1)$, $x(2)$, $x(-2)$, $x(4)$, $x(-4)$, $x(8)$, $x(-8)$, $x(16)$, $x(-16)$, $x(0.5)$, $x(-0.5)$, $x(0.25)$, $x(-0.25)$, $x(0.125)$ and $x(-0.125)$, the result is expressed in Equation 5, denoted as F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, and V, respectively.

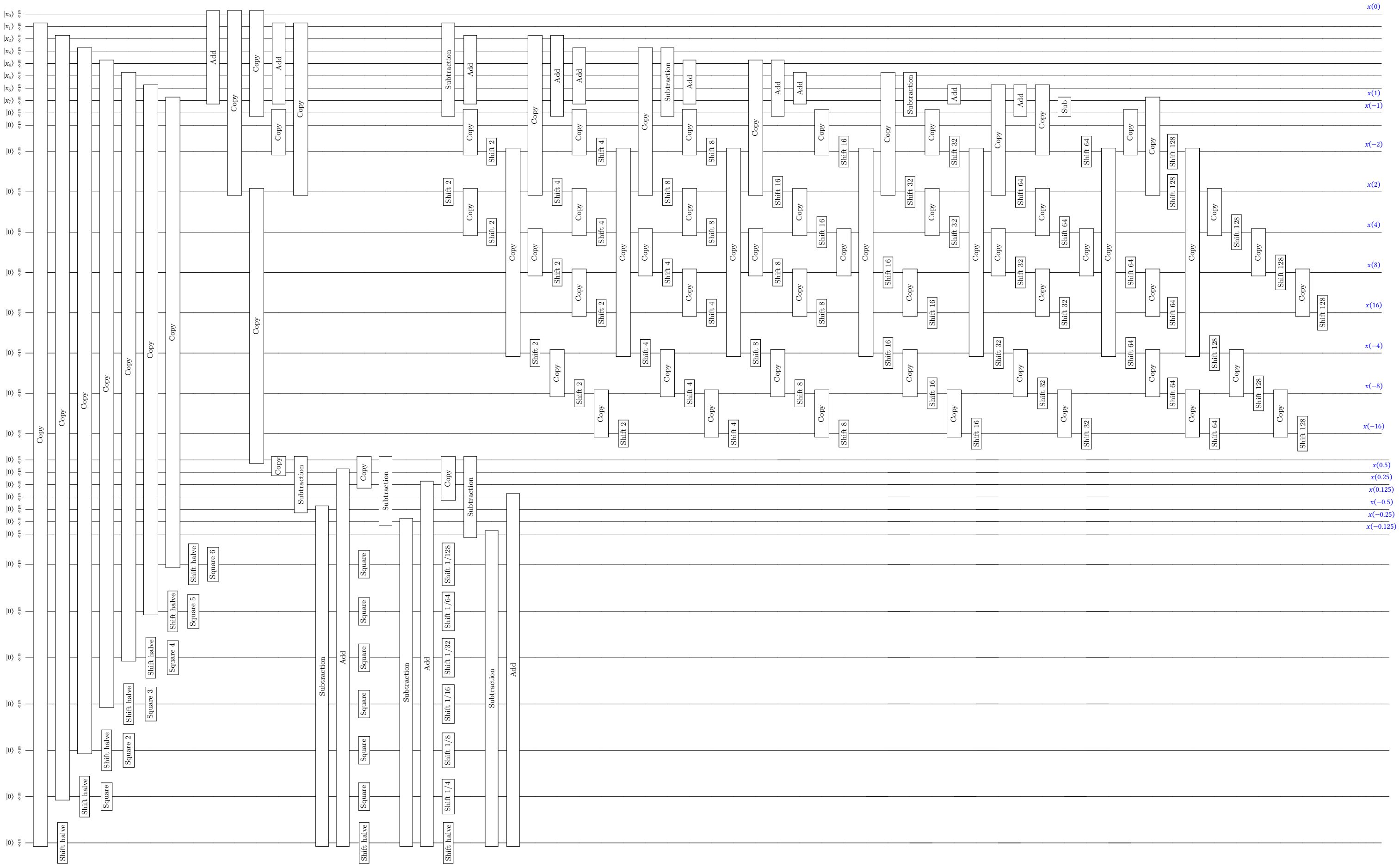
Interpolation. The process of interpolation can be represented mathematically using a matrix, which is the opposite process of multiplying a point, as demonstrated in Equation 6. It needs to be noticed that, in the aforementioned procedure, an inverse matrix derived from the sub-multiplication of coefficients $(k_0 \dots k_{16})$ in Equation 5 is employed. To facilitate comprehension, the inverse matrix is represented as described in Equation 6.

Recomposition The recomposition from the interpolation result is indicated as VV, UU, TT, SS, RR, QQ, PP, OO, NN, MM, LL, KK, JJ, II, HH, GG, and FF in Equation 7 below. The final product of Toom-Cook 8.5-way multiplication is the xy Equation.

$$\begin{pmatrix} VV \\ TT \\ SS \\ RR \\ QQ \\ PP \\ OO \\ NN \\ MM \\ LL \\ KK \\ JJ \\ II \\ HH \\ GG \\ FF \end{pmatrix} = \begin{pmatrix} 1 & 0 & \cdots & 0 & 0 \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \end{pmatrix}^{-1} \begin{pmatrix} F \\ G \\ H \\ I \\ J \\ K \\ L \\ M \\ N \\ O \\ P \\ Q \\ R \\ S \\ T \\ U \end{pmatrix} \tag{6}$$

$$\begin{aligned} xy = {} & FF2^{16j} + GG2^{15j} + HH2^{14j} + II2^{13j} + JJ2^{12j} + KK2^{11j} + LL2^{10j} + MM2^{9j} \\ & + NN2^{8j} + OO2^{7j} + PP2^{6j} + QQ2^{5j} + RR2^{4j} + SS2^{3j} + TT2^{2j} + UU2^{j} + VV \end{aligned} \tag{7}$$

Fig. 3: Evaluation point x



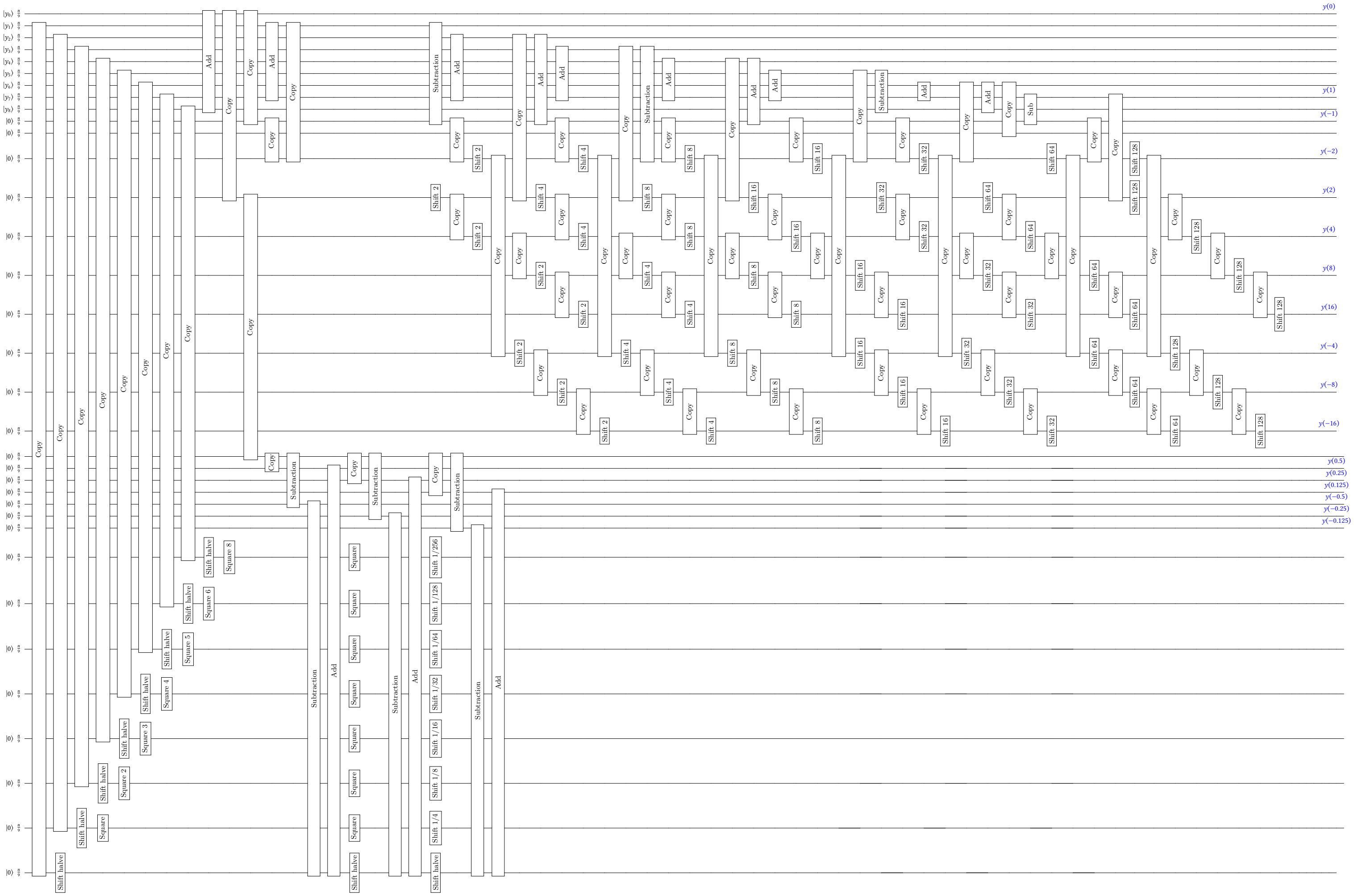


Fig. 4: Evaluation point y

4 Toom-Cook-Based Polynomial Multiplication in the Post-Quantum

Numerous investigations have been conducted pertaining to the enhancement of public-key cryptosystems, aiming to protect against potential attacks deriving from both classical and quantum computing paradigms. The period characterized by the need for quantum-resistant encryption is commonly denoted as the PQC era, as elucidated in [1]. According to the NIST PQC standardization process, the two main algorithms that are suggested for a range of applications, including digital signatures, are Crystals-Kyber [9] for public-key setup and Crystals-Dilithium [25] Lattice-based encryption is expected to exhibit optimal efficiency and resilience against quantum attacks, rendering it a feasible solution within the domain of PQC and appears to be the most rapid implementation as in [27] [24] [6] [5]. Dilithium, Falcon, FrodoKEM, Kyber, NTRU, NTRU Prime, and Saber are seven of the fifteen candidates in the NIST third round that use lattice-based cryptography [1]. In this subsection, we present a brief example of the usage and implementation of Toom-Cook-based multiplication in the Saber and Kyber PQC algorithm, as well as the potential vulnerability that arises from the utilization of lower-degree multiplication.

The primary focus of public key cryptography (PKC) implementation is on compactness, power efficiency, and energy consumption, with a secondary consideration given to throughput or delay [14]. This is due to its main purpose of generating shared secret keys. While the majority of other research concentrates on optimizing NTT-based multiplications, [14] research optimizes a Toom-Cook-based multiplier to an exceptional degree. A memory-efficient striding Toom-Cook with delayed interpolation yields a highly compact, low-power implementation that allows for a very regular memory access scheme. They demonstrate the multiplier's effectiveness and integrate it into one of the four NIST finalists, the Saber post-quantum accelerator. The results of the runtime analysis for a post-quantum lattice-based cryptographic algorithm, specifically a key encapsulation mechanism, are displayed in Figure 5. In this figure, our focus is solely on the Kyber algorithm. The analysis is conducted by comparing the algorithm's runtime behavior and memory consumption statistics, as documented in the work by Mujdei et al. [28].

Polynomial multiplications, such as Toom-Cook and NTT, play a crucial role in lattice-based post-quantum encryption by serving as the essential constituents. Lattice-based cryptographic systems commonly employ either the NTT with time complexity of $(\mathcal{O}(n \log n))$ [30] or the Toom-Cook/Karatsuba algorithm with time complexity of $(\mathcal{O}(n^{1+\epsilon})$, where $0 < \epsilon < 1$), [34] [11] [17], to achieve efficient polynomial multiplication involving n coefficients [28]. These multiplications facilitate the division of the resultant sub-polynomial, as highlighted in [28]. The Saber algorithm employs an additional division of the resultant sub-polynomials into two Karatsuba layers, followed by the execution of a 16-coefficient schoolbook operation [28]. Figure 6 displays an image that portrays an occurrence of Toom-Cook-based multiplication executed within the Saber structures. We redraw from the work of Mera et al. [26] to demonstrate the application of the Toom-Cook 4-way method in the implementation of the Saber post-quantum cryptography algorithm.

The exploitation of side-channel information, such as power consumption, electromagnetic radiation, and execution time, has been shown to be a method for gaining unauthorized access to sensitive data [19]. CPA is widely recognized as a very effective technique that leverages the correlation between a device's power consumption and the data it is processing. This approach exploits power fluctuations that are caused by mathematical processes such as multiplication. Hence, the evaluation of potential risks associated with multiplication exploitation in side-channel analysis attacks, particularly when utilizing the CPA approach, is crucial during the construction of cryp-

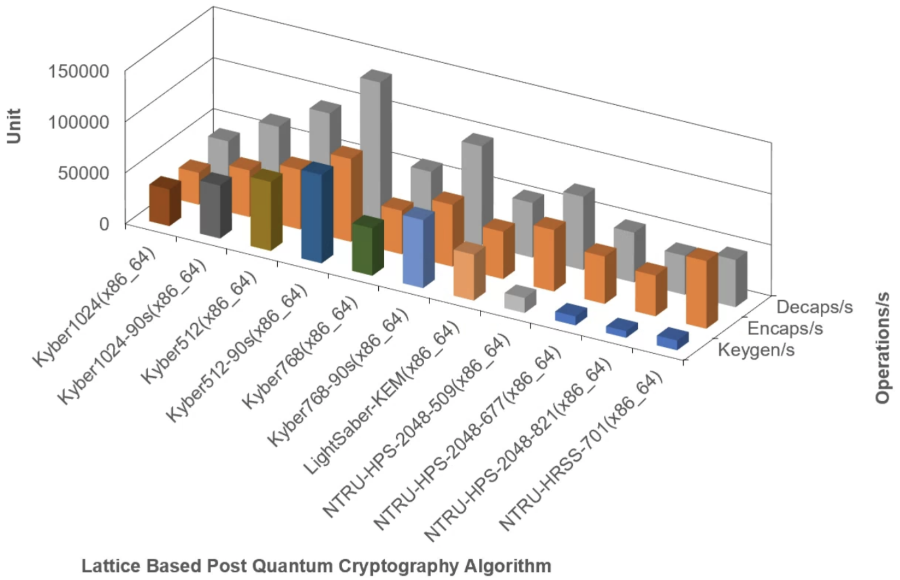


Fig. 5: Runtime analysis of Open Quantum Safe Lattice-based Cryptographic algorithms (Key Encapsulation Mechanisms)

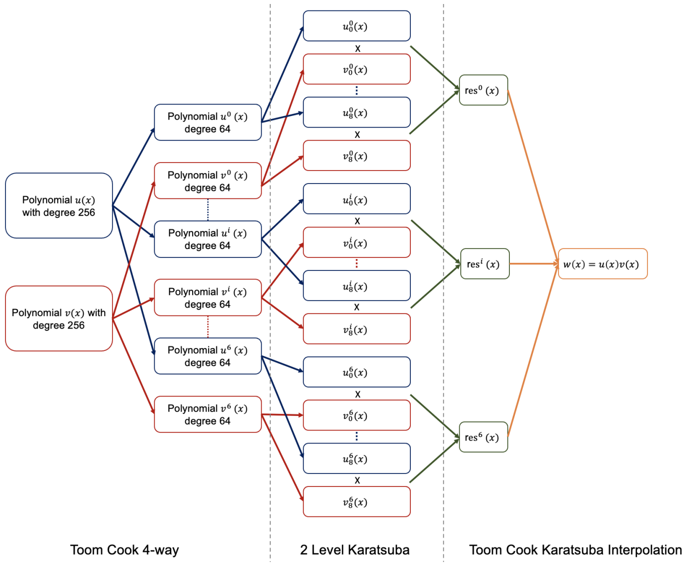

Polynomial $u(x)$ with degree 256
Polynomial $v(x)$ with degree 256
Polynomial $u^0\ (x)$ degree 64
Polynomial $v^0\ (x)$ degree 64
Polynomial $u^i\ (x)$ degree 64
Polynomial $v^i\ (x)$ degree 64
Polynomial $u^6\ (x)$ degree 64
Polynomial $v^6\ (x)$ degree 64
$u_0^0(x)$
X
$v_0^0(x)$
$u_8^0(x)$
X
$v_8^0(x)$
$u_0^i(x)$
X
$v_0^i(x)$
$u_8^i(x)$
X
$v_8^i(x)$
$u_0^6(x)$
X
$v_0^6(x)$
$u_8^6(x)$
X
$v_8^6(x)$
res$^0\ (x)$
res$^i\ (x)$
res$^6\ (x)$
$w(x) = u(x)v(x)$
Toom Cook 4-way
2 Level Karatsuba
Toom Cook Karatsuba Interpolation

Fig. 6: The Toom-Cook 4-way and Karatsuba Multiplication used in Saber Post-Quantum Cryptography Algorithm

tographic algorithms. This concern arises due to the frequent use of arithmetic multiplication as a sub-operation multiplier in real implementations.

The architectural design of all NTRU versions exhibits a common structure, characterized by the presence of four Karatsuba layers, with the exception of *ntruhps*2048509, which features three layers [28]. Further, variations in the schoolbook thresholds are observed [28]. Mujdei et al. conducted an experimental analysis to investigate the potential occurrence of CPA peaks when employing the schoolbook sub-operation in the processing of 3-way and 4-way Toom-Cook within the lattice-based PQC algorithm. The post-quantum algorithm *ntruhps*4096821 elaborated in [28], can be subjected to a multiplication-based attack utilizing side-channel measurements. Mujdei et al. study encompasses an examination of the variance plot of 500 instances of schoolbook multiplication, wherein a comprehensive analysis reveals the identification of a total of 72 apparent peaks. These peaks are specifically associated with the targeted algorithm as described in the work by [28].

PQC refers to a collection of cryptographic methods, specifically algorithms developed for the purpose of public key encapsulation, that are widely acknowledged for their ability to withstand possible attacks from quantum computers. The main goal of PQC is to strengthen and optimize mathematical methods and standards in anticipation of the emergence of quantum computing. Proficiency in mathematical approaches is essential for the development of PQC algorithms that can effectively withstand SCA. Furthermore, the utilization of effective mathematical techniques is imperative in the construction of quantum circuits, which can be employed for the creation of cryptanalysis circuits. The primary function of these cryptanalysis circuits is to evaluate the resilience of a method.

Efficient arithmetic operations, particularly multiplication, play a vital role in conducting comprehensive investigations within the domain of quantum-based cryptanalysis. According to Roche [33], Parent et al. [29], Gidney [15], Banegas et al. [3], and Putranto et al. [32], [31], the development of a fundamental arithmetic constructor that demonstrates efficiency in terms of space use and time consumption is crucial for expediting the cryptanalysis process. The primary objective of these investigations is to reduce the complexity that is typically encountered during the execution of quantum cryptanalysis. The efficacy of basic mathematical operations, particularly multiplication, can significantly impact the predictive analysis of the utilization of multiplication inside the lattice-based PQC algorithm, as well as the quantum computer's ability to solve conventional public key cryptography through cryptanalysis, which further leads to post-quantum security evaluation.

5 Complexity Analysis of High-degree and half-Multiplication

5.1 Toffoli Gate Count

The variable T_n is used to represent the cost incurred when performing multiplication on two larger n-bit quantities utilizing the Toom-Cook multiplier. Thus, A_n denotes the cost associated with the addition or substracting of n bits. To implement a n-bit Toom-Cook 8.5-way multiplication, it is necessary to perform a total of 17 operations involving $\frac{n}{9}$ submultiplications and three types of adders with different lengths. These adders consist of 46 operations for $\frac{n}{9}$-bit adders, 272 operations for $\frac{2n}{9}$-bit adders. The Toffoli cost of an n-bit Toom-Cook 8.5-way multiplication can be determined by employing the equation referenced as Equation 8. Furthermore, the cost increases to 9 for recursive implementations, and Equation 10 becomes equivalent when the Toffoli cost of $A_n = 2n$ is substituted.

$$T_n = 17T_{\frac{n}{9}} + 46A_{\frac{n}{9}} + 272A_{\frac{n}{9}} \tag{8}$$

$$
\begin{aligned}
T_n &= 17^{\log_9 n} T_1 + 46(A\frac{n}{9} + 23A\frac{n}{81} + \cdots + 23^{\log_9(n)-1} A_1) \\
&+ 272(A\frac{2n}{9} + 136A\frac{2n}{81} + \cdots + 95^{\log_9(n)-1} A_2)
\end{aligned}
\tag{9}
$$

$$
T_n = 17^{\log_9 n} + \sum_{i=0}^{\log_9(n)-1} \left[92n(\frac{17}{9})^i \right] \tag{10}
$$

By utilizing the geometric series calculation $\sum_{i=0}^{m-1} r^i = \frac{1-r^m}{1-r}$, it is possible to determine the Toffoli cost of a recursive implementation, as denoted by Equation 11. The result obtained from Equation 11 does not consider the typical uncomputation procedure carried out in a quantum environment. The strategy mentioned in this study is also discussed in previous research conducted by [29], [13], [21], and Putranto et al [32]. Equation 12 in this study incorporates the concept of uncomputed process to prevent a significant increase in the previously determined cost. It is important to acknowledge that the definition of ”clean cost” used in the subsequent equation aligns with Larasati et al.’s [21]and Putranto et al.’s [32] definitions.

$$
\begin{aligned}
T_n &= 17^{\log_9 n} + 92n\left(\frac{1-(\frac{17}{9})^{\log_9 n}}{1-(\frac{17}{9})} \right) \\
&= n^{\log_9 17} + 92n\left(\frac{1-n^{\log_9(\frac{17}{9})}}{1-(\frac{17}{9})} \right) \\
&= 93n^{\log_9 17} - 101n
\end{aligned}
\tag{11}
$$

$$
T_{n(clean)} = 186n^{\log_9 17} - 202n \tag{12}
$$

5.2 Space-Time Complexity Analysis

Bennett in [4] introduced the technique for measuring asymptotic performance improvements in the context of space consumption in the context of space-time complexity analysis. This technique is utilized extensively in reversible computing, which makes time and space complexity analysis possible and enables time-efficient finite-space computing [20]. This method will allow us to evaluate the difference in the cost of the successfully optimized multiplication and compare it to the results of previous studies. We determined the optimal cost of multiplication by following the procedures outlined in [29], [13], [21], and [32].

In the Toom-Cook 8.5-way algorithm, 17 simultaneous multiplications were done in a recursive way to make a quinary eight structure. There are 17^l nodes of size $9^{-l}n$ for an input of size n at level l, and this input has a total circuit cost of $n(\frac{15}{9})^l$. Equations 13 - 15 depict the total price of the quinary tree. For determining the optimal tree height k for optimal performance, use Equation 15.

$$
n\sum_{i=0}^{N} \left(\frac{17}{9} \right)^i, \quad N = \log_9 n \tag{13}
$$

$$n \sum_{i=0}^{N-k-1} \left(\frac{17}{9} \right)^{i} = \frac{1}{9^{N-k}} \sum_{i=0}^{k-1} \left(\frac{17}{9} \right)^{i} \tag{14}$$

In a pattern similar to Equation 12, the identity of the geometric series enables us to locate the boundaries indicated by Equation 15. Thus, the space can be reduced, as shown in qubit count Equation 16. The obtained result from Equation 16, approximately equal to $\mathcal{O}(n^{1.245})$, is lower than the initially required space assessed with Equation 17, which is confined to the value $\mathcal{O}(n^{\log_9 15}) \approx \mathcal{O}(n^{n^{1.30229}})$.

$$k \leq \frac{N}{2 - \frac{\log 9}{\log 17}} \approx 0.8167N \tag{15}$$

$$QC = \mathcal{O}\left(n \left(\frac{17}{9}^{\left(\frac{\log 17}{2 \log 17 - 2 \log 9} \right) \log_9 n} \right) \right) \approx \mathcal{O}(n^{1.236}) \tag{16}$$

$$n \sum_{k=0}^{\log_9 n - 1} \left(\frac{17}{9} \right)^{k} = n \left(\frac{1 - \left(\frac{17}{9} \right)^{\log_9 n}}{1 - \frac{17}{9}} \right) \tag{17}$$

The Toffoli depth of a circuit is a prevalent way to describe its time complexity [13], [2]. It can be calculated by multiplying the number of subtrees S_k at the $k - th$ level by the corresponding depth D_k. Consequently, we can express the Toffoli depth T_d as in Equation 18.

$$\begin{aligned} S_k &= 17^{\left(1 - \frac{\log 17}{2 \log 17 - \log 9} \right) \log_9 n} \\ D_k &= \frac{n}{9^{\left(1 - \frac{\log 17}{2 \log 17 - log 9} \right) \log_9 n}} \\ T_d &= S_k D_k = n \left(\frac{17}{9} \right)^{\left(1 - \frac{\log 17}{2 \log 15 - log 9} \right) \log_9 n} \approx n^{1.0530} \end{aligned} \tag{18}$$

5.3 Complexity Analysis Comparison

The naïve multiplication, which is equivalent to the Toom-Cook 1-way, exhibits a time complexity of $\mathcal{O}(n^2)$, where n represents the size of the input. The Toffoli depth of Naive is also of the order $\mathcal{O}(n \log n)$, according to a more in-depth study done in [12]. In the context of asymptotic performance analysis in quantum implementation, it is observed that the schoolbook technique necessitates a qubit count of $\mathcal{O}(n)$, as well as a Toffoli count and depth values of $\mathcal{O}(n^2)$. The costs associated with quantum multiplication are characterized by a qubit count of $(4n+1)$, a Toffoli depth of $(4n^2-4n+1)$, and a Toffoli count of $(4n^2 - 3n)$ [13] [21].

Karatsuba multiplication, a multiplication equivalency with the Toom-Cook 2-way approach, resulted in a qubit count of $\mathcal{O}(n^{\log_2(3)})$ for both the qubit count and Toffoli count. The improvement study reveals asymptotic values for qubit count ($\mathcal{O}(n^{1.427})$), Toffoli count ($\mathcal{O}(n^{\log_2(3)})$), and Toffoli depth ($\mathcal{O}(n^{1.158})$) [29] [13] [21]. Parent et al. [29] determined the values of the qubit count, denoted

Table 1: Asymptotic Performance and Quantum Implementation Cost Multipliers Comparison. In order to provide a comprehensive analysis of the advancements in complexity multiplication research, specifically focusing on the Karatsuba and Toom-Cook-based approaches, we provide our results pertaining to cost evaluation. This evaluation is conducted utilizing the Toffoli count, qubit count, and Toffoli depth as metrics to assess the space-time complexity.

No	Reference	Multiplication Algorithm	Asymptotic Performance Analysis			Cost of Quantum Implementation of Multiplication			
			Qubit Count	Toffoli Count	Toffoli Depth	Qubit Count	Toffoli Count	Toffoli Depth	CNOT
1	Kepley and Steinwandt (2015, [18])	Karatsuba	$\mathcal{O}(n^{\log_2 3})$	$\mathcal{O}(n^{\log_2^3})$	-	-	-	-	$\mathcal{O}(n^{\log_2 3})$
2	Parent et al. (2017, [29])	Karatsuba	$\mathcal{O}(n^{1.427})$	$\mathcal{O}(n^{\log_2 3})$	$\mathcal{O}(n^{1.158})$	$n(\frac{3}{2})^{\frac{\log 2}{(2\log 3-\log 2)}\log_2 n} \approx n^{1.427}$	$42n^{\log_2 3}$	$n(\frac{3}{2})^{1-\frac{\log 3}{(2\log 3-\log 2)}\log_2 n} \approx n^{1.158}$	-
3	Dutta et al. (2018, [13])	Toom-Cook 2.5-way	$\mathcal{O}(n^{1.404})$	$\mathcal{O}(n^{\log_6^{16}})$	$\mathcal{O}(n^{1.143})$	$n(\frac{8}{3})^{\frac{\log 16}{(6\log 16-\log 6)}\log_6 n} \approx n^{1.404}$	$49n^{log_6 16}$	$n(\frac{8}{3})^{1-\frac{\log 16}{(2\log 16-\log 6)}\log_6 n} \approx n^{1.143}$	-
4	Larasati et al.(2021, [21])	Toom-Cook 3-way	$\mathcal{O}(n^{1.35})$	$O(n^2)$	$\mathcal{O}(n^{1.112})$	$n(\frac{5}{3})^{\frac{\log 5}{(2\log 5-\log 3)}\log_3 n} \approx n^{1.353}$	$8n^2 + 66n^{\log_3 5} - 72$	$n(\frac{5}{3})^{1-\frac{\log 5}{(2\log 5-\log 3)}\log_3 n} \approx n^{1.112}$	-
5	Van Hoof (2020, [16])	Karatsuba	$3n$	$\mathcal{O}(n^{log_2 3})$	-	-	-	-	$\mathcal{O}(n^2)$
6	Putranto et al. (2023), [31])	Karatsuba	$3n$	$\mathcal{O}(n^{log_2 3})$	-	-	-	-	$\mathcal{O}(n^{log_2 3})$
7	Putranto et al. (2023, [32])	Toom Cook 2-way	$\mathcal{O}(n^{1.589})$	$\mathcal{O}(n^{\log_2 3})$	$\mathcal{O}(n^{1.217})$	$n(\frac{3}{2})^{\frac{\log 3}{(2\log 3-\log 2)}\log_2 n} \approx n^{1.589}$	$34n^{\log_2 3} - 32n$	$n(\frac{3}{2})^{1-\frac{\log 3}{(2\log 3-\log 2)}\log_2 n} \approx n^{1.217}$	-
8	Putranto et al. (2023, [32])	Toom Cook 4-way	$\mathcal{O}(n^{1.313})$	$\mathcal{O}(n^{\log_4 7})$	$\mathcal{O}(n^{1.09})$	$n(\frac{7}{4})^{\frac{\log 7}{(2\log 7-\log 4)}\log_4 n} \approx n^{1.313}$	$122n^{\log_4 7} - 160n$	$n(\frac{7}{4})^{1-\frac{\log 7}{(2\log 7-\log 4)}\log_4 n} \approx n^{1.09}$	-
9	Putranto et al. (2023, [32])	Toom Cook 8-way	$\mathcal{O}(n^{1.245})$	$\mathcal{O}(n^{\log_8 15})$	$\mathcal{O}(n^{1.0569})$	$n(\frac{15}{8})^{\frac{\log 15}{(2\log 15-\log 8)}\log_8 n} \approx n^{1.245}$	$112n^{\log_8 15} - 128n$	$n(\frac{15}{8})^{1-\frac{\log 15}{(2\log 15-\log 8)}\log_8 n} \approx n^{1.0569}$	-
10	our	Toom-Cook 8.5-way	$\mathcal{O}(n^{1.236})$	$\mathcal{O}(n^{\log_9 17})$	$\mathcal{O}(n^{1.053})$	$n(\frac{17}{9})^{\frac{\log 17}{(2\log 17-\log 9)}\log_9 n} \approx n^{1.236}$	$186n^{\log_9 17} - 202n$	$n(\frac{17}{9})^{1-\frac{\log 17}{(2\log 17-\log 9)}\log_9 n} \approx n^{1.053}$	-

as $n^{1.427}$, the Toffoli count, denoted as $\mathcal{O}(n^{\log_2 3})$, and the Toffoli depth, denoted as $n^{1.158}$ for Karatsuba. Recently, the Karatsuba variant proposed by Putranto et al. [31] demonstrates a reduction in CNOT usage, changing the $\mathcal{O}(n^2)$ CNOT in the prior work to $\mathcal{O}(n^{\log_2(3)})$.

According to Dutta et al. [13], the Toom-Cook 2.5-way algorithm offers a potential approach for reducing the cost of developing quantum systems by achieving the qubit count ($n^{1.404}$), Toffoli count ($49n^{\log_6 16}$), and Toffoli depth ($n^{1.143}$). Later, Larasati et al. [21] present a comprehensive examination of the asymptotic performance metrics for qubit count, Toffoli count, and Toffoli depth. They report an estimated value of $n^{1.353}$ for the qubit count, $\mathcal{O}(n^2)$ for the Toffoli count, and $n^{1.112}$ for the Toffoli depth.

Recently, from Putranto et al. [32] elaboration, they exhibit a better asymptotic performance analysis in terms of qubit count for the Toom-Cook 8-way approach. Specifically, it is approximated by qubit count with $n(\frac{15}{8})^{\frac{\log 15}{(2\log 15-\log 8)}\log_8 n}$, which is of the order $\mathcal{O}(n^{1.245})$. In the context of Toffoli depth, which is relevant to efficient computation, the Toom-Cook 8-way design results in a lower bound on logical depth of $\mathcal{O}(n^{1.0569})$ and a Toffoli count of $\mathcal{O}(n^{\log_8 15})$.

In the present study, as presented in Table 1, a comparative analysis of various multiplication methods reveals that the Toom-Cook high-degree and half-multiplier, established in this research, demonstrates the lowest desired asymptotic performance in terms of qubit count, Toffoli count, and Toffoli depth when compared to other approaches. In terms of cost, the proposed multiplication in quantum implementation demonstrates lower quantum resources when compared to the alternative Toom-Cook strategy. The high-degree and half-multiplication, specifically the Toom-Cook 8.5-way approach, involves a qubit count of $\mathcal{O}(n^{1.236})$, a logical Toffoli depth of $n(\frac{17}{9})^{1-\frac{\log 17}{(2\log 17-\log 9)}\log_9 n} \approx n^{1.053}$, and a Toffoli count of $186n^{\log_9 17} - 202n$.

6 Discussion

Empirical research has provided evidence indicating that while higher-order procedures may exhibit superior efficiency, the incorporation of the division operation, a crucial component of the

$k-$way Toom-Cook method, can provide difficulties in terms of identifying an effective strategy. In the current research, as shown in Table 1, using the Toom-Cook-8.5 approach and yielding complexity analysis ($\mathcal{O}(n^{1.236})$ qubit Count, $\mathcal{O}(n^{\log_9 17})$ Toffoli Count, and Toffoli Depth of $\mathcal{O}(n^{1.053})$), we established the optimal utilization of resources for multiplication operations. Nevertheless, the design multiplication was not incorporated into the PQC algorithm, and the notable cryptanalysis using the Shor algorithm technique was also not performed. In later stages, it is imperative to also enhance the implementation of a higher degree in the PQC algorithm and provide a more comprehensive examination of multiplication-based attacks employing SCA, or correlation power analysis, methodologies.

Further, it should be noted that the efficiency of the recently developed Toom-Cook method exceeds that of the currently employed Toom-Cook-based multiplication techniques, Karatsuba, and naive schoolbooks. This demonstrates a higher level of efficiency in comparison to existing multipliers based on the Toom-Cook method currently utilized as part of the lattice-based algorithm, the Toom-Cook 4-way approach. In this work, the multiplication is also designed in a quantum environment, facilitating its integration into quantum circuits for cryptanalysis (e.g., [3], [31]). This integration will thereafter enable the evaluation of security in the post-quantum era.

7 Conclusions

The present study undertook a thorough examination of high-degree and half-multiplication, focusing particularly on the Toom-Cook 8.5-way algorithm. The study demonstrated the achievement of the lowest or most optimal multiplication, which is distinguished by its lower asymptotic performance and fewer demands on quantum resources compared to other multiplications. The proposed multiplication was subjected to asymptotic performance analysis, resulting in a qubit count of $n(\frac{17}{9})^{\frac{\log 17}{(2\log 17-\log 9)}\log_9 n} \approx n^{1.236}$, approximately $\mathcal{O}(n^{1.236})$. Additionaly, the Toom-Cook 8.5-way has a Toffoli count of $186n^{\log_9 17} - 202n$ and a Toffoli depth of $n(\frac{17}{9})^{1-\frac{\log 17}{(2\log 17-\log 9)}\log_9 n} \approx n^{1.053}$ for multiplication.

The alternative methods that have been proposed have the potential to reduce the computational resources needed and can result in efficient multiplication with high degrees of multiplication. As part of planned future research, the suggested multiplication operation could be used as an alternative to constructing lattice-based post-quantum algorithms while lowering the risks of attacks that use multiplication. Furthermore, the multiplication technique is intended to be incorporated into a quantum cryptanalysis circuit in order to enhance the efficiency of evaluating post-quantum security.

References

1. Alagic, G., Apon, D., Cooper, D., Dang, Q., Dang, T., Kelsey, J., Lichtinger, J., Miller, C., Moody, D., Peralta, R., et al.: Status report on the third round of the nist post-quantum cryptography standardization process. US Department of Commerce, NIST (2022)
2. Amy, M.: Algorithms for the optimization of quantum circuits. Master's thesis, University of Waterloo (2013)
3. Banegas, G., Bernstein, D.J., van Hoof, I., Lange, T.: Concrete quantum cryptanalysis of binary elliptic curves. IACR Transactions on Cryptographic Hardware and Embedded Systems pp. 451–472 (2021)
4. Bennett, C.H.: Time/space trade-offs for reversible computation. SIAM Journal on Computing **18**(4), 766–776 (1989)

5. Bisheh-Niasar, M., Azarderakhsh, R., Mozaffari-Kermani, M.: High-speed ntt-based polynomial multiplication accelerator for post-quantum cryptography. In: 2021 IEEE 28th Symposium on Computer Arithmetic (ARITH). pp. 94–101. IEEE (2021)
6. Bisheh-Niasar, M., Azarderakhsh, R., Mozaffari-Kermani, M.: Instruction-set accelerated implementation of crystals-kyber. IEEE Transactions on Circuits and Systems I: Regular Papers **68**(11), 4648–4659 (2021)
7. Bodrato, M.: Towards optimal toom-cook multiplication for univariate and multivariate polynomials in characteristic 2 and 0. In: International Workshop on the Arithmetic of Finite Fields. pp. 116–133. Springer (2007)
8. Bodrato, M.: High degree toom'n'half for balanced and unbalanced multiplication. In: 2011 IEEE 20th Symposium on Computer Arithmetic. pp. 15–22. IEEE (2011)
9. Bos, J., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schanck, J.M., Schwabe, P., Seiler, G., Stehlé, D.: Crystals-kyber: a cca-secure module-lattice-based kem. In: 2018 IEEE European Symposium on Security and Privacy (EuroS&P). pp. 353–367. IEEE (2018)
10. Brier, E., Clavier, C., Olivier, F.: Correlation power analysis with a leakage model. In: International workshop on cryptographic hardware and embedded systems. pp. 16–29. Springer (2004)
11. Cook, S.A., Aanderaa, S.O.: On the minimum computation time of functions. Transactions of the American Mathematical Society **142**, 291–314 (1969)
12. Draper, T.G., Kutin, S.A., Rains, E.M., Svore, K.M.: A logarithmic-depth quantum carry-lookahead adder. Quantum Information and Computation **6**(4) (2006)
13. Dutta, S., Bhattacharjee, D., Chattopadhyay, A.: Quantum circuits for toom-cook multiplication. Physical Review A **98**(1), 012311 (2018)
14. Ghosh, A., Mera, J.M.B., Karmakar, A., Das, D., Ghosh, S., Verbauwhede, I., Sen, S.: A 334 microwatt 0.158 mm2 asic for post-quantum key-encapsulation mechanism saber with low-latency striding toom-cook multiplication authors version. arXiv preprint arXiv:2305.10368 (2023)
15. Gidney, C.: Asymptotically efficient quantum karatsuba multiplication. arXiv preprint arXiv:1904.07356 (2019)
16. van Hoof, I.: Space-efficient quantum multiplication polynomials for binary finite fields with sub–quadratic toffoli gate count. Quantum Inf.Comput. **20**(9&10), 721–735 (2020). https://doi.org/10.26421/QIC20.9-10-1, `https://doi.org/10.26421/QIC20.9-10-1`
17. Karatsuba, A.A., Ofman, Y.P.: Multiplication of many-digital numbers by automatic computers. In: Doklady Akademii Nauk. vol. 145, pp. 293–294. Russian Academy of Sciences (1962)
18. Kepley, S., Steinwandt, R.: Quantum circuits for f2n-multiplication with subquadratic gate count. Quantum Information Processing **14**(7), 2373–2386 (2015)
19. Kocher, P., Jaffe, J., Jun, B.: Differential power analysis. In: Annual international cryptology conference. pp. 388–397. Springer (1999)
20. Král'ovič, R.: Time and space complexity of reversible pebbling. In: International Conference on Current Trends in Theory and Practice of Computer Science. pp. 292–303. Springer (2001)
21. Larasati, H.T., Awaludin, A.M., Ji, J., Kim, H.: Quantum circuit design of toom 3-way multiplication. Applied Sciences **11**(9), 3752 (2021)
22. Larasati, H.T., Awaludin, A.M., Ji, J., Kim, H.: Quantum circuit design of toom 3-way multiplication. Applied Sciences **11**(9), 3752 (2021)
23. Li, Y., Zhu, J., Huang, Y., Liu, Z., Tang, M.: Single-trace side-channel attacks on the toom-cook: The case study of saber. IACR Transactions on Cryptographic Hardware and Embedded Systems pp. 285–310 (2022)
24. Liu, Z., Choo, K.K.R., Grossschadl, J.: Securing edge devices in the post-quantum internet of things using lattice-based cryptography. IEEE Communications Magazine **56**(2), 158–162 (2018)
25. Lyubashevsky, V.: Fiat-shamir with aborts: Applications to lattice and factoring-based signatures. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 598–616. Springer (2009)

26. Mera, J.M.B., Karmakar, A., Verbauwhede, I.: Time-memory trade-off in toom-cook multiplication: an application to module-lattice based cryptography. Cryptology ePrint Archive (2020)
27. Micciancio, D., Regev, O.: Post-quantum cryptography, chapter lattice-based cryptography. Computing **85**(1-2), 105–125 (2008)
28. Mujdei, C., Wouters, L., Karmakar, A., Beckers, A., Mera, J.M.B., Verbauwhede, I.: Side-channel analysis of lattice-based post-quantum cryptography: Exploiting polynomial multiplication. ACM Transactions on Embedded Computing Systems (2022)
29. Parent, A., Roetteler, M., Mosca, M.: Improved reversible and quantum circuits for karatsuba-based integer multiplication. In: 12th Conference on the Theory of Quantum Computation, Communication, and Cryptography (TQC 2017). pp. 7:1–7:15. Springer (2017)
30. Pollard, J.M.: The fast fourier transform in a finite field. Mathematics of computation **25**(114), 365–374 (1971)
31. Putranto, D.S.C., Wardhani, R.W., Larasati, H.T., Ji, J., Kim, H.: Depth-optimization of quantum cryptanalysis on binary elliptic curves. IEEE Access (2023)
32. Putranto, D.S.C., Wardhani, R.W., Larasati, H.T., Kim, H.: Space and time-efficient quantum multiplier in post quantum cryptography era. IEEE Access **11**, 21848–21862 (2023)
33. Roche, D.S.: Space-and time-efficient polynomial multiplication. In: Proceedings of the 2009 international symposium on Symbolic and algebraic computation. pp. 295–302 (2009)
34. Toom, A.L.: The complexity of a scheme of functional elements realizing the multiplication of integers. In: Soviet Mathematics Doklady. vol. 3, pp. 714–716 (1963)
35. Zanoni, A.: Toom-cook 8-way for long integers multiplication. In: 2009 11th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing. pp. 54–57. IEEE (2009)