

基于神经网络区分器的 SIMON-like 算法参数安全性评估

侯泽洲^{1,2}, 任炯炯^{1,2}, 陈少真^{1,2}

(1. 信息工程大学, 河南 郑州 450001; 2. 密码科学技术国家重点实验室, 北京 100878)

摘要: 神经网络区分器作为一种新的可被应用于密码算法安全性分析的工具, 一经提出便被应用于多种密码算法的安全性分析。对于 SIMON-like 算法, 其循环移位参数的选择有多种。利用神经网络区分器对分组长度为 32 bit 的 SIMON-like 算法的循环移位参数 (a, b, c) 的安全性进行了研究, 并给出了好的循环移位参数选择。利用 Kölbl 等在 CRYPTO2015 中提出的 SIMON-like 算法仿射等价类思想, 将分组长度为 32 bit 的 SIMON-like 算法的循环移位参数划分至 509 个等价类, 并选择其中使 $\gcd(a-b, 2)=1$ 成立的 240 个等价类进行研究。针对 240 个等价类的代表元构建了自动化搜索差分路径的 SAT/SMT 模型, 并利用 SAT/SMT 求解器搜索了不同代表元的多轮最佳差分路径。利用搜索得到的最佳差分路径的输入差分训练了神经网络区分器, 选择其中准确率最高的神经网络区分器作为代表元的神经网络区分器, 统计了不同代表元的神经网络区分器准确率。发现 Kölbl 等给出的 20 个最佳循环参数并不能使神经网络区分器的准确率最低, 而且其中 4 个循环移位参数对应的神经网络区分器的准确率超过了 80%, 这意味着这 4 个循环移位参数抗神经网络区分器的能力是差的。综合考虑 Kölbl 等的选择和不同代表元的神经网络区分器准确率, 给出了 3 个好的循环移位参数选择, 即 $(6, 11, 1)$ 、 $(1, 8, 3)$ 和 $(6, 7, 5)$ 。

关键词: SIMON-like; 仿射等价类; 神经网络区分器; 循环移位参数

中图分类号: TP393

文献标志码: A

DOI: 10.11959/j.issn.2096-109x.2023029

Security evaluation for parameters of SIMON-like cipher based on neural network distinguisher

HOU Zezhou^{1,2}, REN Jiongjiong^{1,2}, CHEN Shaozhen^{1,2}

1. Information Engineering University, Zhengzhou 450001, China

2. State Key Laboratory of Cryptology, Beijing 100878, China

Abstract: The neural distinguisher is a new tool widely used in crypto analysis of some ciphers. For SIMON-like block ciphers, there are multiple choices for their parameters, but the reasons for designer's selection remain unexplained. Using neural distinguishers, the security of the parameters (a, b, c) of the SIMON-like with a block

收稿日期: 2022-01-05; 修回日期: 2022-03-28

通信作者: 侯泽洲, zezhouhou1998@163.com

基金项目: 数学工程与先进计算国家重点实验室开放基金 (2019A08)

Foundation Item: State Key Laboratory of Mathematical Engineering and Advanced Computing Open Fund (2019A08)

引用格式: 侯泽洲, 任炯炯, 陈少真. 基于神经网络区分器的 SIMON-like 算法参数安全性评估[J]. 网络与信息安全学报, 2023, 9(2): 154-163.

Citation Format: HOU Z Z, REN J J, CHEN S Z. Security evaluation for parameters of SIMON-like cipher based on neural network distinguisher[J]. Chinese Journal of Network and Information Security, 2023, 9(2): 154-163.

size of 32 bits was researched, and good choices of parameters were given. Firstly, using the idea of affine equivalence class proposed by Kölbl et al. in CRYPTO2015, these parameters can be divided into 509 classes. And 240 classes which satisfied $\gcd(a-b, 2) = 1$ were mainly researched. Then a SAT/SMT model was built to help searching differential characteristics for each equivalent class. From these models, the optimal differential characteristics of SIMON-like was obtained. Using these input differences of optimal differential characteristics, the neural distinguishers were trained for the representative of each equivalence class, and the accuracy of the distinguishers was saved. It was found that 20 optimal parameters given by Kölbl et al. cannot make the neural distinguishers the lowest accuracy. On the contrary, there were 4 parameters, whose accuracy exceeds 80%. Furthermore, the 4 parameters were bad while facing neural distinguishers. Finally, comprehensively considering the choice of Kölbl et al. and the accuracy of different neural distinguishers, three good parameters, namely $(6, 11, 1)$, $(1, 8, 3)$, and $(6, 7, 5)$ were given.

Keywords: SIMON-like, affine equivalent class, neural network distinguisher, cyclic shift parameter

0 引言

随着物联网技术及芯片的小型化发展, 资源受限环境下的数据安全问题越来越受到重视。资源受限环境中存在计算能力不足、加解密效率低等问题, 轻量级分组密码算法应运而生, 并致力于在数据安全性、加解密效率、计算成本之间取得平衡。近年来, 研究人员设计提出的轻量级分组密码有 MIBS^[1]、PRESENT^[2]、CHAM^[3]、LBlock^[4]、SIMON^[5]等。

SIMON 算法是一种轻量级分组密码算法, 其结构简单, 加解密效率高, 一被提出, 便受到了密码学界的广泛关注。为此, 密码学者对 SIMON 算法的安全性进行了大量分析。同时, SIMON 算法在加密过程中使用循环移位参数完成运算, 从而引发了密码学者对 SIMON 算法循环移位参数选取的安全性分析。在 CRYPTO2015 中, Kölbl 等^[6]对 SIMON 算法的轮函数进行了详尽的分析, 给出了循环移位参数的等价类划分, 并研究了参数选取对 SIMON-like 算法的安全性的影响。Zhang 等^[7]探讨了循环移位参数选取对 SIMON-like 算法抗积分攻击能力的影响。Kondo 等^[8]对 Kölbl 等的方法进行了补充, 考虑了循环移位参数选取对 SIMON-like 算法抗积分攻击、不可能差分攻击、抗差分攻击和线性攻击的影响, 并发现在抗积分攻击和不可能差分攻击方面优于原始参数的循环移位参数, 在抗差分攻击和线性攻击方面则弱于原始参数。2018 年, 张毅等^[9]给出了不同于以往依赖计算机搜索循环移位参数等价类的计

算等价类公式, 从而优化了 SIMON-like 算法的安全性评估。

随着深度学习理论的推陈出新及高性能计算芯片的发展, 深度学习技术被广泛应用于计算机视觉^[10-11]、自然语言处理^[12-13]等领域。受深度学习在图像领域应用的启发, 密码学者将深度学习技术引入侧信道分析^[14-15]中, 大大推动了侧信道分析的发展。Gohr^[16]将深度学习技术引入密码分析之中, 通过神经网络与密文对数据构造了针对减轮 SPECK32/64 的神经网络区分器, 并将神经网络区分器应用于减轮 SPECK32/64 的密钥恢复攻击, 降低了减轮 SPECK32/64 的密钥恢复攻击的复杂度。Benamira 等^[17]对 Gohr 提出的神经网络区分器的原理进行了研究, 并通过实验表明, 神经网络区分器不仅依赖于密文对的差分分布, 也依赖于倒数第二轮和倒数第三轮的差分分布。基于这一实验结果, Benamira 等^[17]给出了训练 nr 轮神经网络区分器所需要密文对的输入差分的选择方式, 即选择 $nr-1$ 轮或 $nr-2$ 轮最佳路径的输入差分。

本文主要研究了循环移位参数选取对 SIMON-like 算法神经网络区分器准确率的影响。本文首先根据 Kölbl 等提出的仿射等价类的概念, 将分组长度为 32 bit 的 SIMON-like 算法的循环移位参数 (a, b, c) 分为 509 类。由于满足 $\gcd(a-b, 2) = 2$ 的等价类的差分传播概率较高, 因此本文主要研究使 $\gcd(a-b, 2) = 1$ 成立的 240 个等价类, 并从每一个等价类中选择一个代表元来代

表该等价类。同时，对于等价类的代表元，本文利用 Kölbl 等提出的计算差分传播概率的方法对 SIMON-like 算法建立自动化搜索差分路径的 SAT/SMT 模型。基于搜索差分路径的 SAT/SMT 模型，本文求取了各个代表元的 7 轮与 8 轮最佳差分路径。本文根据 Benamira 等提出的选择神经网络区分器输入差分的方法，选取已搜索得到的 7 轮与 8 轮最佳差分路径的输入差分作为 9 轮神经网络区分器的输入差分，并训练各个代表元的神经网络区分器。每一个代表元的最佳差分路径并不唯一，因此每一个代表元可得到多个神经网络区分器，本文选择其中准确率最高的神经网络区分器作为该代表元的神经网络区分器。通过对不同代表元的区分器的准确率进行对比，本文发现当考虑神经网络区分器的准确率时，准确率最低的循环移位参数与 Kölbl 等推荐的循环移位参数并不一致。甚至，Kölbl 等推荐的 20 个循环移位参数中的 4 个循环参数的神经网络区分器的准确率超过了 80%，也就是说其抗神经网络区分器的能力是差的。本文综合考虑 Kölbl 等的选择和神经网络区分器，给出了几个最佳的循环移位参数。

1 准备知识

本节首先介绍常用符号的定义，然后介绍 SIMON 算法和 SIMON 算法的仿射等价类，最后介绍神经网络区分器。

1.1 符号定义

本文的常用符号及其含义如表 1 所示。

表 1 符号及其含义
Table 1 Symbols and definitions

符号	含义
$\text{GF}(2) \{0, 1\}$	二元有限域
$\text{GF}(2)^n$	$\text{GF}(2)$ 上的 n 维向量空间
S^j	比特循环左移 j bit
$\oplus$	比特异或操作 XOR
$\odot$	比特与操作 AND
$\vee$	比特或操作 OR
$\bar{x}$	比特否操作 $\bar{x} = x \oplus 1$
$\text{SIMON-}(a, b, c)$	循环移位参数为 (a, b, c) 、分组长度为 32 bit 的 SIMON-like 算法
$\text{wt}(x)$	x 的汉明重量

1.2 SIMON 算法

SIMON 算法设计者在设计之初便设计了多个版本以供安全性需求及计算效率不同的资源受限设备使用。SIMON 算法的参数如表 2 所示。

表 2 SIMON 算法的参数
Table 2 SIMON algorithm parameters

分组长度/bit	密钥长度/bit	字节长度/bit	密钥字节数/bit	轮数
32	64	16	4	32
48	72	24	3	36
48	96	24	4	36
64	96	32	3	42
64	128	32	4	44
96	96	48	2	52
96	144	48	3	54
128	128	64	2	68
128	192	64	3	69
128	256	64	4	72

表 2 给出了多个版本 SIMON 算法的相关参数。分组长度为 $2n$ bit，密钥长度为 mn bit 的 SIMON 算法，通常用 $\text{SIMON } 2n / mn$ 代替。例如，分组长度为 48 bit，密钥长度为 96 bit 的 SIMON 算法通常用 $\text{SIMON}48/96$ 代表。同时，当不考虑密钥长度时，SIMON 算法通常简写为 $\text{SIMON } 2n$ 。本文主要研究分组长度为 32 bit、密钥长度为 64 bit 的 SIMON 算法。

设 $\text{GF}(2)$ 代表二元有限域， $\text{GF}(2)^{16}$ 代表 $\text{GF}(2)$ 上的 16 维向量空间，那么对于 $k_i \in \text{GF}(2)^{16}$ ， $\text{SIMON}32/64$ 的轮函数可以定义为 Feistel 映射 $R_{k_i}^{(a,b,c)}: \text{GF}(2)^{16} \cdot \text{GF}(2)^{16} \rightarrow \text{GF}(2)^{16} \cdot \text{GF}(2)^{16}$ ，即 $R_{k_i}^{(a,b,c)}(x_i, y_i) = (y_i \oplus F_{(a,b,c)}(x_i) \oplus k_i, x_i)$ ，其中 $F_{(a,b,c)}(x) = (S^a x \odot S^b x) \oplus S^c x$ ， k_i 是轮子密钥， (a, b, c) 是循环移位参数，SIMON 算法的设计者选择的循环移位参数为 (1, 8, 2)。SIMON 算法的加密流程如图 1 所示。由于本文主要考虑循环移位参数的影响，因此不详细介绍 SIMON 算法的密钥生成算法，其具体实现细节参考文献[5]。

1.3 SIMON 算法的仿射等价类

SIMON 算法设计者指定循环移位参数对为 (1, 8, 2)，但设计者并未解释该循环移位参数的选

取原因,这就引起了循环移位参数对 SIMON-like 算法安全性影响的研究。对于 SIMON-like 算法的循环移位参数的选择空间,当不考虑 a 、 b 和 c 三者之间的关系时,共有 n^3 种可能,其中 n 为 SIMON-like 算法的字节长度,而如此大的选择空间严重制约了密码学者对 SIMON-like 算法的安全性分析。SIMON-like 算法中,若循环移位参数对 (a,b,c) 中 $a=b$,则算法的轮函数将变成线性运算,此时算法的安全性较差,因此不妨假设 $a>b$ 。此时,对于分组长度为 $2n$ bit 的 SIMON-like 算法,循环移位参数选择空间的规模从 n^3 减少到 $C_n^2 \cdot n$,其中 C_n^2 为 a 和 b 组合的数量, n 为 c 选择的数量。为进一步减小循环移位参数的选择空间,Kölbl 等^[6]引入了仿射等价的概念。

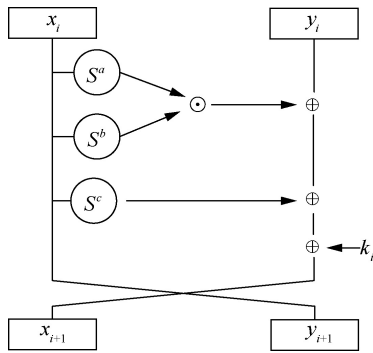


图 1 SIMON 算法的加密流程

Figure 1 Encryption process of SIMON algorithm

命题 1^[6] 令 T 为 n bit 字中的比特置换。那么存在 $s \in \mathbb{Z}_n$ 和 $t \in \mathbb{Z}_n$, 使得第 i bit 被映射到 $s \cdot i + t$ bit, 即

$$T(F_{(a,b,c)}(x)) = F_{(sa, sb, sc)}(T(x)) \quad (1)$$

通过命题 1, 可以发现将 SIMON32 的循环移位参数从 $(1,8,2)$ 变换为 $(3,8,6)$, 并对密钥生成算法进行相应的调整, 那么两个循环移位参数对应的密码算法加密得到的密文在考虑一个置换的情况下是等价的。因此对于循环移位参数 $(1,8,2)$ 和 $(3,8,6)$, 仅需要考虑其中一个即可。考虑更一般的情形, 若循环移位参数 (a_1, b_1, c_1) 可变换为 (a_2, b_2, c_2) , 则称 (a_1, b_1, c_1) 与 (a_2, b_2, c_2) 等价, 记为 $(a_1, b_1, c_1) \Leftrightarrow (a_2, b_2, c_2)$ 。进一步来说, 循环移位参数可通过命题 1 的等价性, 划分为若干个等价类, 并且只需要研究每一个等价类中的代表元即可。

对于分组长度为 32 bit 的 SIMON-like 算法, 共可以得到 509 个等价类, 也就是说循环移位参数的选择空间规模从 $C_{16}^2 \times 16 = 1920$ 个减少到 509 个。同时, 当 $\gcd(a-b, 2) \neq 1$ 时, $F_{(a,b,c)}$ 的差分传播概率将增加^[6]。因此, 对于分组长度为 32 bit 的 SIMON-like 算法, 值得研究的循环移位参数的等价类数量从 509 个降低到 240 个, 将这 240 个等价类的代表元的循环参数集合记作 ψ 。由于主要研究分组长度为 32 bit 的 SIMON-like 算法, 因此本文将循环移位参数为 (a,b,c) 、分组长度为 32 bit 的 SIMON-like 算法记作 SIMON- (a,b,c) , 其中 $(a,b,c) \in \psi$ 。

1.4 神经网络区分器

Gohr^[16]将密文对数据视作神经网络的训练数据, 利用残差神经网络(ResNets, residual neural network)^[18]训练生成了减轮 SPECK32/64 的神经网络区分器。给定一个差分 Δ 和一对明文 (P^0, P^1) , 使用随机密钥和目标密码算法对其进行加密得到 (C^0, C^1) 。通过明文对 (P^0, P^1) 和差分 Δ 的关系, 为该密文对设置对应的标签 $Y_{(C^0, C^1)}$ 。

$$Y_{(C^0, C^1)} = \begin{cases} 1, & P^0 \oplus P^1 = \Delta \\ 0, & \text{其他} \end{cases} \quad (2)$$

密文对 (C^0, C^1) 被称为一个样本, $Y_{(C^0, C^1)}$ 为该

样本对应的标签。同时, 每一对明文加密所使用的密钥均是随机生成的。在进行神经网络训练之前, 本文首先生成足够的带两种标签的样本集合, 以构造训练集和验证集, 而后选择合适的神经网络对其进行训练, 若训练完毕的神经网络的准确率超过 50%, 则该训练完毕的神经网络可被用于区分密文数据和伪随机数据。此时该训练完毕的神经网络被称为神经网络区分器, 而 Δ 被称为该神经网络区分器的输入差分。值得注意的是, 在训练集与验证集中, 标签为 0 与标签为 1 的样本数据应该各占一半, 以使神经网络可以对两类数据进行正确的学习。当从密码算法分析的角度考虑神经网络区分器的准确率时, 目标算法的神经网络区分器准确率越高, 说明目标算法的密文数据与伪随机数据区别越大, 进行区分攻击^[19]的效果越好, 进一步地进行密钥恢复攻击的效果也越好。相反, 当考虑目标密码算法的安全性时,

神经网络区分器的准确率越高, 目标算法的安全性越低。

Gohr 选择差分为 (0x40, 0x0) 作为神经网络区分器的输入差分, 并解释选择 (0x40, 0x0) 作为输入差分是因为 (0x40, 0x0) 能够以高概率传播到低汉明重量的差分。但是搜索得到这样的输入差分是困难的。因此, Benamira 等^[17]首先研究了 Gohr 的神经网络区分器的内在原理, 并对区分器训练所使用的两类数据集进行了研究, 发现神经网络区分器不仅依赖于密文对的差分分布, 也依赖于倒数第二轮和倒数第三轮的差分分布。基于此, 他们提出使用 $nr-1$ 轮或 $nr-2$ 轮的最佳路径的输入差分作为 nr 轮神经网络区分器的输入差分。通过 Benamira 等的结论, 本文使用 SIMON- (a, b, c) 的 7 轮与 8 轮最佳差分路径的输入差分训练 SIMON- (a, b, c) 的 9 轮神经网络区分器, 并对区分器的准确率进行比较, 以研究循环移位参数选取对神经网络区分器的影响。

$$P(\alpha \rightarrow \beta) = \begin{cases} 2^{-n+1}, & \alpha = 1, wt(\gamma) \equiv 0 \pmod{2} \\ 2^{-wt(\text{varibits} \oplus \text{doublebits})}, & \alpha \neq 1, \gamma \odot \overline{\text{varibits}} = 0, (\gamma \oplus S^{a-b}(\gamma)) \odot \text{doublebits} = 0 \\ 0, & \text{其他} \end{cases} \quad (3)$$

其中, $\text{varibits} = S^a(\alpha) \vee S^b(\alpha)$, $\text{doublebits} = S^b(\alpha) \odot \overline{S^a(\alpha)} \odot S^{2a-b}(\alpha)$, $\gamma = \beta \oplus S^c(\alpha)$ 。

2008 年, 微软工程师开发了 SAT/SMT 求解器 Z3-solver^[20], 该求解器不仅可以计算一些变量的取值是否满足约束条件, 也可通过给定变量的约束条件计算符合约束条件的变量的取值。基于 Z3-solver 的计算特性, 根据定理 1, 本文建立了自动化搜索 SIMON-like 差分路径的 SAT/SMT 模型, 并利用 Z3-solver 对 SIMON- (a, b, c) 的 7 轮与 8 轮最佳差分路径进行了搜索, 并保留这些最佳路径的输入差分, 分别记为 $\text{INPUT}_{(a,b,c)}^7$ 和 $\text{INPUT}_{(a,b,c)}^8$, 流程如算法 1 所示。

算法 1 获取 SIMON- (a, b, c) 的最佳差分路径输入差分

输入 Round
输出 $\text{INPUT}_{(a,b,c)}^{\text{Round}}$

2 循环移位参数安全性分析

本节根据 Kölbl 等提出的搜索 SIMON-like 算法最佳差分路径的方法, 搜索了不同等价类代表元的 7 轮与 8 轮最佳差分路径, 并将这些差分路径的输入差分作为神经网络区分器的输入差分训练 9 轮神经网络区分器, 选择其中准确率最高的神经网络区分器作为该代表元的神经网络区分器; 比较不同等价类代表元的神经网络区分器的准确率, 从准确率的角度评估循环移位参数对密码算法安全性的影响。

2.1 神经网络区分器生成

Kölbl 等给出了计算 SIMON- (a, b, c) 中 $F_{(a,b,c)}$ 差分传播概率的计算公式, 见定理 1。

定理 1^[6] 令 $F_{(a,b,c)}(x) = (S^a x \odot S^b x) \oplus S^c x$, 其中 $\gcd(a-b, n) = 1$, n 为偶数, 且 $a > b$ 。令 α 与 β 分别是 $F_{(a,b,c)}$ 的输入与输出差分, 则差分传播概率 $P(\alpha \rightarrow \beta)$ 为

- 1) $\text{INPUT}_{(a,b,c)}^{\text{Round}} \leftarrow \{ \}$;
- 2) $P \leftarrow 0$;
- 3) $X \leftarrow \{x_1, x_2, \dots, x_{\text{Round}+1}\}$;
- 4) $Y \leftarrow \{y_1, y_2, \dots, y_{\text{Round}+1}\}$;
- 5) $C_3 \leftarrow \{ \}$;
- 6) **for** $p \in [1, \dots, 32]$ **do**
- 7) $M = \text{Model}_{z_3}(X, Y, 2^{-p}, C_1, C_2)$;
- 8) **if** $M.\text{check}() = \text{sat}$ **then**
- 9) $P_{\max} = p$;
- 10) **break**;
- 11) **end if**
- 12) **end for**
- 13) $M = \text{Model}_{z_3}(X, Y, 2^{-P_{\max}}, C_1, C_2)$;
- 14) **while** $M.\text{check}() = \text{sat}$ **do**
- 15) $\bar{X}, \bar{Y} \leftarrow M.\text{result}()$;
- 16) $\text{INPUT}_{(a,b,c)}^{\text{Round}} \leftarrow \text{INPUT}_{(a,b,c)}^{\text{Round}} \cup \{(\bar{x}_1, \bar{y}_1)\}$;

17) $C_3 = C_3 \cup \{x_i \neq \bar{x}_i \mid i \in [1, \dots, \text{Round} + 1]\} \cup \{y_i \neq \bar{y}_i \mid i \in [1, \dots, \text{Round} + 1]\}$;
 18) $M = \text{Model}_{23}(X, Y, 2^{-P_{\max}}, C_1, C_2, C_3)$;
 19) **end while**
 20) **return** $\text{INPUT}_{(a,b,c)}^{\text{Round}}$

算法 1 输入差分路径的轮数 Round ，输出该轮数下最佳差分路径的输入差分的集合 $\text{INPUT}_{(a,b,c)}^{\text{Round}}$ 。在算法 1 的第 3)和 4)步中，设定搜索差分路径所需要的变量， X 代表差分路径中高位 16 bit 的集合， Y 代表低位 16 bit 的集合，其中 x_i 代表第 i 轮输入差分的高位 16 bit， y_i 代表第 i 轮输入差分的低位 16 bit。根据 SIMON-like 算法的轮函数，变量应符合 $y_{i+1} = x_i$ 和 $x_{i+1} = y_i \oplus F_{(a,b,c)}(x_i)$ ，其中 $i \in [1, \dots, \text{Round}]$ ，称此类约束条件为 C_1 类约束条件。当考虑差分路径的传播概率时，需要增加关于概率的约束，即 $2^{-p} = \prod_{i=1}^{\text{Round}} P(x_i \rightarrow y_i \oplus x_{i+1})$ ，称此类约束条件为 C_2 类约束条件。

算法 1 的第 6)~12)步以 C_1 类约束条件与 C_2 类约束条件建立 SAT/SMT 模型 M ，获得当前轮数下的最佳差分路径的传播概率 $P_{\max}$ 。

算法的第 13)~19)步通过迭代的方式获得当前轮数下的最佳差分路径输入差分集合。算法 1 的第 13)、18)步固定传播概率为 $2^{-P_{\max}}$ ，通过 Z3-solver 搜索使模型 M 成立的 X 和 Y 的取值，分别记为 $\bar{X} = \{\bar{x}_1, \bar{x}_2, \dots, \bar{x}_{\text{Round}+1}\}$ 和 $\bar{Y} = \{\bar{y}_1, \bar{y}_2, \dots, \bar{y}_{\text{Round}+1}\}$ 。此时的 $\bar{X}$ 和 $\bar{Y}$ 就是当前轮数下的最佳差分路径之一。算法 1 的第 16)步保存计算得到的最佳差分路径的输入差分。

为了求得更多的最佳差分路径，需要将已求得的 X 和 Y 的取值排除，即令 $\{x_i \neq \bar{x}_i \mid i \in [1, \dots, \text{Round} + 1]\}$ 和 $\{y_i \neq \bar{y}_i \mid i \in [1, \dots, \text{Round} + 1]\}$ ，称此类约束条件为 C_3 类约束条件。随着迭代的进行， C_3 的规模不断扩大。通过 C_1 、 C_2 和 C_3 类约束条件不断建立新的模型 M ，不断对 M 求解从而完成对当前轮数下最佳差分路径的输入差分的搜索。

令 $\text{INPUT}_{(a,b,c)} = \text{INPUT}_{(a,b,c)}^7 \cup \text{INPUT}_{(a,b,c)}^8$ ，使用 $\text{INPUT}_{(a,b,c)}$ 中的输入差分，本文训练

SIMON- (a,b,c) 的 9 轮神经网络区分器，并获取 SIMON- (a,b,c) 在不同输入差分下的神经网络区分器及其准确率，具体流程如算法 2 所示。

算法 2 获取 SIMON- (a,b,c) 的神经网络区分器及其准确率

输出 $\text{INPUT}_{(a,b,c)}$

输出 $L_{(a,b,c)}$

1) $L_{(a,b,c)} \leftarrow \{\}$;
 2) **for** $\text{ip} \in \text{INPUT}_{(a,b,c)}$ **do**
 3) $T = \text{Generate}(\text{ip}, 10^7)$;
 4) $V = \text{Generate}(\text{ip}, 10^6)$;
 5) $\text{Test} = \text{Generate}(\text{ip}, 10^6)$;
 6) $\text{Net} = \text{Train}(T, V)$;
 7) $\text{Acc}_{\text{Net}} = \text{Net}(\text{Test})$;
 8) $L_{(a,b,c)} \leftarrow L_{(a,b,c)} \parallel (\text{Net}, \text{Acc}_{\text{Net}})$;
 9) **end for**;
 10) **return** $L_{(a,b,c)}$

在算法 2 中，输入搜索得到的 SIMON- (a,b,c) 的 7 轮与 8 轮最佳路径的输入差分 $\text{INPUT}_{(a,b,c)}$ ，输出以这些输入差分构造的数据集训练的神经网络区分器及其准确率 $L_{(a,b,c)}$ 。算法 2 的第 3)步以 ip 为明文对的输入差分并使用 5×10^6 个随机密钥生成 5×10^6 个密文对，设定这些密文对的标签为 1。而对于标签为 0 的样本数据，首先随机生成 5×10^6 个 16 bit 的数，将这 5×10^6 个随机数作为明文对的差分并使用 5×10^6 个随机密钥生成 5×10^6 个密文对作为标签为 0 的样本数据。这两类样本数据共 10^7 个，共同作为训练神经网络的训练集并标记为 T 。

同样地，算法 2 的第 4)和 5)步生成了 10^6 个密文对作为验证集和测试集，分别标记为 V 和 Test 。算法 2 的第 6)步选用图 2 所示的神经网络结构对训练集 T 和验证集 V 进行训练，训练采用的优化器为 Adam，使用的损失函数为 MSE(mean squared error)，神经网络的参数如表 3 所示，计算环境如表 4 所示。算法 2 的第 7)步使用测试集 Test 计算第 6)步中已训练完毕的神经网络的准确率，并在第 8)步中将神经网络及其准确率保存在 $L_{(a,b,c)}$ 中。

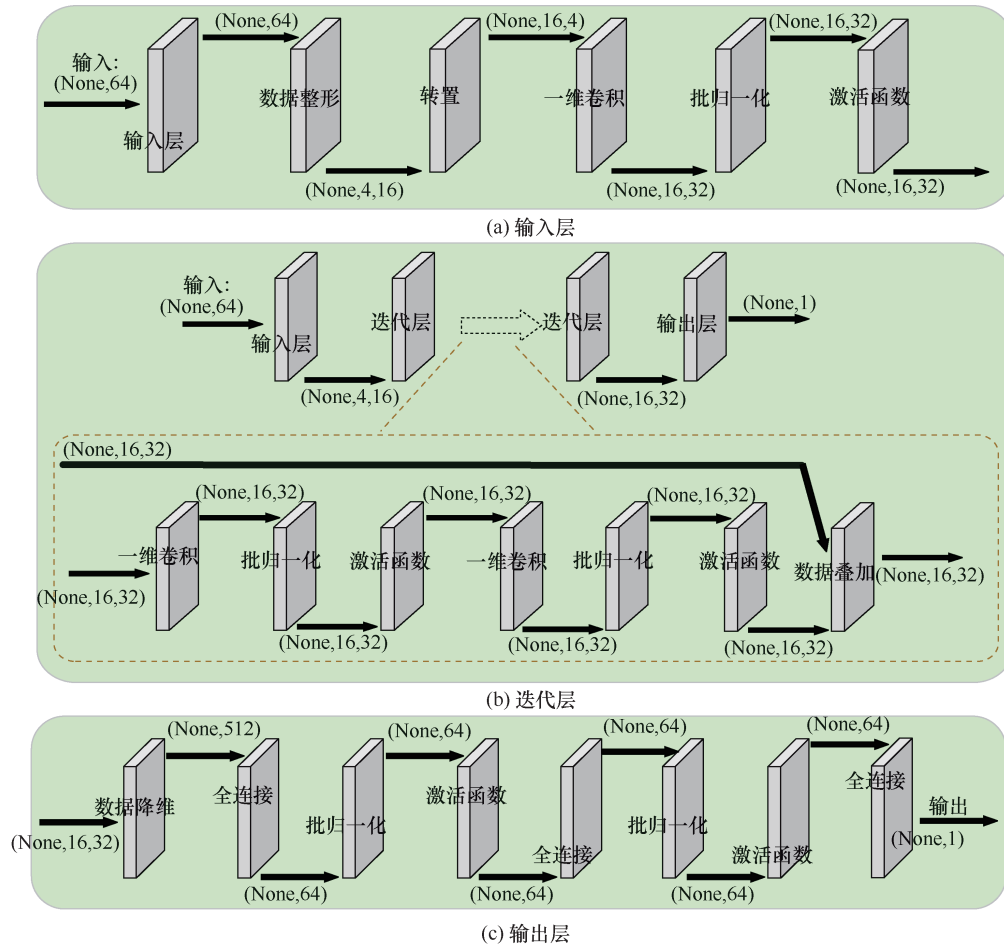


图 2 网络结构

Figure 2 Network architecture

表 3 神经网络的参数列表

Table 3 List of neural network parameters

参数	值
批尺寸	10^4
训练迭代次数	20
正则化参数	10^{-4}

表 4 神经网络的计算环境

Table 4 Computation environment of neural network

设备/软件	参数	数量
CPU	Intel(R) Xeon(R) Gold 6226R @2.90 GHz	2
GPU	NVIDIA GeForce RTX 3090(24 GB)	8
内容容量	512 GB	—
Tensorflow 版本	Tensorflow-gpu2.5.0	—
Keras 版本	Keras2.4.3	—

2.2 神经网络区分器准确率结果

在第 2.1 节中, 本文使用如表 4 所示的计算

环境耗时 2.5 个月计算了 240 个循环参数等价类代表元在 7 轮与 8 轮最佳路径输入差分作为神经网络区分器输入差分的情况下, 神经网络区分器的准确率。对于循环参数等价类的代表元 (a, b, c) , 本文选择 $L_{(a, b, c)}$ 中准确率最高的神经网络区分器作为该代表元的神经网络区分器, 记为 $\text{Net}_{(a, b, c)}$ 。同时对于代表元 (a, b, c) , 重新生成规模为 10^8 的新测试集, 并对神经网络区分器 $\text{Net}_{(a, b, c)}$ 进行测试, 记录其准确率, 将该准确率视为该代表元的准确率, 记为 $\text{Acc}_{(a, b, c)}$ 。

将准确率位于 0.5~0.6 的代表元划分至 A 类, 将准确率位于 0.6~0.7 的代表元划分为 B 类, 以此类推, 如表 5 所示。对于循环移位参数 (a, b, c) , 其神经网络区分器的准确率越高, 说明 $\text{SIMON-}(a, b, c)$ 的密文数据与伪随机数据的区别越大。因此当考虑神经网络区分器的准确率时,

表 5 等价类划分
Table 5 Equivalent class partitioning

类别	准确率 $Acc_{(a,b,c)}$	等价类代表元 (a,b,c)
A	[0.5, 0.6)	(10,15,3), (1,6,3), (3,12,7), (4,9,6), (1,12,3), (4,13,7), (5,12,6), (2,11,1), (5,14,1), (5,12,1), (1,12,5), (4,11,1), (7,12,3), (7,14,1), (7,12,6), (3,12,2), (3,14,5), (2,7,1), (4,13,1), (4,11,6), (4,15,3), (5,6,1), (1,12,2), (10,11,1), (11,12,1), (2,15,3), (6,7,1), (1,14,3), (14,15,9), (1,10,5), (4,7,2), (7,14,3), (7,10,1), (5,6,3), (12,13,1), (4,5,1), (2,3,5), (4,5,2), (1,8,2), (0,5,2), (6,11,1), (2,9,3), (6,15,5), (5,10,1), (8,15,9), (2,9,1), (0,7,1), (1,8,3), (10,13,1), (9,10,5), (4,9,3), (8,9,2), (8,15,3), (5,12,2), (2,13,1), (6,7,5), (2,13,5), (2,11,7), (4,9,2), (3,12,1), (9,14,1), (<u>0,11,1</u>), (9,10,1), (4,11,2), (13,14,5), (10,11,3), (1,10,4), (1,2,9), (6,9,1), (6,13,1), (7,12,2), (11,14,1), (3,6,1), (5,8,1), (8,11,1), (4,15,5), (0,5,1), (3,14,1), (14,15,3), (7,8,1), (2,5,1), (6,11,4), (2,9,4), (12,13,2), (12,15,2), (<u>1,2,3</u>), (1,14,5), (1,4,2), (6,13,2), (3,4,1), (<u>5,12,3</u>), (3,4,2), (3,10,2), (<u>7,12,1</u>), (1,10,2), (3,6,4), (6,11,2), (2,5,4), (6,9,2), (3,14,10), (6,7,2), (3,6,2), (3,14,4), (2,5,10), (1,6,2), (5,10,2), (7,10,2), (13,14,1), (10,11,2), (5,6,2)
B	[0.6, 0.7)	(4,7,1), (6,7,4), (10,13,2), (10,15,2), (<u>4,5,3</u>), (8,13,1), (1,2,4), (7,8,3), (8,15,2), (7,8,2), (0,3,2), (3,8,2), (0,13,1), (0,7,2), (8,11,2), (4,11,3), (9,12,7), (11,12,5), (5,14,7), (3,10,4)
C	[0.7, 0.8)	(4,15,1), (11,12,3), (8,9,4), (8,11,4), (1,8,4), (4,9,1), (12,15,1), (9,12,1), (6,9,4), (3,8,1), (8,9,3), (3,10,1), (3,8,4), (7,14,2), (1,8,9), (10,11,6), (0,3,4), (6,7,10), (1,6,4), (0,3,1), (10,15,6), (2,15,5), (2,7,4), (0,1,4), (5,14,2), (3,14,2), (2,3,1), (1,14,2), (14,15,2), (0,9,1), (8,9,1), (6,13,4), (2,9,8), (7,10,4), (1,10,0), (1,2,8), (5,6,4), (<u>8,13,2</u>), (7,14,9), (2,5,0), (<u>5,8,2</u>), (2,3,4)
D	[0.8, 0.9)	(1,4,8), (4,7,0), (4,5,8), (<u>0,1,2</u>), (3,4,0), (6,15,1), (11,14,5), (4,7,8), (3,4,8), (4,5,0), (14,15,1), (9,14,7), (1,4,0), (10,15,1), (13,14,3), (2,15,1), (2,7,8), (1,6,0), (5,6,8), (2,3,0)
E	[0.9, 1.0)	(1,6,8), (5,6,0), (2,3,8), (2,7,0), (1,6,6), (2,7,2), (5,6,6), (2,3,2), (2,7,7), (1,6,1), (1,14,1), (2,3,3), (2,9,2), (1,10,10), (4,7,7), (2,5,2), (1,12,1), (1,2,2), (1,4,1), (4,5,5), (2,9,9), (1,10,1), (2,5,5), (1,2,1), (3,12,4), (1,8,0), (1,12,4), (12,15,4), (12,13,4), (8,15,1), (8,9,0), (0,1,8), (7,8,9), (0,15,1), (1,10,8), (2,9,0), (2,5,8), (1,2,0), (4,7,4), (4,5,4), (1,4,4), (3,4,4), (1,8,1), (8,9,9), (0,1,1), (1,8,8), (8,9,8), (0,1,0)

可以认为归属于 A 类的等价类代表元的安全性较高，而归属于 E 类的等价类代表元的安全性较低。

表 5 罗列了 240 个代表元分类结果。从表 5 可以看到，不同循环移位参数的选取对神经网络区分器的准确率的影响是巨大的。在 240 个等价类代表元中，循环移位参数 (10,15,3) 的准确率最低约 52.114%；而对于准确率最高的循环移位参数 (0,1,0)，其准确率接近 100%。

2.3 基于神经网络区分器的循环移位参数安全性分析

在 CRYPTO2015 中，Kölbl 等^[6]给出了考虑差分路径传播概率的 20 个最佳的循环移位参数为：(0,1,2)、(0,1,3)、(0,5,10)、(0,5,15)、(0,7,14)、(0,13,7)、(0,13,10)、(1,2,3)、(1,8,3)、(1,12,7)、(3,4,5)、(3,8,14)、(4,5,3)、(5,10,15)、(5,12,3)、(6,7,5)、(6,11,1)、(7,8,5)、(7,12,1)、(8,13,2)。

计算这 20 个循环移位参数在本文中的等价类代表元，有 (0,1,2) $\Leftrightarrow$ (0,1,2) $\Leftrightarrow$ (0,5,10) $\Leftrightarrow$ (0,7,14) $\Leftrightarrow$ (0,13,10)，(0,11,1) $\Leftrightarrow$ (0,1,3) $\Leftrightarrow$ (0,5,15) $\Leftrightarrow$ (0,13,7)，(1,2,3) $\Leftrightarrow$ (1,2,3) $\Leftrightarrow$ (5,10,15)，(5,12,3) $\Leftrightarrow$ (3,4,5) $\Leftrightarrow$ (1,12,7) $\Leftrightarrow$ (5,12,3)，(4,5,3) $\Leftrightarrow$ (4,5,3)，(6,7,5) $\Leftrightarrow$ (6,7,5)，(1,8,3) $\Leftrightarrow$ (1,8,3) $\Leftrightarrow$ (7,8,5)，(5,8,2) $\Leftrightarrow$ (3,8,14)，(6,11,1) $\Leftrightarrow$ (6,11,1)，(7,12,1) $\Leftrightarrow$ (7,12,1)，(8,13,2) $\Leftrightarrow$ (8,13,2)。其中，下划线部分为本文选择的等价类代表元。Kölbl 等给出的 20 个最佳的循环移位参数的准确率如表 6 所示。

从表 6 可以看到，Kölbl 等给出的 20 个最佳的循环移位参数并不都属于 A 类，仍存在部分代表元属于安全性较差的 C 类甚至是 D 类。对于 SIMON 算法设计者选择的循环移位参数 (1,8,2)，其在神经网络区分器上的准确率与 Kölbl 等推

荐的(6,11,1)相近,但均不是最佳的循环移位参数选择。同时,对于SIMECK32/64^[21]算法,其轮函数与SIMON32/64算法的轮函数一致,仅循环移位参数选择不一样,当不考虑密钥扩展算法的影响时,循环移位参数(0,5,1)并不是最佳的选择。

表6 Kölbl的20个最佳循环参数划分
Table 6 Kölbl's 20 optimal parameters partitioning

循环参数	准确率	所属类别
(0,1,2)	0.816 2	D
(0,1,3)	0.560 9	A
(1,2,3)	0.577 9	A
(3,4,5)	0.582 9	A
(0,5,10)	0.816 2	D
(0,5,15)	0.560 9	A
(4,5,3)	0.607 0	B
(0,7,14)	0.816 2	D
(6,7,5)	0.559 4	A
(1,8,3)	0.554 5	A
(3,8,14)	0.797 78	C
(7,8,5)	0.554 5	A
(5,10,15)	0.577 9	A
(6,11,1)	0.547 0	A
(1,12,7)	0.582 9	A
(5,12,3)	0.582 9	A
(7,12,1)	0.587 4	A
(0,13,10)	0.816 2	D
(0,13,7)	0.560 9	A
(8,13,2)	0.795 1	C

这些结果反映了循环移位参数对差分路径和神经网络区分器的影响是不同的。Kölbl等考虑循环移位参数对最佳差分路径的影响,而对于神经网络区分器其不仅仅考虑了最佳路径,同时考虑了截断差分对区分器的影响^[17],这就导致了Kölbl等推荐的循环移位参数并不都能在神经网络区分器上有较低的准确率。

综合考虑抗差分攻击及抗神经网络区分器的

能力,SIMON-like算法应该选择循环移位参数(6,11,1)、(1,8,3)和(6,7,5)。

3 结束语

深度学习技术在密码分析领域的应用推动了算法安全性分析的多元发展。本文主要研究了循环移位参数的选取对SIMON-like算法的神经网络区分器的准确率的影响。首先通过将循环移位参数按照等价类划分,降低了需要研究的循环移位参数空间。而后为SIMON-(a,b,c)建立自动化搜索差分路径的模型,搜索了SIMON-(a,b,c)的最佳差分路径的输入差分,并利用搜索得到的输入差分训练了SIMON-(a,b,c)的神经网络区分器。对于目标算法的神经网络区分器,其准确率越高,目标算法的安全性越低,对攻击者越有利。当考虑神经网络区分器的准确率时,本文发现Kölbl等推荐的20个抗差分攻击能力强的循环移位参数并不能使神经网络区分器有较低的准确率。

在传统差分分析中,SIMON-like算法的安全性考量主要基于相同轮数的最佳路径的传播概率,传播概率越低说明选择该循环参数的安全性越高。而在深度学习技术引入密码算法的安全性分析后,发现抗差分攻击能力强的循环移位参数并不一定能保证神经网络区分器的准确率取值最低。影响神经网络区分器准确率的因素较多,特别是差分路径的聚合效应对神经网络区分器准确率的影响。因此,对于SIMON-like算法的安全性,不仅要考虑其抗传统攻击的能力,也要考虑其抗神经网络区分器区分攻击的能力。深度学习技术为密码算法的安全性分析增加了一条新的判别准则。

本文后续将进一步利用神经网络区分器探究循环移位参数对长分组的SIMON-like算法的安全性影响。同时,对于与SIMON算法一同被提出的SPECK算法,其轮函数也有循环移位参数,后续也将对循环移位参数选取对SPECK算法的安全性影响进行分析。

参考文献:

- [1] IZADI M, SADEGHIYAN B, SADEGHIAN S S, et al. MIBS: a new lightweight block cipher[C]//Proceedings of the Cryptology and Network Security. 2009: 334-348.

- [2] BOGDANOV A, KNUDSEN L R, LEANDER G, et al. Present: an ultra-lightweight block cipher[C]//Proceedings of the Cryptographic Hardware and Embedded Systems. 2007: 450-466.
- [3] BONWOOK K, DONGYOUNG R, HYEONJIN K, et al. CHAM: a family of lightweight block ciphers for resource-constrained devices[C]//Proceedings of the Information Security and Cryptology. 2017: 3-25.
- [4] WU W L, ZHANG L. LBlock: a lightweight block cipher[C]//Proceedings of the Applied Cryptography and Network Security. 2011: 327-344.
- [5] BEAUKIEU R, SHORS D, SMITH J, et al. The SIMON and SPECK lightweight block ciphers[C]//Proceedings of the 52nd Annual Design Automation Conference. 2015: 1-6.
- [6] KÖLBL S, LEANDER G, TIESSEN T. Observations on the SIMON block cipher family[C]//Proceedings of the Advances in Cryptology. 2015: 161-185.
- [7] ZHANG H L, WU W. Structural evaluation for SIMON-like designs against integral attack[C]//Proceedings of the Information Security Practice and Experience. 2016: 194-208.
- [8] KONDO K, YU S, IWATA T. On the design rationale of SIMON block cipher: Integral attacks and impossible differential attacks against SIMON variants[C]//Proceedings of the Applied Cryptography and Network Security. 2016: 518-536.
- [9] 张毅, 刘国强, 李超. 类 SIMON 算法参数等价类研究[J]. 密码学报, 2019, 6(3): 293-303.
ZHANG Y, LIU G Q, LI C. Research on equivalent class of SIMON-like cipher parameters[J]. Journal of Cryptologic Research, 2019, 6(3): 293-303.
- [10] ZEYNEP A, ANDREAS G, TORSTEN S. Computer vision and pattern recognition 2020[J]. International Journal of Computer Vision, 2021, 129(12).
- [11] FABRIZIO B, ALESSANDRO M. Computer vision meets image processing and UAS PhotoGrammetric data integration: from HBIM to the eXtended reality project of arco della pace in milan and its decorative complexity[J]. Journal of Imaging, 2021, 7(7): 118.
- [12] MATTIA V, MAURO G. Analysis of news sentiments using natural language processing and deep learning[J]. AI & Society, 2021, 36(3): 931-937.
- [13] LI J J, LIU X K, YIN W P, et al. Empirical evaluation of multi-task learning in deep neural networks for natural language processing[J]. Neural Computing and Applications, 2021, 33(9): 4417-4428.
- [14] CHEN D J, ZHAO Z H, QIN X, et al. MAGLeak: a learning-based side-channel attack for password recognition with multiple sensors in IIoT environment[J]. IEEE Transactions on Industrial Informatics, 2022, 18(1): 467-476.
- [15] ROBISSOUT R, BOSSUET L, HABRARD A, et al. Improving deep learning networks for profiled side-channel analysis using performance improvement techniques[J]. ACM Journal on Emerging Technologies in Computing Systems, 2021, 17(3): 1-30.
- [16] GOHR A. Improving attacks on round-reduced speck32/64 using deep learning[C]//Proceedings of the Advances in Cryptology. 2019: 150-179.
- [17] BENAMIRA A, GERAULT D, PEYRIN T, et al. A deeper look at machine learning-based cryptanalysis[C]//Proceedings of the Advances in Cryptology. 2021: 805-835.
- [18] HE K M, ZHANG X Y, REN S Q, et al. Deep residual learning for image recognition[C]//Proceedings of the Conference on Computer Vision and Pattern Recognition. 2016: 770-778.
- [19] YI C, YU H B. Bridging machine learning and cryptanalysis via EDLCT[J]. IACR Cryptology ePrint Archive, 2021, 705: 1-30.
- [20] LEONARDO M M, NIKOLAJ B. Z3: an efficient SMT solver[C]//Proceedings of the Tools and Algorithms for the Construction and Analysis of Systems. 2008: 337-340.
- [21] YANG G Q, ZHU B, SUDER V, et al. The simeck family of lightweight block ciphers[C]//Proceedings of the Cryptographic Hardware and Embedded Systems. 2015: 307-329.

[作者简介]



侯泽洲（1998—），男，山西朔州人，信息工程大学硕士生，主要研究方向为基于深度学习的轻量级分组密码安全性分析。



任炯炯（1995—），男，甘肃天水人，信息工程大学讲师，主要研究方向为分组密码的安全性分析。



陈少真（1967—），女，河南新乡人，信息工程大学教授、博士生导师，主要研究方向为密码算法的安全性分析。