

简化 AES 密码算法的量子电路优化

丁 浪¹ 罗庆斌¹ 吕 轶¹ 郑圆梦² 廖颢羽²

1 湖北民族大学智能科学与工程学院 湖北 恩施 445000

2 湖北民族大学数学与统计学院 湖北 恩施 445000

(dinglang1101@163.com)

摘 要 AES 是当前最广泛使用的国际标准化分组密码算法,美国国家标准与技术研究院(NIST)将 AES 的量子安全强度作为评估后量子密码安全性的参考,因此实现 AES 算法的量子电路并分析其量子安全性已成为密码学研究的热点。然而,由于 AES 算法的量子电路实现需耗费数百个量子比特和数万个量子门,简化 AES 密码算法的量子电路实现与优化成为重要的研究方向。首先,在加密算法中,基于 S 盒查找表,利用 DORCIS 工具成功实现 S 盒的量子电路;其次,通过借用密钥中的一个量子比特,将该量子电路中具有 3 个控制位的 CCCNOT 门分解为 4 个 Toffoli 门,将使用的量子门控制在 NCT 门集内;然后,在移位操作中,通过置换变量的方式避免了加密算法中交换门的使用;最后,通过 S 盒查找表计算出了 S 盒的布尔表达式,设计并实现了密钥扩展中 8 量子比特的 S 盒量子电路。在此基础上优化了简化 AES 密码算法的量子电路,该量子电路的正确性在 Qiskit Aer 量子模拟器中得到了验证。量子资源分析结果表明,整体量子电路实现仅需 32 个量子比特、51 个 NOT 门、220 个 CNOT 门和 120 个 Toffoli 门。与已有研究相比,所提方法减少了量子资源的消耗,从而提升了简化 AES 算法量子电路的实现效率。

关键词: 分组密码;简化 AES;S 盒;量子电路;布尔表达式

中图分类号 TP309

Quantum Circuit Optimization for Simplified AES Cryptographic Algorithm

DING Lang¹, LUO Qingbin¹, LYU Yi¹, ZHENG Yuanmeng² and LIAO Haoyu²

1 College of Intelligent Systems Science and Engineering, Hubei Minzu University, Enshi, Hubei 445000, China

2 School of Mathematics and Statistics, Hubei Minzu University, Enshi, Hubei 445000, China

Abstract AES is currently the most widely used internationally standardized block cipher algorithm. The National Institute of Standards and Technology(NIST) uses the quantum security strength of AES as a reference for evaluating the security of post-quantum cryptography. Therefore, implementing the quantum circuit of the AES algorithm and analyzing its quantum security has become a research hotspot in cryptography. However, since implementing the AES algorithm's quantum circuit requires hundreds of qubits and tens of thousands of quantum gates, making the implementation and optimization of Simplified AES quantum circuits has become an important research direction. This study successfully implements the S-box quantum circuit using the DORCIS tool based on the S-box lookup table, decomposes the CCCNOT gate into four Toffoli gates by borrowing one qubit, and avoids swap gates in shift operations by permuting variables. Additionally, an 8-qubit S-box quantum circuit for key expansion is designed and implemented using the Boolean expression of the S-box. The optimized S-AES quantum circuit is verified in the Qiskit Aer simulator, requiring only 32 qubits, 51 NOT gates, 220 CNOT gates, and 120 Toffoli gates. Compared to existing methods, this approach significantly reduces quantum resource consumption, enhancing the efficiency of implementing the Simplified AES quantum circuit.

Keywords Block cipher, Simplified AES, S box, Quantum circuit, Boolean expression

1 引言

与经典计算相比,量子计算在某些特定类型的问题(如大整数分解和离散对数问题)上,能够凭借量子叠加与量子纠缠

实现显著加速,有时甚至可将经典算法的指数级时间复杂度降至多项式级。这使得量子计算在处理这些问题时展现出潜在的巨大优势,从而可能威胁到基于此类数学难题的公钥密码体制的安全性。例如,基于离散对数问题的 ElGamal 密码

基金项目:国家自然科学基金(62262020,12164037);湖北省自然科学基金联合基金项目(2024AFD066);湖北民族大学研究生教育创新项目(MYK2025076)

This work was supported by the National Natural Science Foundation of China(62262020,12164037),Hubei Provincial Natural Science Foundation Joint Fund(2024AFD066) and Hubei Minzu University Graduate Education Innovation Project(MYK2025076).

通信作者:罗庆斌(qingbinluo@126.com)

系统^[1]可以通过 Shor 算法^[2]在量子计算机上以多项式时间被攻破。类似地, Simon 算法^[3]对在 CBC-MAC, PMAC, GMAC 和 OCB 等模式^[4]下使用分组密码的安全性构成了严重威胁。Grover 算法^[5]则为任何穷举搜索攻击提供了二次加速,理论上将分组密码的安全级别从 n 位降低到 $n/2$ 位。

在量子环境下评估密码算法的安全性已成为密码学研究的热门趋势。安全性分析的前提是实现能够执行目标密码算法功能的量子电路,因此如何高效地构建和优化这些量子电路已成为密码学界的重要研究方向。现有的分组密码(如 AES, SM4 等)的量子电路实现主要集中在 AES 上。Chung 等^[6]采用高效塔式域构造以降低在有限域 $GF(2^8)$ 上乘法逆运算的 AND 深度。通过将 8 位逆运算分解为 3 个 4 位乘法、1 个 4 位逆运算以及其他一些次要操作,降低了 AES 量子电路的总体深度和宽度。Liu 等^[7]引入混合 XOR 技术,通过重复利用空闲量子比特,减少了 AES S 盒的辅助量子比特数。此外,还提出了结合共享技术的组合流水线架构,以同时结合 S 盒及其逆操作,进一步优化了量子资源的使用。Shi 等^[8]提出了一种改进的贪婪算法,用于加深 AES 量子电路的整体深度,同时提出一种名为压缩流水线结构的新结构,用于设计加密电路中的迭代轮函数和新的密钥扩展电路。

在量子电路中, Toffoli 门的数量与布局可能影响电路的 Toffoli 深度,并可能增加对辅助量子比特的需求。由于 Toffoli 门的时间和空间实现成本是 CNOT 门的几十倍甚至上百倍^[9],减少量子电路中 Toffoli 门的数量具有重要研究价值。

S 盒作为分组密码中关键的非线性组件^[10],其量子实现的资源消耗成为密码算法量子资源评估中的重要因素。现有文献中^[11-13],尽管对简化 AES(S-AES)^[14]进行了多种经典密码学分析,但针对其量子实现的研究仍然有限。在量子环境中,已有文献探索了不同方法来实现 S-AES 的 S 盒。文献^[15]提出了基于费马求逆法的有限域乘法逆操作实现方法。该方案实现 S 盒量子电路需要 16 个量子比特、2 个 NOT 门、55 个 CNOT 门和 48 个 Toffoli 门;整体量子电路实现则需 72 个量子比特、26 个 NOT 门、1 080 个 CNOT 门和 576 个 Toffoli 门。文献^[10]使用布尔函数设计了另一种 S 盒量子电路,资源消耗进一步降低,需要 8 个量子比特、2 个 NOT 门、23 个 CNOT 门和 14 个 Toffoli 门;整体实现需要 48 个量子比特、75 个 NOT 门、364 个 CNOT 门和 168 个 Toffoli 门。相比之下,文献^[16]采用了 S 盒查表法,并结合 LIGHTER-R 工具^[17]高效实现量子电路,不仅资源消耗进一步减少,还无需额外的量子比特。该方法共使用了 4 个量子比特、2 个 NOT 门、3 个 CNOT 门、6 个 Toffoli 门和 1 个三控制位的 CCCNOT 门;整体实现需要 32 个量子比特、35 个 NOT 门、144 个 CNOT 门、96 个 Toffoli 门和 16 个三控制位的 CCCNOT 门,成为目前量子资源消耗最优的实现方案。综上所述,尽管文献^[16]在 S 盒和整体量子电路资源消耗方面优于文献^[10]和文献^[15],但在其密钥扩展的量子实现过程中需多次使用 S 盒及其逆操作,这正是本文优化的重点之一。

本文从加密和密钥扩展两个方面对 S-AES 算法的量子电路进行了优化研究,主要工作如下。

1) S 盒量子电路优化:基于 S 盒的查表法,本文利用自动

化工具 DORCIS^[18],以低成本实现了仅需 4 个量子比特、3 个 NOT 门、8 个 CNOT 门、4 个 Toffoli 门和 1 个三控制位 CCCNOT 门的量子电路,是目前的最优实现结果。

2) CCCNOT 门替代方案:在 S 盒实现基础上,本文通过借用密钥中的一个量子比特,将三控制位的 CCCNOT 门替换为 4 个 Toffoli 门。该方案无需额外增加量子比特,并将量子门集限制在 NCT 门库内,便于资源评估。

3) 密钥扩展优化:针对密钥扩展中的 S 盒量子电路,本文通过计算 S 盒的布尔表达式,设计了 8 量子比特的电路。该电路在还原输入位的同时,将输出结果直接异或到目标密钥的 4 个量子比特上。此设计免除了输入密钥值的还原操作,简化了异或操作流程,并减少了所需的 Toffoli 门数量。

4) 加密过程优化:在加密过程中,本文通过变量置换取消了 S-AES 的移位操作,减少了交换门的使用。由于一个交换门需使用 3 个 CNOT 门,该优化有效降低了量子资源的消耗。

2 预备知识

2.1 常用量子逻辑门

在量子计算中,基本量子门是操纵量子信息的核心模块,用于实现量子态的变换和量子算法的逻辑操作。本文主要使用 NCT 库,即 NOT 门、CNOT 门和 Toffoli 门。

NOT 门(也称为 X 门)将一个量子比特 $|B\rangle$ 映射为 $|B\oplus 1\rangle$,如图 1(a)所示。CNOT 门将两个量子比特 $|B_1\rangle$ 和 $|B_2\rangle$ 映射为 $|B_1\rangle|B_2\rangle \rightarrow |B_1\rangle|B_2\oplus B_1\rangle$,如图 1(b)所示,其中 $|B_1\rangle$ 是控制量子比特, $|B_2\rangle$ 是目标量子比特。Toffoli 门将 3 个量子比特 $|B_1\rangle$, $|B_2\rangle$ 和 $|B_3\rangle$ 映射为 $|B_1\rangle|B_2\rangle|B_3\rangle \rightarrow |B_1\rangle|B_2\rangle|B_3\oplus B_1B_2\rangle$,如图 1(c)所示,其中 $|B_1\rangle$ 和 $|B_2\rangle$ 是控制量子比特, $|B_3\rangle$ 是目标量子比特。

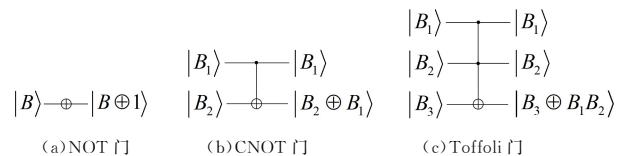


图 1 常用量子逻辑门

Fig. 1 Common quantum gates

2.2 简化 AES 加密算法

简化 AES(S-AES)对理解和分析 AES 分组密码算法具有重要作用。它保留了 AES 算法的核心结构,但为了便于理解和实现,简化了密钥和分组的大小以及轮数。具体而言, S-AES 的分组长度与密钥长度均为 16 位,且整体加密过程仅需两轮迭代。这一简化设计有助于更加直观地理解 AES 加密过程的基本原理。

S-AES 的加密过程主要包括半字节置换(Sbox)、行移位(SR)、混合列(MC)和轮密钥异或(AddKey)4 个步骤。与 AES 类似, S-AES 的最后一轮不执行混合列操作。

在轮密钥异或操作中,16 位的状态块和 16 位的轮密钥进行异或运算,生成新的 16 位状态块。S-AES 以半字节(4 位)为单位对状态块执行替换操作, S 盒的查找表如表 1 所列。

表 1 S 盒查找
Table 1 S-box lookup

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$Sbox(x)$	9	4	A	B	D	1	8	5	6	2	0	3	C	E	F	7

随后,行移位操作将状态块中的 4 位 (s_4, s_5, s_6, s_7) 和 4 位 ($s_{12}, s_{13}, s_{14}, s_{15}$) 进行互换。最后,对整个 16 位状态块执行混合列操作,具体过程为:

$$\begin{bmatrix} B_0' & B_2' \\ B_1' & B_3' \end{bmatrix} = \begin{bmatrix} 1 & x^2 \\ x^2 & 1 \end{bmatrix} \begin{bmatrix} B_0 & B_2 \\ B_1 & B_3 \end{bmatrix} \bmod (x^4 + x + 1)$$

2.3 简化 AES 密钥扩展算法

S-AES 的密钥扩展算法的设计类似于 AES 的密钥扩展算法。对于简化的 AES 版本,它使用一个 16 位的密钥,记作 ($k_0, \dots, k_{15}$)。该密钥需要扩展到总共 48 位,即生成 ($k_0, \dots, k_{47}$),其中前 16 位密钥与原始密钥相同,其扩展过程如下。

令 $RC[i] = x^{i+2} \in GF(2^4)$,所用到的不可约多项式为:
 $x^4 + x + 1$

因此, $RC[1] = x^3 = 1\ 000$, $RC[2] = x^4 = x + 1 = 0011$,其中轮常量为 $RCON[i] = RC[i]0000$ ($RCON$ 为 round constant 的缩写)。

定义函数 SwapNib 为:

$$SwapNib(N_0 N_1) = N_1 N_0$$

以及函数 SubNib 为:

$$SubNib(N_0 N_1) = Sbox(N_0) Sbox(N_1)$$

它们分别是交换半字节和替换半字节的缩写,其中 N_0 和 N_1 都是半字节。然后定义一个数组 W 以字节(8 位)为单位, $W[0]$ 和 $W[1]$ 依次填充原始密钥,扩展过程如算法 1 所示。

算法 1

输入:8 位的原始密钥 $W[0]$ 和 $W[1]$

输出:8 位的密钥 $W[i]$

其中 $2 \leq i \leq 5$ 。

当 i 为偶数时(即 i 除以 2 的余数为 0):

步骤 1 取前第二个字 $W[i-2]$ 。

步骤 2 应用轮常数函数 $RCON(i/2)$ 。

步骤 3 对前第一个字 $W[i-1]$ 进行位交换(SwapNib),然后通过 S 盒(SubNib)进行非线性替换。

步骤 4 将以上 3 个结果进行异或($\oplus$)运算,得到当前字 $W[i]$:

$$W[i] = W[i-2] \oplus RCON(i/2) \oplus SubNib(SwapNib(W[i-1]))$$

当 i 为奇数时(即 i 除以 2 的余数不为 0):

步骤 1 取前第二个字 $W[i-2]$ 和前第一个字 $W[i-1]$ 。

步骤 2 将这两个字进行异或($\oplus$)运算,得到当前字 $W[i]$:

$$W[i] = W[i-2] \oplus W[i-1]$$

数组 W 中包含的位可以表示为 ($k_0, \dots, k_{47}$)。当 $0 \leq i \leq 2$ 时,令 $K_i = W[2i]W[2i+1]$,因此 $K_0 = k_0 \dots k_{15}$, $K_1 = k_{16} \dots k_{31}$, $K_2 = k_{32} \dots k_{47}$ 。对于 $i \geq 1$, K_i 是在第 i 轮结束时使用的轮密钥, K_0 在第一轮之前使用。

3 量子电路优化

3.1 加密算法的量子电路优化

S 盒是 S-AES 密码算法中的关键非线性组件。在经典

计算中,S 盒的输入可以通过查找表直接映射到输出。然而,在量子计算中,由于量子操作必须是可逆且幺正的,传统的查找表方法无法直接用于实现 S 盒映射。因此,需要通过基本量子门构建可逆的量子电路来实现 S 盒的量子映射^[19]。基于量子门实现的 S 盒消耗大量量子资源,成为 S-AES 加密与密钥扩展过程中资源消耗最为显著的组件之一。为此,本文给出了经过优化的 S 盒量子电路(见图 2),以有效降低相关量子资源的使用。

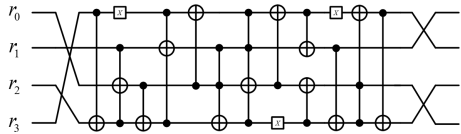


图 2 用 DORCIS 实现的 S 盒量子电路

Fig. 2 S-box quantum circuit implemented with DORCIS

为了实现高效的 S 盒设计,本文根据表 1,使用 DORCIS 工具生成了优化的量子电路。DORCIS 是一种自动化工具,相较于基础的 LIGHTER-R 工具,DORCIS 通过量子门分解并在量子深度和 T 深度上进行优化,从而实现了更高效的电路。图 2 展示了通过 DORCIS 工具实现的 S 盒量子电路。该电路无需辅助量子比特,但需要使用一个三控制位的 CCCNOT 门。优化后的 S 盒电路所需的量子资源包括 3 个 NOT 门、8 个 CNOT 门、4 个 Toffoli 门和 1 个三控制位的 CCCNOT 门。

对于图 2 中的三控制位 CCCNOT 门的分解,本文借鉴了文献[20]中的方案,利用一个包含 5 个量子比特的电路,将三控制位 CCCNOT 门分解为由 4 个 Toffoli 门组成的电路结构,并能够恢复任意辅助量子比特的状态(见图 3)。在此基础上,本文进一步分析了 S-AES 的密钥扩展过程,发现可以在分解过程中利用密钥中的一个量子比特作为辅助资源,从而实现更高效的电路设计。这样不仅将所需门集限制在 NCT 门库内,还避免了增加额外的量子比特的需求。

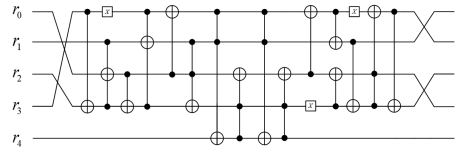


图 3 借用密钥的一个量子比特将 CCCNOT 门分解后 S 盒的量子电路

Fig. 3 S-box quantum circuit after decomposing the CCCNOT gate by borrowing one qubit from the key

S-AES 加密过程共需迭代两轮,其中轮密钥的异或操作共执行 3 次。本文采用 CNOT 门实现这些异或操作,总共消耗了 48 个 CNOT 门。对于置换操作,本文使用了如图 3 所示的 S 盒量子电路,加密电路一共使用了 8 个 S 盒,需要的量子资源消耗为 24 个 NOT 门、64 个 CNOT 门和 64 个 Toffoli 门。在移位操作中,本文通过直接置换变量的方式避免了交换门的使用,从而节省了资源(因为一个交换门通常需要 3

个 CNOT 门来实现)。混合列操作仅需执行一轮,根据文献 [16] 的实现结果,将 16 位的量子块分成两组,在 8 量子比特的量子电路中执行混合列操作,如图 4 所示,该操作仅需 16 个 CNOT 门。所提出的加密算法量子电路结构如图 5 所示。综上所述,S-AES 加密部分所需的量子资源为 24 个 NOT 门、128 个 CNOT 门和 64 个 Toffoli 门,如表 2 所列。

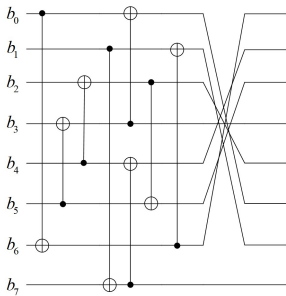


图 4 混合列量子电路
Fig. 4 MixColumns quantum circuit

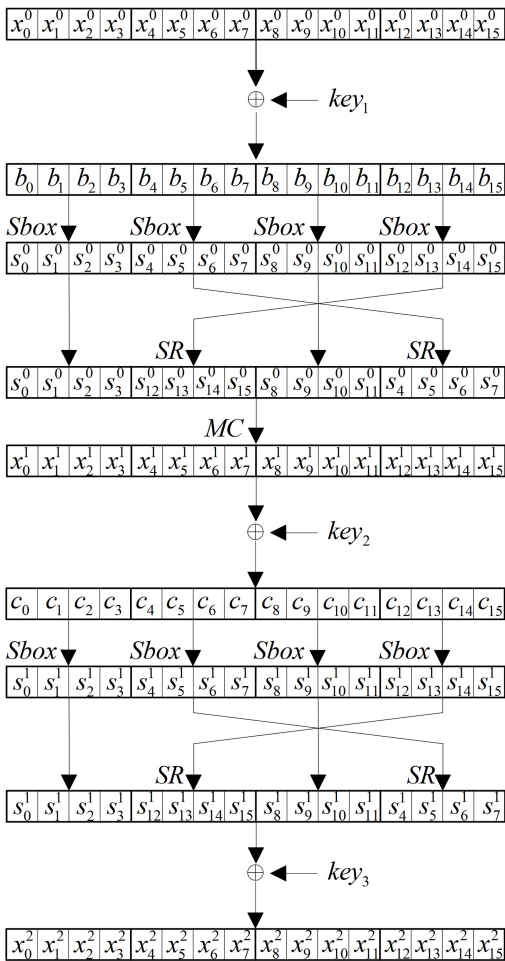


图 5 S-AES 加密方案
Fig. 5 Simplified AES encryption scheme

表 2 用于 S-AES 加密量子电路的量子资源
Table 2 Quantum resources for the Simplified AES encryption quantum circuit

文献	Qubits	NOT	CNOT	Toffoli
[15]	32	16	512	384
[10]	32	48	256	112
[16]	16	16	88	80
本文	16	24	128	64

3.2 密钥扩展算法的量子电路优化

在文献 [16] 中,为了避免在实现密钥扩展的量子电路时添加辅助量子比特用于存储轮密钥,通过执行 S 盒的逆操作来恢复先前的密钥值。尽管这种方法实现了无需额外量子比特的密钥扩展,但显然并非最优方案。本文根据表 1,计算得出 S 盒 4 个输出的布尔表达式,具体如下:

$$\begin{aligned} f_0 &= r_0 + r_3 + r_0 r_1 + r_0 r_3 + r_2 r_3 + r_0 r_1 r_2 + r_0 r_1 r_3 + 1 \\ f_1 &= r_0 + r_2 + r_3 + r_0 r_1 + r_1 r_2 + r_1 r_3 + r_2 r_3 + r_0 r_1 r_2 + \\ &\quad r_0 r_2 r_3 \\ f_2 &= r_1 + r_3 + r_1 r_2 + r_2 r_3 + r_0 r_1 r_3 + r_0 r_2 r_3 + r_1 r_2 r_3 \\ f_3 &= r_0 + r_1 + r_3 + r_0 r_2 + r_0 r_3 + r_1 r_3 + r_0 r_1 r_2 + r_0 r_1 r_3 + \\ &\quad r_0 r_2 r_3 + r_1 r_2 r_3 + 1 \end{aligned}$$

综合这 4 个布尔表达式,设计并实现了一个含有 8 个量子比特的 S 盒量子电路,如图 6 所示。该 S 盒电路保持输入量子比特 (r_0, r_1, r_2, r_3) 不变,以便后续进行异或操作;输出则直接异或到量子比特 (f_0, f_1, f_2, f_3) 上,其中 (r_0, r_1, r_2, r_3) 对应相应异或操作的 4 位密钥量子比特。该电路共用 6 个 NOT 门、19 个 CNOT 门、14 个 Toffoli 门以及 8 个量子比特。相比之下,文献 [16] 通过直接使用 S 盒及其逆操作来实现密钥扩展,并在此基础上将 S 盒的输出与密钥进行异或操作,其所需量子资源包括 4 个 NOT 门、10 个 CNOT 门、12 个 Toffoli 门、2 个三控制位的 CCCNOT 门以及 8 个量子比特。根据文献 [21],一个具有 n 个控制位的广义 Toffoli 门 (多控制位 Toffoli 门) 可以分解为 $2(n-1)$ 个 Toffoli 门、1 个 CNOT 门以及 $n-1$ 个辅助量子比特。在此基础上,本文对文献 [16] 中的三控制位 CCCNOT 门进行了分解,需使用 4 个 Toffoli 门、1 个 CNOT 门以及 1 个辅助量子比特。分解后,资源消耗为 4 个 NOT 门、12 个 CNOT 门、20 个 Toffoli 门以及 10 个量子比特。由于在 NCT 门库中 Toffoli 门的实现成本较高,本文提出的量子电路方案相比文献 [16] 节省了 6 个 Toffoli 门和 2 个量子比特,从而优化了量子资源的消耗。

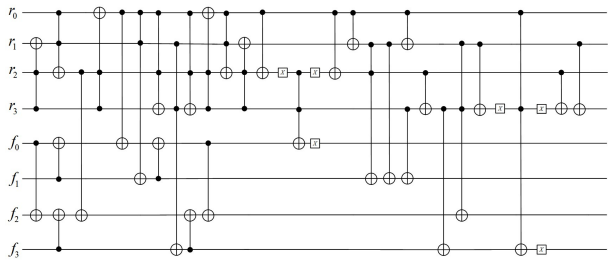


图 6 密钥扩展算法的 8-qubit S 盒量子电路
Fig. 6 8-qubit S-box quantum circuit of key expansion algorithm

最终实现 S-AES 密钥扩展所需的量子资源如表 3 所列。在密钥扩展的量子电路设计中,本文采用了优化的 8 量子比特 S 盒电路。该电路执行了 4 次 S 盒操作,总资源消耗为 24 个 NOT 门、76 个 CNOT 门和 56 个 Toffoli 门。此外,密钥之间的异或操作消耗了 16 个 CNOT 门。对于轮常量的加操作,由于轮常量是已知的,本文在常数位为 1 的位置直接应用 NOT 门以实现取非操作,总计消耗 3 个 NOT 门。综上所述,S-AES 密钥扩展所需的总量子资源为 27 个 NOT 门、92 个 CNOT 门和 56 个 Toffoli 门。

表 3 用于 S-AES 密钥扩展量子电路的量子资源

Table3 Quantum resources for the Simplified AES key expansion quantum circuit

文献	Qubits	NOT	CNOT	Toffoli
[15]	40	10	568	192
[10]	16	27	108	56
[16]	24	19	64	80
本文	16	27	92	56

3.3 整体算法的量子电路优化

最终实现的 S-AES 量子资源如表 4 所列。为了便于与文献 [16] 中的量子电路进行资源对比,本文对其量子电路中的三控制位 CCCNOT 门进行了分解处理。具体而言,加密算法中 S 盒量子电路的三控制位 CCCNOT 门被分解为 4 个 Toffoli 门,分解方法与第 3.1 节相同;而密钥扩展算法中的三控制位 CCCNOT 门的处理方法已在第 3.2 节中详细解释。本文设计的量子电路在 Qiskit Aer 量子模拟器中得到了验证,输入的明文为:plaintext=0110 1111 0110 1011(对应 ASCII 码中的“ok”);密钥为:key=1010 0111 0011 1011;最终得到的密文为:ciphertext=0000 0111 0011 1000。输出结果与 S-AES 设计者在文献 [14] 中所举的例子相同。与之前的最优方案相比,本文方法节省了 40 个 Toffoli 门和 8 个量子比特,优化了整体量子资源的使用。

表 4 用于 S-AES 整体量子电路的量子资源

Table 4 Quantum resources for the overall Simplified AES quantum circuit

文献	Qubits	NOT	CNOT	Toffoli
[15]	72	26	1 080	576
[10]	48	75	364	168
[16]	40	35	152	160
本文	32	51	220	120

4 仿真实验

本章主要描述仿真实验的环境配置与验证结果。实验环境部署于阿里云平台,选用配置为 256GB 内存的云服务器(见表 5),基于 Anaconda 搭建 Python 编程环境,并安装 Qiskit 量子计算框架。实验中构建的量子电路通过 Qiskit Aer 模拟器进行验证,具体实验代码如下。

```
用 DORCIS 实现的 S 盒量子电路代码:
def Sbox_D(q0,q1,q2,q3):
    qc.swap(q3,q0)
    qc.swap(q3,q2)
    qc.cx(q0,q3)
    qc.ccx(q1,q3,q2)
    qc.cx(q2,q3)
    qc.x(q0)
    qc.ccx(q3,q0,q1)
    qc.cx(q2,q0)
    qc.ccx(q2,q1,q3)
    # qc.cccx(q1[0],q1[1],q1[3],q1[2])
    qc.append(gate3,[q1,q0,q3,q2])
    qc.x(q3)
    qc.cx(q2,q0)
    qc.cx(q0,q1)
    qc.x(q0)
    qc.cx(q3,q2)
```

```
qc.cx(q1,q3)
qc.ccx(q2,q3,q0)
qc.cx(q0,q3)
qc.swap(q1,q0)
qc.swap(q3,q2)
    CCCNOT 门分解后得到的 S 盒量子电路代码:
def Sbox(q0,q1,q2,q3,q4):
    qc.swap(q3,q0)
    qc.swap(q3,q2)
    qc.cx(q0,q3)
    qc.ccx(q1,q3,q2)
    qc.cx(q2,q3)
    qc.x(q0)
    qc.ccx(q3,q0,q1)
    qc.cx(q2,q0)
    qc.ccx(q2,q1,q3)
    qc.ccx(q0,q1,q4)
    qc.ccx(q3,q4,q2)
    qc.ccx(q0,q1,q4)
    qc.ccx(q3,q4,q2)
    qc.x(q3)
    qc.cx(q2,q0)
    qc.cx(q0,q1)
    qc.x(q0)
    qc.cx(q3,q2)
    qc.cx(q1,q3)
    qc.ccx(q2,q3,q0)
    qc.cx(q0,q3)
    qc.swap(q1,q0)
    qc.swap(q3,q2)
    密钥扩展算法中 S 盒量子电路代码:
def Sbox_8(q0,q1,q2,q3,q4,q5,q6,q7):
    qc.cx(q4,q6)
    qc.cx(q7,q6)
    qc.cx(q5,q4)
    qc.ccx(q2,q3,q1)
    qc.ccx(q0,q1,q2)
    qc.cx(q2,q6)
    qc.ccx(q2,q3,q0)
    qc.cx(q0,q4)
    qc.ccx(q0,q1,q5)
    qc.ccx(q2,q0,q3)
    qc.cx(q5,q4)
    qc.ccx(q1,q3,q7)
    qc.ccx(q2,q0,q3)
    qc.cx(q7,q6)
    qc.ccx(q2,q3,q0)
    qc.cx(q4,q6)
    qc.ccx(q0,q1,q2)
    qc.ccx(q2,q3,q1)
    qc.cx(q0,q2)
    qc.x(q2)
    qc.ccx(q2,q3,q4)
    qc.x(q2)
    qc.x(q4)
    qc.cx(q0,q2)
```

```
qc.cx(q0,q1)
qc.ccx(q2,q1,q5)
qc.cx(q1,q5)
qc.cx(q0,q1)
qc.cx(q3,q5)
qc.cx(q2,q3)
qc.cx(q3,q7)
qc.ccx(q1,q3,q6)
qc.cx(q1,q3)
qc.x(q3)
qc.ccx(q0,q3,q7)
qc.x(q3)
qc.cx(q2,q3)
qc.cx(q1,q3)
qc.x(q7)

S-AES的量子电路代码:
def SAES(k,x):
    for i in range(16):
        qc.cx(k[i],x[i])
        Sbox(x[3],x[2],x[1],x[0],k[0])
        Sbox(x[7],x[6],x[5],x[4],k[1])
        Sbox(x[11],x[10],x[9],x[8],k[2])
        Sbox(x[15],x[14],x[13],x[12],k[3])
        MC(x[0],x[1],x[2],x[3],x[12],x[13],x[14],x[15])
        MC(x[8],x[9],x[10],x[11],x[4],x[5],x[6],x[7])
        Sbox_8(k[8],k[9],k[10],k[11],k[4],k[5],k[6],k[7])
        Sbox_8(k[12],k[13],k[14],k[15],k[0],k[1],k[2],k[3])
        qc.x(k[0])
    for i in range(8):
        qc.cx(k[i],k[i+8])
    for i in range(4):
        qc.cx(k[i],x[i])
        qc.cx(k[i+4],x[i+12])
        qc.cx(k[i+8],x[i+8])
        qc.cx(k[i+12],x[i+4])
    Sbox(x[3],x[2],x[1],x[0],k[0])
    Sbox(x[7],x[6],x[5],x[4],k[1])
    Sbox(x[11],x[10],x[9],x[8],k[2])
    Sbox(x[15],x[14],x[13],x[12],k[3])
    Sbox_8(k[8],k[9],k[10],k[11],k[4],k[5],k[6],k[7])
    Sbox_8(k[12],k[13],k[14],k[15],k[0],k[1],k[2],k[3])
    qc.x(k[2])
    qc.x(k[3])
    for i in range(8):
        qc.cx(k[i],k[i+8])
    for i in range(16):
        qc.cx(k[i],x[i])
```

表 5 实验环境配置

配置名称	配置参数
实例规格	64 核 256Gecs, g8aecs-7
系统盘	ESSD 云盘/dev/xvda40GB 模块属性性能级别 0
宽带	1 Mbps 按固定带宽
CPU	64
操作系统	Windows Server 2022 数据中心版 64 位 中文版 Windows64 位
内存	256 GB
地域	华北 2(北京)

在加密验证环节,分别采用以下两种组合进行测试:

1)使用密钥(1010 0111 0011 1011)对明文(0110 1111 0110 1011)执行量子电路加密,Aer 模拟器输出密文为(0000 0111 0011 1000);

2)使用密钥(0000 0000 0000 0000)对明文(0110 1111 0110 1011)执行量子电路加密,Aer 模拟器输出密文为(0001 1011 1001 1000)。

两组实验的密文结果均与理论预期值完全一致,验证了量子电路设计的正确性。通过 Qiskit 资源分析模块统计,本次量子电路实现共消耗 32 个量子比特、51 个 NOT 门、220 个 CNOT 门和 120 个 Toffoli 门。

结束语 本文对 S-AES 密码算法的量子电路进行了优化,减少了所需的量子资源。具体而言,基于 S 盒的查找表,采用 DORCIS 工具优化其量子电路,使资源消耗降低至仅需 3 个 NOT 门、8 个 CNOT 门、4 个 Toffoli 门和 1 个三控制位的 CCCNOT 门。在 S 盒优化的基础上,使用了一种无需辅助量子比特即可将 CCCNOT 门分解为 4 个 Toffoli 门的方法。随后,通过变量置换的方法,取消了加密算法中移位操作所需的交换门。此外,综合 S 盒的布尔表达式,设计并实现了密钥扩展中 8 量子比特的 S 盒量子电路,使输出直接异或到目标密钥的 4 位量子比特上,同时保持输入不变。该设计避免了多次执行 S 盒逆操作以恢复密钥值,从而优化了密钥扩展的资源消耗。优化后的密钥扩展中,S 盒量子电路所需的资源为 6 个 NOT 门、19 个 CNOT 门和 14 个 Toffoli 门。整体优化后的 S-AES 量子电路总共需要 32 个量子比特、51 个 NOT 门、220 个 CNOT 门和 120 个 Toffoli 门。与现有最优方案相比,该方法减少了 40 个 Toffoli 门和 8 个量子比特的消耗,降低了 S-AES 所需的量子资源和计算复杂度。

本文实现的简化分组密码量子电路,为在量子环境下执行密钥恢复攻击奠定了基础。这对于未来抗量子密码的发展,以及精确评估算法对量子攻击的抵抗能力,具有重要的研究意义。此外,在当前量子计算机环境下,研究在针对简化分组密码的量子密钥恢复攻击电路中引入噪声,并评估其对攻击效率和成功率的影响,也是未来的重要研究方向。

参 考 文 献

[1] ELGAMAL T. A public key cryptosystem and a signature scheme based on discrete logarithms[J]. IEEE transactions on information theory,1985,31(4):469-472.

[2] SHOR P W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer[J]. SIAM review,1999,41(2):303-332.

[3] SIMON D R. On the power of quantum computation[J]. SIAM journal on computing,1997,26(5):1474-1483.

[4] KAPLAN M,LEURENT G,LEVERRIER A,et al. Breaking symmetric cryptosystems using quantum period finding[C]//Advances in Cryptology-CRYPTO 2016:36th Annual International Cryptology Conference,Santa Barbara,CA,USA,August 14-18,2016,Proceedings, Part II 36. Springer Berlin Heidelberg,2016:207-237.

[5] GROVER L K. A fast quantum mechanical algorithm for database search[C]//Proceedings of the twenty-eighth annual ACM symposium on Theory of computing. 1996:212-219.

[6] CHUNG D,LEE S,CHOI D,et al. Alternative tower field construction for quantum implementation of the AES S-box[J]. IEEE Transactions on Computers,2021,71(10):2553-2564.

[7] LIU Q,PRENEEL B,ZHAO Z,et al. Improved quantum circuits for AES;Reducing the depth and the number of qubits[C]//International Conference on the Theory and Application of Cryptology and Information Security. Singapore: Springer Nature Singapore,2023:67-98.

[8] SHI H,FENG X. Quantum circuits of AES with a low-depth linear layer and a new structure[C]//International Conference on the Theory and Application of Cryptology and Information Security. Singapore:Springer Nature Singapore,2024:358-395.

[9] O'GORMAN J,CAMPBELL E T. Quantum computation with realistic magic-state factories [J]. Physical Review A, 2017, 95(3):032338.

[10] WANG Z G,WEI S J, LONG G L. A quantum circuit design of AES requiring fewer quantum qubits and gate operations[J]. Frontiers of Physics,2022,17(4):41501.

[11] SIMMONS S. Algebraic Cryptanalysis of Simplified AES * [J]. Cryptologia,2009,33(4):305-314.

[12] SAEED R,BHERY A. Cryptanalysis of Simplified-AES Using Intelligent Agent[C]// Hybrid Artificial Intelligent Systems: 10th International Conference, HAIS 2015, Bilbao, Spain, June 22-24,2015,Proceedings 10. Springer International Publishing, 2015:173-187.

[13] CAMPBELL S,GRINCHENKO M,SMITH W. Linear cryptanalysis of simplified AES under change of S-Box[J]. Cryptologia,2013,37(2):120-138.

[14] MUSA M A,SCHAEFER E F,WEDIG S. A simplified AES algorithm and its linear and differential cryptanalyses[J]. Cryptologia,2003,27(2):148-177.

[15] ALMAZROOIE M,ABDULLAH R,SAMSUDIN A,et al. Quantum grover attack on the simplified aes[C]// Proceedings of the 2018 7th International Conference on Software and Computer Applications. 2018:204-211.

[16] JANG K B,SONG G J,KIM H J,et al. Grover on simplified aes [C]//2021 IEEE International Conference on Consumer Electronics-Asia(ICCE-Asia). IEEE,2021:1-4.

[17] JEAN J,PEYRIN T,SIM S M,et al. Optimizing implementations of lightweight building blocks[J]. Cryptology ePrint Archive,2017,4:130-168.

[18] CHUN M,BAKSI A,CHATTOPADHYAY A. Dorcis;depth optimized quantum implementation of substitution boxes[J]. Cryptology ePrint Archive,2023,2(8):6-16.

[19] SHENDE V V,BULLOCK S S,MARKOV I L. Synthesis of quantum logic circuits[C]// Proceedings of the 2005 Asia and South Pacific Design Automation Conference. 2005:272-275.

[20] LUO Q,LI Q,LI X,et al. Quantum circuit implementations of SM4 block cipher optimizing the number of qubits[J]. Quantum Information Processing,2024,23(5):177.

[21] NIELSEN M A,CHUANG I L. Quantum computation and quantum information[M]. Cambridge University Press,2010.



DING Lang, born in 2001, postgraduate. His main research interests include quantum circuit and quantum attack.



LUO Qingbin, born in 1987, professor. His main research interests include quantum circuit and quantum cryptography.