ACE密码算法的积分分析

叶 涛[①②] 韦永壮*[①②] 李灵琛[②]

[①](桂林电子科技大学广西无线宽带通信与信号处理重点实验室 桂林 541004)

[②](桂林电子科技大学广西密码学与信息安全重点实验室 桂林 541004)

摘 要：ACE是国际轻量级密码算法标准化征集竞赛第2轮候选算法之一。该算法具有结构简洁，软硬件实现快、适用于资源受限环境等特点，其安全性备受业界广泛关注。该文引入字传播轨迹新概念，构建了一个传播轨迹的描述模型，并给出一个可以自动化评估分组密码算法抵抗积分攻击能力的方法。基于ACE算法结构特点，将该自动化搜索方法应用于评估ACE算法的安全性。结果表明：ACE置换存在12步的积分区分器，需要的数据复杂度为2^{256}，时间复杂度为2^{256}次12步的ACE置换运算，存储复杂度为8 Byte。相比于ACE算法设计者给出的积分区分器，该新区分器的步数提高了4步。

关键词：积分区分器；混合整数线性规划；自动化分析方法；ACE置换

中图分类号：TN918 **文献标识码**：A **文章编号**：1009-5896(2021)04-0908-07

DOI: 10.11999/JEIT200234

Integral Cryptanalysis of ACE Encryption Algorithm

YE Tao[①②] WEI Yongzhuang[①②] LI Lingchen[②]

[①](*Guangxi Key Laboratory of Wireless Wideband Communication and Signal Processing, Guilin University of Electronic Technology, Guilin* 541004, *China*)

[②](*Guangxi Key Laboratory of Cryptography and Information Security, Guilin University of Electronic Technology, Guilin* 541004, *China*)

Abstract: ACE as an authenticated encryption algorithm is selected as one of the round 2 candidates of the lightweight crypto standardization process. Since its excellent design advantages, e.g. simple structure, high performance in software and hardware, and suitable for constrained environments, the security of ACE is received extensive attention. In this paper, the concept of word propagation trail is introduced, and an exact model is constructed to describe the trail. A new automatic method for evaluating the security of word-based cipher against the integral attack is also proposed by using this model. Moreover, based on the structure of ACE, the security of ACE permutation is evaluated by using this new automatic method. More specifically, a new 12-step integral distinguisher of ACE permutation is verified by using this method, which requires the data complexity of about 2^{256} chosen data, the time complexity of about 2^{256} 12-step ACE permutation operations, and the memory complexity of about 8 Byte. Compared with the distinguishers given by ACE's designer, this new result prominently increases 4 steps indeed.

Key words: Integral distinguisher; Mixed Integer Linear Programming(MILP); Automatic cryptanalysis; ACE permutation

收稿日期：2020-04-03；改回日期：2020-01-03；网络出版：2021-02-26

*通信作者： 韦永壮 walker_wyz@guet.edu.cn

基金项目：国家自然科学基金(61872103)，广西重点研发计划(桂科AB18281019)，广西自然科学基金创新研究团队项目(2019GXNSFGA245004)，广西研究生教育创新计划(YCBZ2018051)，认知无线电与信息处理省部共建教育部重点实验室主任基金(CRKL180107)

Foundation Items: The National Natural Science Foundation of China(61872103), The Foundation of Guangxi Science and Technology Program (Guike AB18281019), The Innovation Research Team Project of Guangxi Natural Science Foundation(2019GXNSFGA245004), The Innovation Project of Guangxi Graduate Education(YCBZ2018051), The Foundation of Key Laboratory of Cognitive Radio and Information Processing, Ministry of Education (Guilin University of Electronic Technology)(CRKL180107)

1　引言

随着物联网技术和5G技术的快速发展，信息安全问题日趋突出，分组密码算法作为一种主流的加密体制，由于其安全性高、加密速度快、安全性易于评估等特点，具有非常重要的应用。对于一个安全的分组密码算法，其需要抵抗已知的各种攻击方法，比如线性密码分析[1]、差分密码分析[2]、不可能差分分析[3,4]、立方分析[5]、积分分析[6]等。目前除属性[7]以及自动化分析工具在密码分析中的广泛应用，使得积分分析成为近些年来国内外密码学者的研究热点之一。

积分分析最初是由Knudsen等人[6]为了分析Square的安全性而提出的分析方法，其基本思想是选取某些输入的明文字节为任意的固定值，其余的明文字节活跃，如果这个明文集合通过密码算法后，某些输出位的和恒为0，则攻击者构建了一个积分区分器。搜索积分区分器的典型方法一般可以分为如下3种：第1种方案是通过观察积分性质的传播轨迹来构建积分区分器[8,9]。第2种方案是通过评估密码算法输出表达式的代数次数来构建积分区分器[10,11]。第3种方法是利用除属性[7,12]构建出相应的模型，然后，利用优化工具来搜索积分区分器[13–17]。但是，这些构建积分区分器的方法受到密码结构或计算能力的限制。在2020年的国际快速软件加密会议(FSE2020)上，Zhang等人[18]给出了一种新的搜索算法。其主要思想是将内部状态的表达式用明文或密文字来表示，通过统计这些明文或密文字在内部状态中出现的次数，评估密码算法抵抗积分分析的能力。但是该方法需要手动推导出内部状态关于明文字或密文字的具体表达式形式，不能做到自动化分析。如何给出一种自动分析方法是目前的研究难点。

本文引入字传播轨迹新概念，基于混合整数线性规划技术(Mixed Integer Linear Programming, MILP)，构建了一个传播轨迹的描述模型，并给出一个新的积分区分器自动化搜索方法。利用这个自动化搜索算法，不需手动推导复杂的表达式，只需要判断模型是否有解就可以快速地得到明文(密文)字在内部状态的表达式中出现的次数，降低了密码分析者的工作量。利用这个新的自动化搜索工具，本文对国际轻量级密码算法标准化过程的第2轮候选算法之一ACE[19]的安全性进行了分析。结果表明：ACE密码算法存在12步的积分区分器，需要的数据复杂度为2^{256}，需要的时间复杂度为2^{256}次12步的ACE置换操作，存储复杂度为8 Byte。

本文后续章节的安排如下，第2节主要介绍一些基础知识；第3节主要介绍如何利用MILP结合文献[18]中的思想搜索积分区分器；第4节利用新的自动化分析方法对ACE置换的安全性进行分析；第5节是对全文的总结。文中用到的符号定义如表1所示。

2　基础知识

2.1　ACE置换

ACE认证加密算法是由加拿大滑铁卢大学Aagaard等人[19]设计的，它是国际轻量级密码算法标准化[20]过程的第2轮候选算法之一。其中，ACE的置换包含16步，每一步都对320 bit的数据进行操作，其基本结构基于广义Feistel结构[21]，并且每一步都使用不带密钥的减轮Simeck密码算法[22]作为非线性部件。由于ACE算法每一步的操作仅包含与、异或、循环移位运算等基本操作，其具有很好的软硬件实现性能。在设计ACE的原文档中，算法的设计者给出了ACE置换的8步的积分区

表 1　文中用到的符号

符号	定义
$\boldsymbol{X}\oplus\boldsymbol{Y}$	表示$\boldsymbol{X}$和$\boldsymbol{Y}$之间按位异或
$+/-$	十进制加/减
$\boldsymbol{X}\|\|\boldsymbol{Y}$	表示$\boldsymbol{X}$和$\boldsymbol{Y}$串联
$1^n/0^n$	表示n bit全1或全0的比特串
$\mathrm{SB}_j^i(\boldsymbol{X})$	表示ACE置换第i步的第j个密码S盒
$\|\boldsymbol{X}\|$	表示集合$\boldsymbol{X}$中元素的个数
$(\alpha,\boldsymbol{\beta})=\max(x[0],x[1],\cdots,x[n-1])$	计算数组$[x[0],x[1],\cdots,x[n-1]]$的最大值$\alpha$以及最大值对应的下标的集合$\boldsymbol{\beta}$，例如$\max(2,2,1,0)=(2,[0,1])$
$\alpha=\max'(x[0],x[1],\cdots,x[n-1])$	计算数组$[x[0],x[1],\cdots,x[n-1]]$中的最大值$\alpha$
$(\alpha,\boldsymbol{\beta})=\min(x[0],x[1],\cdots,x[n-1])$	计算数组$[x[0],x[1]\cdots,x[n-1]]$的最小值α以及最小值对应的下标的集合$\boldsymbol{\beta}$，例如$\min(2,2,1,0)=(0,[3])$
$\alpha=\min'(x[0],x[1],\cdots,x[n-1])$	计算数组$[x[0],x[1],\cdots,x[n-1]]$的最小值$\alpha$

分器，但是，是否存在步数更高的积分区分器还有待于进一步探究。

本文将ACE置换第i步的输入定义为$\boldsymbol{A}^{i-1}$, $\boldsymbol{B}^{i-1},\boldsymbol{C}^{i-1},\boldsymbol{D}^{i-1},\boldsymbol{E}^{i-1}$，第$i$步的输出状态定义为$\boldsymbol{A}^{i}$, $\boldsymbol{B}^{i},\boldsymbol{C}^{i},\boldsymbol{D}^{i},\boldsymbol{E}^{i}$，其中$1\le i\le 16$，$\boldsymbol{A}^{i-1},\boldsymbol{B}^{i-1},\boldsymbol{C}^{i-1}$, $\boldsymbol{D}^{i-1},\boldsymbol{E}^{i-1}$分别为64 bit的字。ACE置换的非线性操作由部件SB_j^i提供，其中，$1\le i\le 16$，$0\le j<3$。轮常数rc_j^i使用在部件SB_j^i中，步常数sc_j^i使用在ACE置换的每一步的操作中，轮常数和步常数的具体数值可以查阅文献[19]。ACE置换的具体结构如图1所示。

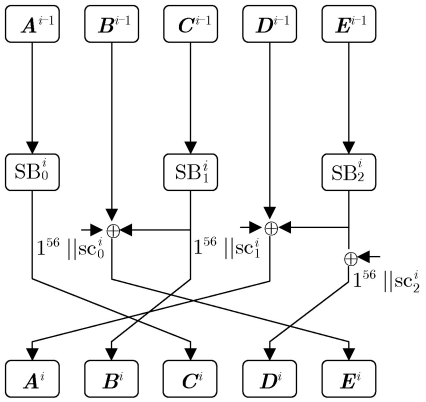

图 1 ACE置换

2.2 利用置换函数性质构建积分区分器

文献[18]给出了新的积分区分器的构建方法，主要用到了置换函数的性质。

性质1 如果函数$F(\boldsymbol{X}):F_2^b\to F_2^b$是关于$\boldsymbol{X}$的置换函数，$f(\boldsymbol{Y}):F_2^b\to F_2^b$为任意布尔函数，则对于$\boldsymbol{X},\boldsymbol{Y}\in F_2^b$，存在

$$\bigoplus_{\substack{\boldsymbol{X}\text{遍历}F_2^b\\ \boldsymbol{Y}\in F_2^b}} F(\boldsymbol{X}\oplus f(\boldsymbol{Y}))=0^b \tag{1}$$

文献[18]构建积分区分器的主要步骤如下：

(1) 利用轮函数的结构，对明文进行加密运算，并观察输出状态的表达式，直到第j个明文字达到了全扩散，并且至少存在1个输出状态的表达式中第j个明文字只出现1次，此时记录加密的轮数为r_{enc}^j。

(2) 利用轮函数的结构，对密文进行解密运算，观察输出状态的表达式，直到第j个密文字达到了全扩散，记录此时对应的解密轮数为r_{dec}^j，并且，当解密r_{dec}^j-1轮后，至少存在1个输出状态的表达式中不出现第j个密文字，记录下此时对应的r_{dec}^j。

(3) 遍历$0\le j<m$，其中m为轮函数的输入数据包含的字的数目，得到对应的r_{enc}^j和r_{dec}^j。($r_{\mathrm{enc}}+r_{\mathrm{dec}},\boldsymbol{K})=\max(r_{\mathrm{rec}}^0+r_{\mathrm{dec}}^0,r_{\mathrm{rec}}^1+r_{\mathrm{dec}}^1,\cdots,r_{\mathrm{rec}}^{m-1}+r_{\mathrm{dec}}^{m-1})$其中$r_{\mathrm{enc}}=r_{\mathrm{enc}}^{\boldsymbol{K}[0]},r_{\mathrm{dec}}=r_{\mathrm{dec}}^{\boldsymbol{K}[0]}$。

则利用性质2，可以评估密码算法抵抗积分分析的能力。

性质2 对于一个SPN结构的分组密码算法，令r_{enc}和r_{dec}分别为加密和解密方向达到全扩散的最小轮数，则这个密码算法存在$r_{\mathrm{enc}}+r_{\mathrm{dec}}$轮的积分区分器[18]。

3 新的积分区分器自动化搜索算法

3.1 分组密码算法字传播轨迹

为了观察每一个输入字在内部状态中出现的次数，攻击者需要利用轮函数的具体形式来推导出多轮后输出状态关于输入字的表达式形式，然后再观察每一个输入字在输出状态中出现的次数。以ACE密码算法的第0个字$\boldsymbol{A}^0$为例，本文定义数组$\boldsymbol{X}_0^i=[x_0^i[0],x_0^i[1],\cdots,x_0^i[4]]$为第$i$步的输出状态的表达式$\boldsymbol{A}^i,\boldsymbol{B}^i,\boldsymbol{C}^i,\boldsymbol{D}^i,\boldsymbol{E}^i$中包含第0个字$\boldsymbol{A}^0$的次数。首先，在不考虑步常数的条件下，可以得到ACE置换第i步的输入与输出之间的关系为

$$\left.\begin{aligned}\boldsymbol{A}^i&=\boldsymbol{D}^{i-1}\oplus\mathrm{SB}_2^i(\boldsymbol{E}^{i-1})\\ \boldsymbol{B}^i&=\mathrm{SB}_1^i(\boldsymbol{C}^{i-1})\\ \boldsymbol{C}^i&=\mathrm{SB}_0^i(\boldsymbol{A}^{i-1})\\ \boldsymbol{D}^i&=\mathrm{SB}_2^i(\boldsymbol{E}^{i-1})\\ \boldsymbol{E}^i&=\boldsymbol{B}^{i-1}\oplus\mathrm{SB}_1^i(\boldsymbol{C}^{i-1})\end{aligned}\right\} \tag{2}$$

利用式(2)，当只有$\boldsymbol{A}^0$为活跃块时，在加密方向上，可以得到$\boldsymbol{A}^0$的传播轨迹为

$$\begin{aligned}&\boldsymbol{X}_0^0=[1,0,0,0,0]\overset{\text{第}1\text{步}}{\to}\boldsymbol{X}_0^1=[0,0,1,0,0]\overset{\text{第}2\text{步}}{\to};\\ &\boldsymbol{X}_0^2=[0,1,0,0,1]\overset{\text{第}3\text{步}}{\to}\boldsymbol{X}_0^3=[1,0,0,1,1]\overset{\text{第}4\text{步}}{\to};\\ &\boldsymbol{X}_0^4=[2,0,1,1,0]\cdots\overset{\text{第}7\text{步}}{\to}\boldsymbol{X}_0^7=[4,1,1,3,3]\overset{\text{第}8\text{步}}{\to};\\ &\boldsymbol{X}_0^8=[6,1,4,3,2]\overset{\text{第}9\text{步}}{\to}\boldsymbol{X}_0^9=[5,4,6,2,5].\end{aligned}$$

由于$x_0^9[j]>1(0\le j<5)$，则$\boldsymbol{A}^9,\boldsymbol{B}^9,\boldsymbol{C}^9,\boldsymbol{D}^9,\boldsymbol{E}^9$都不是关于输入字$\boldsymbol{A}^0$的置换；由于$x_0^8[1]=1$，所以，$\boldsymbol{B}^8$是输入块$\boldsymbol{A}^0$的置换，则$r_{\mathrm{enc}}^0=8$($\boldsymbol{A}$为步函数输入的第0个字)。

由上面的实例可以发现，对于一个字，经过轮函数后，在每一轮的输出状态中出现的次数是固定的。根据这个特点，本文定义字传播轨迹如下。

定义1 字传播轨迹：定义迭代分组密码系统的轮函数为f_{r}，初始输入状态为$\boldsymbol{S}^0=[s^0[0],s^0[1],\cdots,s^0[m-1]]\in(F_2^n)^m$，第$i$轮函数输入状态为$\boldsymbol{S}^{i-1}=[s^{i-1}[0],s^{i-1}[1],\cdots,s^{i-1}[m-1]]\in(F_2^n)^m$，输出状态为$\boldsymbol{S}^i\in(F_2^n)^m$。$\boldsymbol{X}_j^i=[x_j^i[0],x_j^i[1],\cdots,x_j^i[m-1]]$为$i$轮后$\boldsymbol{S}^i$中每一个元素的表达式中包含第$j$个字

$s^0[j]$的个数，其中，向量$\boldsymbol{X}_j^i$中每一个元素都为大于等于0的正整数，则存在如下的传播轨迹

$$\boldsymbol{X}_j^0 \xrightarrow{f_{\mathrm{r}}} \boldsymbol{X}_j^1 \xrightarrow{f_{\mathrm{r}}} \boldsymbol{X}_j^2 \xrightarrow{f_{\mathrm{r}}} \boldsymbol{X}_j^3 \xrightarrow{f_{\mathrm{r}}} \cdots \tag{3}$$

本文定义$(\boldsymbol{X}_j^0, \boldsymbol{X}_j^1, \cdots, \boldsymbol{X}_j^r)$为第$j$个字的传播轨迹。同样，解密方向具有同样的性质。

3.2 构建分组密码算法字传播模型

如果直接手动推导字传播轨迹，需要的时间较长，并且容易发生错误。为了提高密码分析者的效率，本文给出了字传播轨迹的混合整数规划模型(MILP)，利用这个模型结合优化工具(Gurobi8.0)给出了一个自动化搜索积分区分器的方法。

本文的方法主要是在不考虑S盒内部结构的前提下，根据密码算法线性部件的特点来构建字传播模型。在密码算法的线性部件中，基本的操作包括置换运算、异或运算，下面分别讨论如何构建这两个运算的字传播轨迹。

性质3　置换运算的字传播轨迹模型：由于置换运算只是对某些字进行位置上的置换，不影响输入字在输出状态中出现的次数。如果轮函数中存在置换操作$s^i[k] \to s^i[k_1]$，则可以构建出如下的模型来描述第j个字的传播轨迹

$$x_j^i[k] - x_j^i[k_1] = 0 \tag{4}$$

性质4　异或运算的字传播轨迹模型：由于在分组密码算法的轮函数中，线性操作之前一般都会经过非线性层，所以，进行异或操作的变量都是非线性部件的输出，则经过异或操作后的输出中某一个字出现的次数为异或操作前这个变量在对应的字中出现的次数之和。如果轮函数的线性层中包含异或操作$s^i[k_1] \oplus s^i[k_2] \oplus s^i[k_3] \oplus \cdots \oplus s^i[k_l] = s^{i+1}[k]$，则可以构建如式(5)的模型来描述第$j$个字的传播轨迹

$$x_j^i[k_1] + x_j^i[k_2] + \cdots + x_j^i[k_l] = x_j^{i+1}[k] \tag{5}$$

通过将密码算法的线性层拆分为异或和置换操作，可以构建出这个密码算法每一轮的字传播模型，将其迭代r轮后可以得到r轮字传播模型M。

3.3 新积分区分器自动化搜索算法

基于文献[18]中的积分区分器构建方法，结合本文构建的MILP模型，同时利用优化求解工具，本文设计了积分区分器自动化搜索算法。

首先，给出如何设置模型的初始的输入。如果想得到输入的第k个字的传播轨迹，由于$s^0[k]$只在$s^0[k]$中出现1次，在其余的位置没有出现，则初始输入对应的模型变量$\boldsymbol{X}_k^0$为

$$\left.\begin{array}{l} x_k^0[k] = 1 \\ x_k^0[j] = 0, j \neq k \end{array}\right\} \tag{6}$$

密码算法加密r轮后，如果至少存在$x_k^r[j] = 1(j \in [0, m))$，则可以证明加密$r$轮后，$s^r[j]$为$s^0[k]$的置换，则存在式(7)的关系

$$\left.\begin{array}{ll} \overset{\oplus}{s^0[k]\text{遍历}F_2^n} & s^r[j] = 0 \\ s^0[j] \in F_2^n, j \neq k & \end{array}\right\} \tag{7}$$

所以，利用字传播轨迹，如果向量$\boldsymbol{X}_k^r$中至少存在1个元素等于1，则存在r轮的关于第k个字的积分区分器。则有如下的性质。

性质5　定义$\boldsymbol{X}_k^r = [x_k^r[0], x_k^r[1], \cdots, x_k^r[m-1]]$为加密$r$轮后第$k$个输入字在输出状态中出现的次数，如果$\min'(\boldsymbol{X}_k^r) > 1$，则加密方向不存在关于第$k$个字的$r$轮积分区分器。

利用性质5，如果加密$r+1$轮后，第1次出现$\min'(\boldsymbol{X}^{r+1}) > 1$，则存在$r$轮的积分区分器，此时令$r_{\mathrm{e}}^k = r$。

为了向解密方向继续扩展积分区分器的轮数，需要判断出第k个字解密方向上没有达到全扩散的最大轮数r_{d}^k，即解密$r_{\mathrm{d}}^k + 1$轮后第k个输入字第1次达到了全扩散，但是解密r_{d}^k轮后第k个字没有达到全扩散。根据这个关系，有如下的性质。

性质6　定义$\boldsymbol{Y}_k^r = [y_k^r[0], y_k^r[1], \cdots, y_k^r[m-1]]$为解密$r$轮后第$k$个字在输出状态中出现的次数，如果$\min'(\boldsymbol{Y}_k^r) > 0$，则第$k$个字解密$r$轮后达到全扩散。

利用性质6，如果解密$r+1$轮后，第1次出现$\min'(\boldsymbol{Y}_k^{r+1}) > 0$，则可以证明解密方向上第$k$个字没有达到全扩散的最大轮数为$r$，此时令$r_{\mathrm{d}}^k = r$。

为了快速地计算出r_{e}^k和r_{d}^k，本文分别设计了算法1和算法2。本文的方法使用了二分法进行搜索，这样减少了需要搜索的次数，同时使算法更易于实现。以算法1为例，如表2，首先令r_{h}和r_{l}分别表示r_{e}^k的上界和下界，对于一个轮函数为f_{r}，加密轮数为R的分组密码算法，令$r = \lfloor (r_{\mathrm{h}} + r_{\mathrm{l}})/2 \rfloor = R/2$，然后利用性质3和性质4结合轮函数$f_{\mathrm{r}}$的结构构建出$r$轮加密的字传播轨迹，同时根据性质5，添加了第(7)行的限制条件到模型中，然后利用求解器进行求解。如果模型有解，则r_{e}^k的下界为$\lfloor R/2 \rfloor$，再令$r_{\mathrm{l}} = r$，如果模型无解，则r_{e}^k的上界为$\lfloor R/2 \rfloor$，再令$r_{\mathrm{h}} = r$。这样的过程仅需迭代$\lceil \log_2 R \rceil$次，r_{e}^k的值就可以被确定。再利用求解器得到模型变量$(x_k^{r_{\mathrm{e}}^k}[0], x_k^{r_{\mathrm{e}}^k}[1], \cdots, x_k^{r_{\mathrm{e}}^k}[m-1])$中对应的值为1的下标集合$\boldsymbol{\omega}_k$。算法2的原理与算法1几乎相同，如表3，不同的只是第(5)行构建的模型和第(7)行添加的限制条件。

表 2 算法1：利用字传播模型确定r_{e}^k

输入：分组密码轮函数$f_{\mathrm{r}} \in (F_2^n)^m$，加密轮数$R$，活跃字的下标$k$

输出：r_{e}^k，以及加密r_{e}^k轮后，当第k个字活跃时，输出状态中的平衡字的下标集合$\boldsymbol{\omega}_k$

(1) $r_{\mathrm{h}} = R$, $r_{\mathrm{l}} = 0$, $r_{\mathrm{e}}^k = 0$, $r = 0$, flag= 0

(2) $x_k^i[0], x_k^i[1], ..., x_k^i[m-1]$为$i$轮加密后，输出状态对应的MILP模型变量，每一个MILP变量的值范围是大于等于0的整数

(3) While $r_{\mathrm{h}} - r_{\mathrm{l}} > 1$ do

(4) $r = \lfloor (r_{\mathrm{h}} + r_{\mathrm{l}})/2 \rfloor$

(5) 利用性质3和性质4构建出r轮的字传播模型M_{e}

(6) $M_{\mathrm{e}}.\mathrm{con} \leftarrow x_k^0[k] = 1, M.\mathrm{con} \leftarrow x_k^0[j] = 0, j \in [0,m), j \neq k$

(7) $M_{\mathrm{e}}.\mathrm{con} \leftarrow \min'\{(x_k^r[0], x_k^r[1], \cdots, x_k^r[m-1])\} = 1$

(8) 利用求解器对模型M_{e}进行求解

(9) If M_{e} 有解

(10) $r_{\mathrm{l}} = r, \mathrm{flag} = 1$

(11) else

(12) $r_{\mathrm{h}} = r, \mathrm{flag} = 0$

(13) End If

(14) End While

(15) If flag == 1

(16) $r_{\mathrm{e}}^k = r$

(17) else

(18) $r_{\mathrm{e}}^k = r - 1$

(19)End If

(20)$(1, \boldsymbol{\omega}_k) = \min\{(x_k^{r_{\mathrm{e}}^k}[0], x_k^{r_{\mathrm{e}}^k}[1], \cdots, x_k^{r_{\mathrm{e}}^k}[m-1])\}$

(21) return r_{e}^k和$\boldsymbol{\omega}_k$

表 3 算法2：利用字传播模型确定r_{d}^k

输入：分组密码解密轮函数$f_{\mathrm{r}}{}^{-1} \in (F_2^n)^m$，解密轮数$R$，活跃字的下标$k$

输出：r_{d}^k，以及解密r_{d}^k轮后，输出状态中的不包含第k个字的下标集合$\boldsymbol{\varphi}_k$

(1) $r_{\mathrm{h}} = R$, $r_{\mathrm{l}} = 0$, $r_{\mathrm{d}}^k = 0$, $r = 0$, flag $= 0$

(2) $y_k^i[0], y_k^i[1], ..., y_k^i[m-1]$为第$i$轮解密输出状态对应的MILP模型变量，每一个MILP变量的值范围是大于等于0的整数

(3) While $r_{\mathrm{h}} - r_{\mathrm{l}} > 1$ do

(4) $r = \lfloor (r_{\mathrm{h}} + r_{\mathrm{l}})/2 \rfloor$

(5) 利用性质3和性质4构建出r轮的字解密传播模型M_{d}

(6) $M_{\mathrm{d}}.\mathrm{con} \leftarrow y_k^0[k] = 1, M.\mathrm{con} \leftarrow y_k^0[j] = 0, j \in [0,m), j \neq k$

(7) $M_{\mathrm{d}}.\mathrm{con} \leftarrow \min'\{(y_k^r[0], y_k^r[1], \cdots, y_k^r[m-1])\} = 0$

(8) 利用求解器对模型M_{d}进行求解

(9) If M_{d} 有解

(10) $r_{\mathrm{l}} = r, \mathrm{flag} = 1$

(11) else

(12) $r_{\mathrm{h}} = r, \mathrm{flag} = 0$

(13) End If

(14) End While

(15) If flag == 1

(16) $r_{\mathrm{d}}^k = r$

(17) else

(18) $r_{\mathrm{d}}^k = r - 1$

(19) End If

(20) $(0, \boldsymbol{\varphi}_k) = \min\{(y_k^{r_{\mathrm{d}}^k}[0], y_k^{r_{\mathrm{d}}^k}[1], \cdots, y_k^{r_{\mathrm{d}}^k}[m-1])\}$

(21) return r_{d}^k和$\boldsymbol{\varphi}_k$

定义密码算法的线性层可以使用矩阵$\boldsymbol{A} = (a_{i,j}) \subset F_{2^n}^{m \times m}$来表示，非线性层使用$\mathrm{SB}_j^i$表示第$i$轮中的第$j$个密码S盒，则在不考虑轮常数和轮密钥的条件下，可得第i轮的输出状态与输入状态的关系为

$$s^i[k'] = \bigoplus_{l=0}^{m-1} a_{k',l} \times \mathrm{SB}_l^i(s^{i-1}[l]) \tag{8}$$

利用算法1，可以得到r_{e}^k和$\boldsymbol{\omega}_k$，如果$l \notin \boldsymbol{L}$，$a_{k',l}$的值为0，同时，$\boldsymbol{L} \subset \boldsymbol{\omega}_k$，利用置换函数的性质可以证明$s^{r_{\mathrm{e}}^k+1}[k']$仍然为$s^0[k]$的置换函数，因此，在加密方向上，$r_{\mathrm{e}}^k$轮的积分区分器可以继续向下扩展1轮，即如果输入的明文中第k个字活跃，可以得到$r_{\mathrm{e}}^k + 1$轮的积分区分器。然后，再根据算法2，继续在解密方向上扩展积分区分器的轮数。遍历$k \in [0,m)$，最终可以构建出$\max'(r_{\mathrm{e}}^0 + r_{\mathrm{d}}^0, r_{\mathrm{e}}^1 + r_{\mathrm{d}}^1, \cdots, r_{\mathrm{e}}^{m-1} + r_{\mathrm{d}}^{m-1}) + 1$轮的积分区分器。

4 ACE密码算法新积分分析

对于一个给定的迭代分组密码算法，本文给出了如下的积分区分器搜索步骤：

(1)先构建出密码算法轮函数的加密和解密方向的字传播模型；

(2)然后利用算法1和算法2分别确定r_{e}^k, $\boldsymbol{\omega}_k$, r_{d}^k和$\boldsymbol{\varphi}_k$, $k \in [0,m)$；

(3)计算$[r_{\mathrm{ed}}, \phi] = \max(r_{\mathrm{e}}^0 + r_{\mathrm{d}}^0, r_{\mathrm{e}}^1 + r_{\mathrm{d}}^1, \cdots, r_{\mathrm{e}}^{m-1} + r_{\mathrm{d}}^{m-1})$；

(4)根据算法结构，观察r_{ed}轮的积分区分器能否继续向加密方向扩展1轮。

利用上面的步骤对ACE置换进行积分分析。首先，在加密方向上，定义初始输入$(\boldsymbol{A}^0, \boldsymbol{B}^0, \boldsymbol{C}^0, \boldsymbol{D}^0, \boldsymbol{E}^0)$对应的模型变量为$(x_k^0[0], x_k^0[1], \cdots, x_k^0[4])$, $k \in [0,5)$，其中，k表示在初始输入中第k个字是活跃的，ACE置换加密方向第i步的输出对应的模型变量为$(x_k^i[0], x_k^i[1], \cdots, x_k^i[4])$，例如，$x_0^3[0]$表示加密

3步后，字$\boldsymbol{A}^0$在字$\boldsymbol{A}^3$的表达式中出现的次数，其他的变量代表的含义同理。则根据ACE置换的步函数的结构，可以得到加密方向第i步的步函数对应的字传播模型为

$$\left.\begin{array}{l} x_k^{i-1}[3]+x_k^{i-1}[4]-x_k^{i}[0]=0\\ x_k^{i-1}[2]-x_k^{i}[1]=0\\ x_k^{i-1}[0]-x_k^{i}[2]=0\\ x_k^{i-1}[4]-x_k^{i}[3]=0\\ x_k^{i-1}[1]+x_k^{i-1}[2]-x_k^{i}[4]=0\\ k\in[0,5) \end{array}\right\} \tag{9}$$

定义ACE置换解密方向第i步的输出对应的模型变量为$(y_k^i[0],y_k^i[1],\cdots,y_k^i[4])$，则第$i$步解密方向的步函数对应的字传播模型为

$$\left.\begin{array}{l} y_k^{i-1}[2]-y_k^{i}[0]=0\\ y_k^{i-1}[1]+y_k^{i-1}[4]-y_k^{i}[1]=0\\ y_k^{i-1}[1]-y_k^{i}[2]=0\\ y_k^{i-1}[0]+y_k^{i-1}[3]-y_k^{i}[3]=0\\ y_k^{i-1}[3]-y_k^{i}[4]=0\\ k\in[0,5) \end{array}\right\} \tag{10}$$

利用式(9)和式(10)，r次迭代后就可以得到r步的字传播模型。然后，将模型输入到算法1和算法2中，使用求解器对模型进行求解(求解器：Gurobi8.0，编程语言：Python2.7)，并遍历$k\in[0,5)$，得到的结果如表4和表5所示。

由表4和表5可得$r_{\mathrm{ed}}=12$，$\phi=[0,2,3]$，则ACE置换存在1个12步的积分区分器，对应的区分器的具体形式如表6所示。

由于13步加密后，仅有字$\boldsymbol{E}^{13}$的表达式中包含了字$\boldsymbol{B}^{12}$，同时$\boldsymbol{E}^{13}$中还包含了$\boldsymbol{C}^{12}$，则12步的积分区分器不能继续向下扩展1步，所以，本文得到的ACE置换的积分区分器的步数为12步。相比于ACE设计文档中给出的结果[19]，本文给出的积分区分器步数更高，需要的数据量更少。结果对比如表7。

表 4　ACE置换对应的r_{e}^k和ω_k

k	r_{e}^k	ω_k
0	8	[1]
1	8	[3]
2	7	[1]
3	9	[1]
4	7	[3]

表 5　ACE置换对应的r_{d}^k和φ_k

k	r_{d}^k	φ_k
0	4	[0]
1	3	[4]
2	5	[0]
3	3	[0]
4	4	[4]

表 6　ACE置换12步的积分区分器

输入形式	输出形式
$c^{64}a^{64}a^{64}a^{64}a^{64}$	$u^{64}b^{64}u^{64}u^{64}u^{64}$

注：c^{64}表示任意的一个64 bit的常数，a^{64}表示64 bit的活跃字集，b^{64}表示64 bit的平衡字集，u^{64}表示64 bit的未知字集。

表 7　ACE置换积分分析结果对比

分析方法	积分区分器步数	选择明文量	方法
除属性	8	2^{319}	文献[19]
字传播轨迹	12	2^{256}	本文

5　结束语

本文给出了字传播轨迹的概念，构建了相应的MILP模型来描述字传播轨迹，并在此基础上设计了一种新的积分区分器自动化搜索方法，同时利用二分法减少了需要求解模型的次数。利用这个自动化分析工具，对ACE置换抵抗积分分析的能力进行了评估，发现了12步的积分区分器，在ACE的设计文档中，设计者利用可分性搜索到了8步的积分区分器，所以，相比ACE设计文档中的区分器，本文的区分器提高了4步。虽然，本文的区分器没有对ACE认证加密算法的安全性造成威胁(ACE认证加密算法中ACE置换的总步数为16步)，但是本文给出了一种自动化搜索分组密码算法积分区分器的新方法，提高了密码分析的效率，为以后的分组密码算法的设计与分析提供了强有力的工具。

参 考 文 献

[1] MATSUI M. Linear cryptanalysis method for DES cipher[C]. Workshop on the Theory and Application of Cryptographic Techniques on Advances in Cryptology, Lofthus, Norway, 1993: 386–397. doi: 10.1007/3-540-48285-7_33.

[2] BIHAM E and SHAMIR A. Differential cryptanalysis of DES-like cryptosystems[J]. *Journal of Cryptology*, 1991, 4(1): 3–72. doi: 10.1007/BF00630563.

[3] BIHAM E, BIRYUKOV A, and SHAMIR A. Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials[C]. International Conference on the Theory and Application of Cryptographic Techniques Prague on Advances in Cryptology, Prague, Czech Republic, 1999: 12–23. doi: 10.1007/3-540-48910-X_2.

[4] 韦永壮, 史佳利, 李灵琛. LiCi分组密码算法的不可能差分分析[J]. 电子与信息学报, 2019, 41(7): 1610–1617. doi: 10.11999/

JEIT180729.
WEI Yongzhuang, SHI Jiali, and LI Lingchen. Impossible differential cryptanalysis of LiCi block cipher[J]. *Journal of Electronics & Information Technology*, 2019, 41(7): 1610–1617. doi: 10.11999/JEIT180729.

[5] DINUR I and SHAMIR A. Cube attacks on tweakable black box polynomials[C]. The 28th Annual International Conference on the Theory and Applications of Cryptographic Techniques on Advances in Cryptology, Cologne, Germany, 2009: 278–299. doi: 10.1007/978-3-642-01001-9_16.

[6] KNUDSEN L and WAGNER D. Integral cryptanalysis[C]. The 9th International Workshop on Fast Software Encryption, Leuven, Belgium, 2002: 112–127. doi: 10.1007/3-540-45661-9_9.

[7] TODO Y. Structural evaluation by generalized integral property[C]. The 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques on Advances in Cryptology, Sofia, Bulgaria, 2015: 287–314. doi: 10.1007/978-3-662-46800-5_12.

[8] LI Yanjun, WU Wenling, and ZHANG Lei. Improved integral attacks on reduced-round CLEFIA block cipher[C]. The 12th International Workshop on Information Security Applications, Jeju Island, Korea, 2011: 28–39. doi: 10.1007/978-3-642-27890-7_3.

[9] Z'ABA M R, RADDUM H, HENRICKSEN M, *et al*. Bit-Pattern based integral attack[C]. The 15th International Workshop on Fast Software Encryption, Lausanne, Switzerland, 2008: 363–381. doi: 10.1007/978-3-540-71039-4_23.

[10] LIU Meicheng. Degree evaluation of NFSR-based cryptosystems[C]. The 37th Annual International Cryptology Conference on Advances in Cryptology, Santa Barbara, USA, 2017: 20–24. doi: 10.1007/978-3-319-63697-9_8.

[11] BOURA C, CANTEAUT A, and DE CANNIÈRE C. Higher-order differential properties of KECCAK and *LUFFA*[C]. The 18th International Workshop on Fast Software Encryption, Lyngby, Denmark, 2011: 252–269. doi: 10.1007/978-3-642-21702-9_15.

[12] TODO Y and MORII M. Bit-based division property and application to SIMON family[C]. The 23rd International Workshop on Fast Software Encryption, Bochum, Germany, 2016: 357–377. doi: 10.1007/978-3-662-52993-5_18.

[13] XIANG Zejun, ZHANG Wentao, BAO Zhenzhen, *et al*. Applying MILP method to searching integral distinguishers based on division property for 6 lightweight block ciphers[C]. The 22nd International Conference on the Theory and Application of Cryptology and Information Security on Advances in Cryptology, Hanoi, Vietnam, 2016: 648–678. doi: 10.1007/978-3-662-53887-6_24.

[14] WANG Senpeng, HU Bin, GUAN Jie, *et al*. MILP-aided method of searching division property using three subsets and applications[C]. The 25th International Conference on the Theory and Application of Cryptology and Information Security on Advances in Cryptology, Kobe, Japan, 2019: 398–427. doi: 10.1007/978-3-030-34618-8_14.

[15] ESKANDARI Z, KIDMOSE A B, KÖLBL S, *et al*. Finding integral distinguishers with ease[C]. The 25th International Conference on Selected Areas in Cryptography, Calgary, Canada, 2018: 115–138. doi: 10.1007/978-3-030-10970-7_6.

[16] ZHANG Wenying and RIJMEN V. Division cryptanalysis of block ciphers with a binary diffusion layer[J]. *IET Information Security*, 2019, 13(2): 87–95. doi: 10.1049/iet-ifs.2018.5151.

[17] 徐洪, 方玉颖, 戚文峰. SIMON64算法的积分分析[J]. 电子与信息学报, 2020, 42(3): 720–728. doi: 10.11999/JEIT190230.
XU Hong, FANG Yuying, and QI Wenfeng. Integral attacks on SIMON64[J]. *Journal of Electronics & Information Technology*, 2020, 42(3): 720–728. doi: 10.11999/JEIT190230.

[18] ZHANG Wenying, CAO Meichun, GUO Jian, *et al*. Improved security evaluation of SPN block ciphers and its applications in the single-key attack on SKINNY[J]. *IACR Transactions on Symmetric Cryptology*, 2020, 2019(4): 171–191. doi: 10.13154/tosc.v2019.i4.171-191.

[19] AAGAARD M, ALTAWY R, GONG Guang, *et al*. ACE: An authenticated encryption and hash algorithm[EB/OL]. https://uwaterloo.ca/communications-security-lab/lwc/ace, 2020.

[20] Computer Security Resource Center. Lightweight cryptography[EB/OL]. https://csrc.nist.gov/Projects/lightweight-cryptography, 2020.

[21] BOGDANOV A and SHIBUTANI K. Generalized Feistel networks revisited[J]. *Designs, Codes and Cryptography*, 2013, 66(1): 75–97. doi: 10.1007/s10623-012-9660-z.

[22] YANG Gangqiang, ZHU Bo, SUDER V, *et al*. The SIMECK family of lightweight block ciphers[C]. The 17th International Workshop on Cryptographic Hardware and Embedded Systems, Saint-Malo, France, 2015: 307–329. doi: 10.1007/978-3-662-48324-4_16.

叶 涛：男，1991年生，博士生，研究方向为对称密码算法设计与分析.

韦永壮：男，1976年生，教授，博士生导师，研究方向为对称密码算法设计与分析、加密芯片侧信道攻击与防御技术、网络安全协议分析.

李灵琛：女，1988年生，博士，研究方向为分组密码算法设计与分析.

责任编辑：马秀强