轻量级LEA的代数统计故障分析

李玮[1,2,3]，张爱琳[1]，王弈[4]，谷大武[3]，李诗雅[1]
（1. 东华大学计算机科学与技术学院，上海 201620；2. 东华大学数字化纺织服装技术教育部工程研究中心，上海 201620；
3. 上海交通大学网络空间安全学院，上海 200240；4. 华东政法大学智能科学与信息法学系，上海 200042）

摘　要：针对LEA的实现安全，结合其结构和实现特点，提出了一种新型唯密文故障分析方法，即代数统计故障分析。该方法基于随机半字节故障模型，分析和构造代数关系并结合注入故障前后的统计分布变化来破译LEA，设计了Hellinger距离、Hellinger距离–汉明重量和Hellinger距离–极大似然3种新型区分器。实验结果表明，所提方法使故障注入轮数更深一轮，新型区分器最少仅需72个故障即可破译LEA的128 bit原始密钥，为智能小型设备中密码算法的安全性评估提供了有价值的参考。

关键词：LEA；轻量级密码；密码分析；代数关系；智能小型设备
中图分类号：TP309.7
文献标志码：A
DOI：10.11959/j.issn.1000–436x.2025136

Algebraic statistical fault analysis of the lightweight LEA

LI Wei[1,2,3], ZHANG Ailin[1], WANG Yi[4], GU Dawu[3], LI Shiya[1]
1. School of Computer Science and Technology, Donghua University, Shanghai 201620, China
2. Engineering Research Center of Digitalized Textile and Fashion Technology, Ministry of Education, Donghua University, Shanghai 201620, China
3. School of Cyber Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
4. Department of Intelligence Science and Information Law, East China University of Political Science and Law, Shanghai 200042, China

Abstract: A novel ciphertext-only fault analysis method termed algebraic statistical fault analysis was proposed for enhancing the implementation security of the LEA in light of its structural and implementation characteristics. Based on the random nibble-oriented fault model, algebraic relationships were analyzed and constructed, and coupled with statistical inference between pre-injection and post-injection intermediate states, the LEA was decrypted. Additionally, Hellinger distance, Hellinger distance-Hamming weight, and Hellinger distance-maximum likelihood distinguishers were designed. Experimental results demonstrate that the proposed method extends fault injection to an additional deeper round, and the novel distinguisher successfully recovers the 128 bit secret key of the LEA with a minimum of 72 fault injections, providing valuable references for security evaluation of other cryptographic algorithms in smart small devices.

Keywords: LEA, lightweight cipher, cryptanalysis, algebraic relation, smart small device

收稿日期：2025–04–21；修回日期：2025–07–23
通信作者：李玮，liwei.cs.cn@gmail.com
基金项目：国家自然科学基金资助项目（No.62472286）；上海市自然科学基金资助项目（No.24ZR1401300）；上海市扬帆计划基金资助项目（No.23YF1401000）；中央高校基本科研业务费专项资金资助项目（No.223202D-25）；数字化纺织服装技术教育部工程研究中心自主研究课题基金资助项目
Foundation Items：The National Natural Science Foundation of China (No.62472286), The Natural Science Foundation of Shanghai (No.24ZR1401300), Shanghai Sailing Plan (No.23YF1401000), The Fundamental Research Funds for the Central Universities (No.223202D-25), The Research Fund of Engineering Research Center of Digitalized Textile and Fashion Technology, Ministry of Education

0　引言

近年来，随着各种传感技术、近场通信技术以及云计算技术的集成应用，智能手机、平板电脑和蓝牙耳机等智能小型设备在物联网环境下得到了广泛应用，如图1所示。作为物联网的关键终端节点，智能小型设备在采集用户生理特征、行为轨迹及位置信息等敏感数据的同时，面临着严峻的信息安全挑战。此类设备生成的高价值隐私数据若遭受恶意截获，将直接导致用户身份泄露、财产损失等安全风险。然而，受限于嵌入式平台的计算能力和存储资源预算的约束，传统分组密码算法难以实现安全性能与能效比的平衡。基于此背景，轻量级密码算法（LEA, lightweight encryption algorithm）应运而生，其目的在于为资源受限的应用场景提供高效且安全的解决方案，确保智能小型设备的数据安全性。随着国际标准化组织、美国国家标准与技术研究院等机构积极推动轻量级密码标准的制定工作，轻量级密码算法的研究已成为当前国际密码学界的一个重要方向[1-7]。

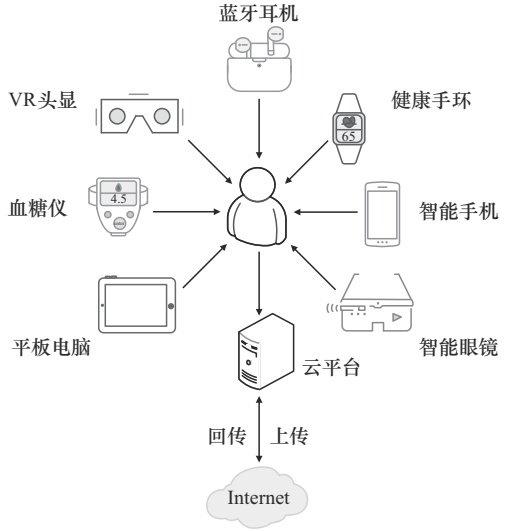

图1　智能小型设备

Hong等[8]提出了轻量级密码算法，该算法已被确立为国际轻量级密码标准ISO/IEC 29192－2:2019，并被纳入韩国密码标准KS X 3246之中，以其高效实现特性和鲁棒性安全等特点，成为物联网和资源受限环境下数据安全保护的重要算法之一。现有LEA的传统密码分析方法包括飞去来器分析、零相关线性分析、积分分析、差分分析和差分线性分析等[8-12]。但是，此类方法在实际物联网环境下的可行性较低，而攻击者运用故障分析通过操控电压、温度或电磁脉冲等物理手段在密码设备运行时诱发设备自行产生计算错误，使加密过程中寄存器或S盒等关键模块的中间值产生异常，再统计错误密文和正常密文的关联性差异进行密码破译的这种方法，更容易在实际应用环境下实现。

故障分析是Boneh等[13]提出的一种分析方法，现已成为评估密码算法安全性的重要指标之一。该方法通过向密码算法的执行过程引入特定干扰，导致算法产生错误输出，从而获得正确密文与因故障引发的错误密文对。基于这些密文对间的差异性和相关性，研究人员就能够运用概率统计学的相关方法逐步推断出加密密钥，实现对密码设备的破解。当前，为提高攻击效率，攻击者在实际攻击过程中常会采用多种方法结合的策略。因此，近年来故障分析领域涌现出多种组合攻击方法，其中，差分故障分析和不可能差分故障分析通过将差分和不可能特性与故障分析相结合以破译密码[14-15]；代数故障分析和中间相遇故障分析分别利用密码结构中的代数关系和中间相遇策略与故障分析结合来恢复密钥[16-17]；链接故障分析利用指令跳过机制，在2个半字节值之间创建相等关系以恢复密钥[18]；而统计故障分析则是在密码算法加密过程中的最后几轮引入故障从而获取错误密文，再利用获取的错误密文倒推至注入故障位置的中间状态值并结合统计分析进行部分解密，进而逐步提取相关密钥的全部信息[19]。此外，近年来研究表明，攻击者通过注入故障对密码设备中有效的侧信道分析防护机制进行破坏，再利用侧信道分析便可提升密钥的恢复效率并缩短恢复时间[20]。上述各类密码分析方法，已成为轻量级密码安全实现领域的重要威胁。

在故障分析中，根据攻击者获取密码信息的能力，密码分析方法可以分为唯密文攻击、已知明文攻击和选择明文攻击。针对LEA的密码分析的基本假设主要是选择明文攻击和已知明文攻击，即攻击者需要截获明文密文对或者特定明文产生的密文，这对攻击者的能力要求较高。而唯密文攻击仅需攻击者获取密文，在资源受限的小型设备上更易实现。

LEA公布之后，国内外学者对其展开了系统性的研究和讨论。Hong等[8]首次对其进行了线性、飞去来器和不可能差分分析，通过抑制线性掩码传

播搜索到的线性逼近表达式并结合Matsui算法，成功实现了11轮的线性分析。此外，还使用飞去来器攻击，以$2^{116.3}$的数据复杂度恢复了15轮的密钥。同时，他们还利用10轮不可能差分特征，实现了11轮的不可能差分分析，可有效恢复最后一轮轮密钥的部分信息。Zhang等[9]通过改进零相关线性密码分析成功破译了LEA的第9轮密钥。李航等[10]利用密钥扩展算法的性质和部分和技术并结合零相关区分器和积分区分器，以2^{120}的攻击计算复杂度实现了10轮积分分析。李艳俊等[11]通过多路径差分特征，开发出具有14轮攻击深度的新型差分分析。Chen等[12]提出了新型差分线性逼近函数和分区树结构，成功实现了17轮的差分线性分析。

随着密码学研究的深入，国内外学者的分析重点从传统理论转向实现层面，故障分析与功耗分析、侧信道攻击等其他方法的结合成为研究的重要方向。Lim等[21]通过优化差分特征并构建密文索引机制，将所需故障数降低了70.9%，实现了LEA第23轮的差分故障分析，显著扩展了差分故障分析的攻击范围。张金煜等[22]利用统计故障分析在倒数第2轮注入随机故障，恢复了LEA的128 bit密钥。表1汇总了LEA的安全性分析对比。

本文基于唯密文攻击，对LEA注入随机半字节故障，并结合该算法的代数和统计特性，提出了新型代数统计故障分析方法，并结合汉明重量（HW, Hamming weight）、拟合优度（GF, goodness of fit）和极大似然（ML, maximum likelihood）估计等经典区分器，以及Hellinger距离（HD, Hellinger distance）区分器、Hellinger距离–汉明重量（HD-HW, Hellinger distance-Hamming weight）和Hellinger距离–极大似然（HD-ML, Hellinger distance-maximum likelihood）3种新型组合区分器，恢复了该算法的原始密钥。本文采用的随机半字节故障模型的可行性和实操性，已经通过Balasch等[23]对时钟线路进行精密毛刺注入的实验获得了验证。表2展示了使用上述2种故障分析方法和6种区分器破译LEA的结果对比。

表1　　LEA的安全性分析对比

分析类型	基本假设	攻击轮数/轮	故障注入轮	文献
线性分析	已知明文攻击	11	—	文献[8]
飞去来器分析	已知明文攻击	15	—	文献[8]
零相关线性分析	选择明文攻击	9	—	文献[9]
不可能差分分析	选择明文攻击	11	—	文献[8]
积分分析	选择明文攻击	11	—	文献[10]
差分分析	选择明文攻击	14	—	文献[11]
差分线性分析	选择明文攻击	17	—	文献[12]
差分故障分析	选择明文攻击	24	第23轮	文献[21]
统计故障分析	唯密文攻击	24	第23轮	文献[22]
代数统计故障分析	唯密文攻击	24	第22轮	本文方法

目前，国内外针对LEA等基于ARX（addition-rotation-XOR）结构的密码算法的传统唯密文攻击仅能支持在倒数第2轮实施故障注入。然而，本文方法通过建立轮函数间的代数关系，成功将故障分析深度扩展至倒数第3轮。同时，本文提出的新型区分器在保证99.0%及以上攻击成功率的同时，将所需故障数降低了84.2%。更少的故障数不但可以减少资源消耗，而且更容易在实际应用环境下实现。本文方法为评估基于ARX结构的密码算法在抵御统计故障分析方面的安全性能提供了重要的参考依据。

表2　　统计故障分析和代数统计故障分析恢复LEA原始密钥的结果

区分器	统计故障分析			代数统计故障分析		
	故障注入轮	故障数/个	成功率	故障注入轮	故障数/个	成功率
GF	第23轮	480	≥99.0%	第22轮	220	≥99.0%
ML	第23轮	468	≥99.0%	第22轮	180	≥99.0%
HW	第23轮	456	≥99.0%	第22轮	152	≥99.0%
HD	—	—	—	第22轮	85	≥99.0%
HD-HW	—	—	—	第22轮	78	≥99.0%
HD-ML	—	—	—	第22轮	72	≥99.0%

1　LEA简介

1.1　符号说明

设Z_2^b为b bit的二进制向量集；

记$M \in (Z_2^{32})^4$为明文，$Y \in (Z_2^{32})^4$为密文，$\hat{Y} = \hat{Y}^0 \| \hat{Y}^1 \| \hat{Y}^2 \| \hat{Y}^3 \in (Z_2^{32})^4$为错误密文；

记$K=K^0\|K^1\|K^2\|K^3 \in (Z_2^{32})^4$为128 bit原始密钥；

记$\mathrm{RK}_i = \mathrm{rk}_i^0 \| \mathrm{rk}_i^1 \| \mathrm{rk}_i^2 \| \mathrm{rk}_i^3 \| \mathrm{rk}_i^4 \| \mathrm{rk}_i^5 \in (Z_2^{32})^6$为第$i+1$轮轮密钥的6个32 bit数据块，其中，$i \in [0,23]$；

记$X_i = x_i^0 \| x_i^1 \| x_i^2 \| x_i^3 \in (Z_2^{32})^4$为第$i$+1轮中间状态的4个32 bit数据块，其中，$i \in [0,23]$；

记⊕、⊞、⊟、‖和&分别表示按位异或、模2^{32}加、模2^{32}减、级联和按位与操作，<<<和>>>分别表示比特串的循环左移和循环右移。

1.2　LEA

国际轻量级密码标准LEA由Hong等[8]于2014年提出，其框架基于ARX结构构建。该算法采用128 bit的固定分组长度并支持128 bit、192 bit和256 bit的密钥长度，其迭代轮数根据密钥长度的不同分别是24、28和32轮。

1.2.1　加解密过程

LEA的每轮轮函数包括3种操作：按位异或、模2^{32}加和循环移位。该算法的轮函数结构如图2所示，算法1给出了128 bit密钥长度的LEA的加密过程。解密算法是加密算法的逆变换过程，通过逆向轮序调用轮密钥，并将模加操作替换为模减操作实现解密。

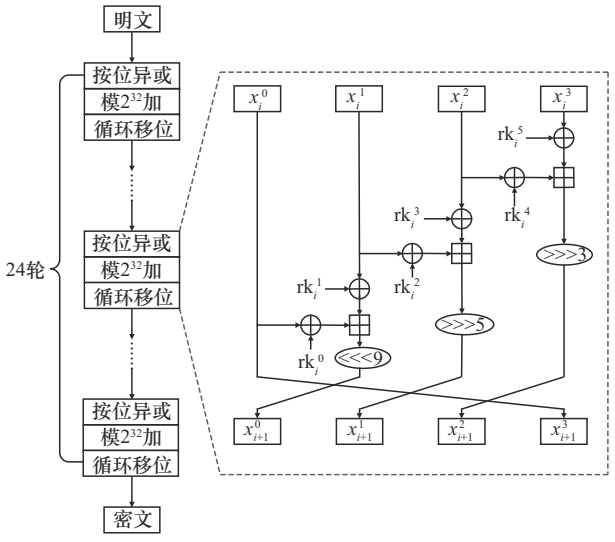

图2　LEA轮函数结构

算法1　LEA加密过程

输入　M，RK_i

输出　Y

1) $x_0^0\|x_0^1\|x_0^2\|x_0^3 = M$

2) for i = 0 to 23 do

3)　$x_{i+1}^0 = (x_i^0 \oplus \mathrm{rk}_i^0) \boxplus (x_i^1 \oplus \mathrm{rk}_i^1) <<< 9$

4)　$x_{i+1}^1 = (x_i^1 \oplus \mathrm{rk}_i^2) \boxplus (x_i^2 \oplus \mathrm{rk}_i^3) >>> 5$

5)　$x_{i+1}^2 = (x_i^2 \oplus \mathrm{rk}_i^4) \boxplus (x_i^3 \oplus \mathrm{rk}_i^5) >>> 3$

6)　$x_{i+1}^3 = x_i^0$

7) end for

8) $Y = x_{24}^0\|x_{24}^1\|x_{24}^2\|x_{24}^3$

9) return Y

1.2.2　密钥编排方案

算法2给出了LEA的密钥编排方案，其中，$\delta[\tau]$为常数[8]，$i \in [0,23]$，$\tau = i \bmod 4 \in [0,3]$。

算法2　LEA密钥编排方案

输入　K，$\delta[\tau]$

输出　$\mathrm{RK}_0,\mathrm{RK}_1,\cdots,\mathrm{RK}_{23}$

1) for i = 0 to 23 do

2)　$K^0 = (K^0 \boxplus (\delta[\tau] <<< i)) <<< 1$

3)　$K^1 = (K^1 \boxplus (\delta[\tau] <<< (i+1))) <<< 3$

4)　$K^2 = (K^2 \boxplus (\delta[\tau] <<< (i+2))) <<< 6$

5)　$K^3 = (K^3 \boxplus (\delta[\tau] <<< (i+3))) <<< 11$

6)　$\mathrm{RK}_i = K^0\|K^1\|K^2\|K^1\|K^3\|K^1$

7) end for

8) return $\mathrm{RK}_0,\mathrm{RK}_1,\cdots,\mathrm{RK}_{23}$

2　代数统计故障分析

2.1　代数分析

作为现代密码评估的重要方法之一，代数分析在其发展历程中得到了国内外研究者的广泛关注和深入探索。早期代数分析的核心是根据密码的结构构建代数方程组，攻击者再使用多元多项式方程来求解并恢复密钥。Courtois等[24]针对多元多项、稀疏方程组的求解提出了XL（extended linearation）算法，并且在2002年改进了该算法用于分析Rijndeal和Serpent算法[25]。Bard等[26]将分组密码KATAN表示成若干二次方程形式，并使用改进优化后的SAT（satisfiability problem）求解器来求解方程，破解了KATAN32、KATAN48和KATAN64的79、64和60轮。

近年来，代数分析方法与其他密码分析方法的结合逐步成为重要趋势。关杰等[27]将流密码Salsa20

表示成若干非线性方程并结合该算法的2个3轮高概率截断差分传递链，实现了5轮Salsa20算法的代数—截断差分攻击。黄静等[28]对轻量级分组密码PRESENT算法的加解密过程和密钥编排方案进行了分析并建立了代数方程，结合故障分析实现了对该算法的代数故障分析。Le等[29]利用简化Gröbner基求解、SAT求解器攻击和差分轨迹分析3种不同的代数方法与差分故障分析相结合，恢复了分组密码SIMON家族的完整密钥。Li等[30]针对杂凑函数GMiMCHash，引入多元变量改进所提代数差分分析方法，在降低方法复杂度的同时，提升了发现碰撞的概率，拓展了代数分析的范围。

2.2 统计故障分析

随着密码学的发展，统计故障分析逐渐成为国内外学术界和工业界的研究热点。以唯密文攻击为基本假设的统计故障分析由Fuhr等[31]提出，该方法通过统计故障注入导致的不均匀分布，构建基于假设检验理论的统计区分器，对密钥搜索范围进行缩减，成功破译了分组密码AES。Nozaki等[32]使用统计故障分析，通过在时钟中插入毛刺产生故障，利用中间状态的汉明重量最小平均值，恢复了轻量级分组密码TWINE的80 bit主密钥。Harmouch等[33]使用统计故障分析对RC4和WAKE等流密码进行了研究，发现一些常见的流密码所生成的密文分布存在不均匀现象，拓展了统计故障分析的攻击范围。

李玮等[34-35]利用统计故障分析破译了SIMON算法，并将中间相遇策略与统计故障分析相结合成功破译PRESENT轻量级密码，为统计故障分析与其他分析方法相结合提供了新思路。

2.3 代数统计故障分析

本文针对LEA进行统计故障分析时发现，该算法的结构特性导致单一的统计故障分析在倒数第3轮注入故障时，该算法会输出规模为2^{17}数量级的候选密钥值。同时，通过增加统计故障分析的限制条件也难以有效缩减候选密钥范围。

基于上述问题，本文深入研究了LEA的加解密过程及其密钥编排方案，发现该算法的最后2轮轮密钥间存在特定的代数变换关系，同时，最后2轮轮密钥内部也存在特定的代数约束条件，2.4节详细分析了代数关系。然而，统计故障分析方法的核心是依赖概率统计意义上的分布拟合程度来筛选密钥。所发现的2种代数关系本质上是确定性的约束条件。因此，在理论上，利用这2个内在的代数关系对统计故障分析输出的候选密钥范围进行2轮筛选，可实现候选密钥范围的缩减。

因此，本文提出了一种代数统计故障分析方法，通过将LEA内在的代数关系与统计故障分析相结合，改进了单一统计故障分析方法在攻击LEA时无法注入更深轮以及输出的候选密钥值多的问题。表3展示了其他相似分析方法和本文方法攻击LEA的对比结果。

表3 相似分析方法攻击LEA的对比结果

分析类型	故障注入轮	故障数/个	文献
差分故障分析	第24轮	300	文献[36]
差分故障分析	第24轮	258	文献[37]
统计故障分析	第23轮	456	文献[22]
代数统计故障分析	第22轮	72	本文方法

2.4 代数关系分析

根据轮密钥的符号定义可知

$$\begin{cases} \mathrm{RK}_{22} = \mathrm{rk}_{22}^{0}\|\mathrm{rk}_{22}^{1}\|\mathrm{rk}_{22}^{2}\|\mathrm{rk}_{22}^{3}\|\mathrm{rk}_{22}^{4}\|\mathrm{rk}_{22}^{5} \\ \mathrm{RK}_{23} = \mathrm{rk}_{23}^{0}\|\mathrm{rk}_{23}^{1}\|\mathrm{rk}_{23}^{2}\|\mathrm{rk}_{23}^{3}\|\mathrm{rk}_{23}^{4}\|\mathrm{rk}_{23}^{5} \end{cases} \tag{1}$$

由于LEA每轮所使用的轮密钥是由前一轮轮密钥通过密钥编排算法正向推导而来。因此，攻击者可以利用已获取的轮密钥通过密钥编排算法逆向推导得到前一轮的轮密钥信息。即最后2轮轮密钥会存在如式(2)所示代数变换关系。

$$\begin{cases} \mathrm{rk}_{22}^{0} = (\mathrm{rk}_{23}^{0} >>> 1) \boxminus (\delta[3] <<< 23) \\ \mathrm{rk}_{22}^{1} = (\mathrm{rk}_{23}^{1} >>> 3) \boxminus (\delta[3] <<< 24) \\ \mathrm{rk}_{22}^{2} = (\mathrm{rk}_{23}^{2} >>> 6) \boxminus (\delta[3] <<< 25) \\ \mathrm{rk}_{22}^{3} = \mathrm{rk}_{22}^{1} \\ \mathrm{rk}_{22}^{4} = (\mathrm{rk}_{23}^{4} >>> 11) \boxminus (\delta[3] <<< 26) \\ \mathrm{rk}_{22}^{5} = \mathrm{rk}_{22}^{1} \end{cases} \tag{2}$$

同时，攻击者根据2.6节攻击过程中步骤2，将枚举的轮密钥和错误密文集合逆向推导至LEA第22轮的中间状态值，并基于该中间状态值的统计特性利用2.7节中的区分器，对错误候选密钥进行初步过滤，并将剩余的候选密钥表示为$(\mathrm{RK}_{22},\mathrm{RK}_{23})$的密钥对形式。对于剩余的候选密钥对，攻击者结合式(1)的定义，再利用式(2)的代数变换关系对每一个候选密钥对中的RK_{23}进行逆向推导并得到其对应的理论值记为RK'_{22}，若满足

$$\mathrm{RK}_{22} = \mathrm{RK}'_{22} \tag{3}$$

则保留为有效候选密钥对，否则筛除该密钥对。

此外，鉴于LEA的密钥编排算法特点，同一轮轮密钥内部也存在代数变换关系，表示为

$$\mathrm{rk}^1_{23} = \mathrm{rk}^3_{23} = \mathrm{rk}^5_{23} \tag{4}$$

若某个候选密钥能同时满足式(3)和式(4)，则该候选密钥极有可能是正确的密钥，故保留。同时，不满足上述代数变换关系的一定不是正确密钥，故排除。

综合上述分析，本文研究发现LEA存在上述2种代数变换关系，利用这2种代数变换关系依次对统计故障分析输出的候选密钥集合进行筛选，便可达到降低密钥搜索复杂度的目的。

2.5　基本假设和故障模型

在进行实验攻击前，攻击者需先定好基本假设和采用的故障模型。本文将对攻击者的先验知识和能力要求较低的唯密文攻击作为基本假设，即攻击者仅需收集若干个密钥加密随机明文所生成的多组错误密文，便可结合统计分析恢复密钥。该种假设在实际应用环境下更易实现。

本文采用随机半字节故障模型，即通过按位“与”的操作，在加密过程的某一轮中注入半字节随机故障，影响中间状态对应比特的值呈现非均匀分布。

以单比特为例，当密码算法正常运行时，某一中间状态值理论上出现0和1的概率是1:1，即每个值出现的概率为50%，属于均匀分布。

本文采用的“与”操作的特性使该中间状态值出现0和1的概率变为3:1，即非均匀分布。式(5)给出了单比特经“与”操作后出现的非均匀分布的结果。

$$a\&b = \begin{cases} 0, a = 0且b = 0 \\ 0, a = 0且b = 1 \\ 0, a = 1且b = 0 \\ 1, a = 1\ 且b = 1 \end{cases} \tag{5}$$

2.6　攻击过程

本节提出的代数统计故障分析的攻击过程包括以下5个步骤。

步骤1　故障注入。攻击者将随机半字节故障通过“与”操作注入LEA加密过程的第22轮中并收集错误密文。图3以故障导入在第22轮的首个半字节为例，即$x^0_{21}[0]$、$x^0_{21}[1]$、$x^0_{21}[2]$和$x^0_{21}[3]$，并展示了故障扩散路径。

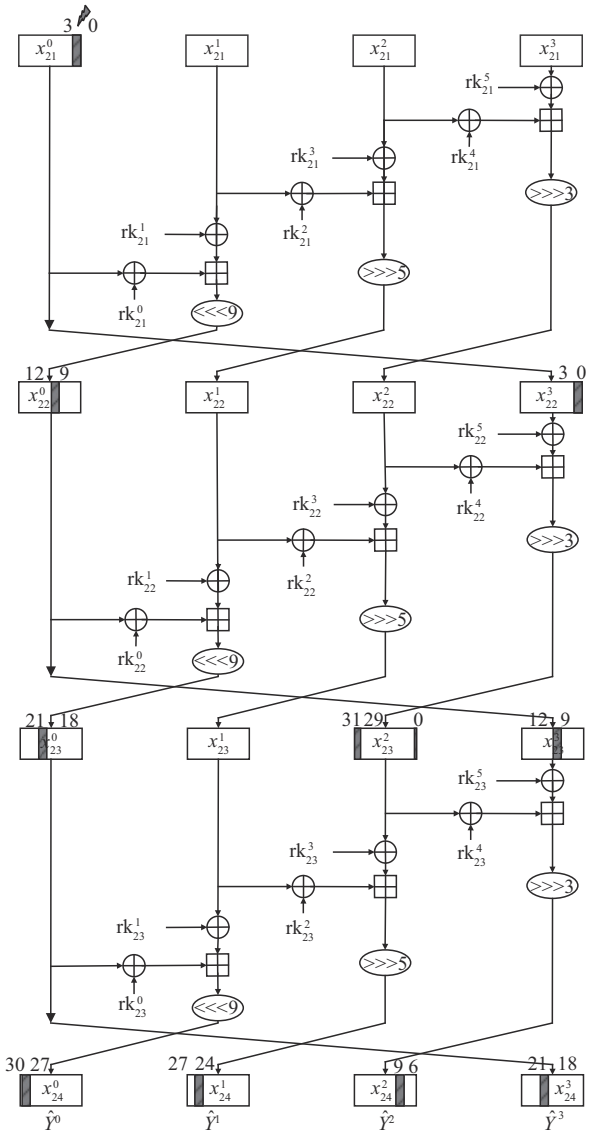

图3　注入故障后的故障扩散路径

步骤2　统计关系分析。攻击者通过结合受故障影响的错误密文和枚举的轮密钥，将错误密文倒推至LEA倒数第3轮的中间状态。恢复$x^0_{21}[0]$的推导过程如式(6)所示。

$$\begin{aligned} x^0_{21}[0] = & (((x^1_{24}[24] \boxminus ((x^0_{24}[6] \boxminus (x^3_{24}[29] \oplus \\ & \mathrm{rk}^0_{23}[29])) \oplus \mathrm{rk}^1_{23}[29] \oplus \mathrm{rk}^2_{23}[29])) \oplus \\ & \mathrm{rk}^3_{23}[29]) \boxminus ((((x^0_{24}[4] \boxminus (x^3_{24}[27] \oplus \\ & \mathrm{rk}^0_{23}[27])) \oplus \mathrm{rk}^1_{23}[27]) \boxminus ((x^3_{24}[9] (((x^2_{24}[29] \boxminus \\ & ((x^1_{24}[27] \boxminus ((x^0_{24}[9] \boxminus (x^3_{24}[0] \oplus \mathrm{rk}^0_{23}[0])) \oplus \\ & \mathrm{rk}^1_{23}[0] \oplus \mathrm{rk}^2_{23}[0])) \oplus \mathrm{rk}^3_{23}[0] \oplus \mathrm{rk}^4_{23}[0])) \oplus \\ & \mathrm{rk}^5_{23}[0]) \oplus \mathrm{rk}^0_{22}[0])) \oplus \mathrm{rk}^1_{22}[0] \oplus \\ & \mathrm{rk}^2_{22}[0])) \oplus \mathrm{rk}^3_{22}[0] \oplus \mathrm{rk}^4_{22}[0])) \oplus \mathrm{rk}^5_{22}[0] \end{aligned} \tag{6}$$

恢复$x_{21}^{0}[1]$的推导过程如式(7)所示。

$$x_{21}^{0}[1]=(((x_{24}^{1}[25]\boxminus((x_{24}^{0}[7]\boxminus(x_{24}^{3}[30]\oplus \mathrm{rk}_{23}^{0}[30]))\oplus \mathrm{rk}_{23}^{1}[30]\oplus \mathrm{rk}_{23}^{2}[30]))\oplus \mathrm{rk}_{23}^{3}[30])\boxminus((((x_{24}^{0}[5]\boxminus(x_{24}^{3}[28]\oplus \mathrm{rk}_{23}^{0}[28]))\oplus \mathrm{rk}_{23}^{1}[28])\boxminus((x_{24}^{3}[10]((((x_{24}^{2}[30]\boxminus((x_{24}^{1}[28]\boxminus((x_{24}^{0}[10]\boxminus(x_{24}^{3}[1]\oplus \mathrm{rk}_{23}^{0}[1]))\oplus \mathrm{rk}_{23}^{1}[1]\oplus \mathrm{rk}_{23}^{2}[1]))\oplus \mathrm{rk}_{23}^{3}[1]\oplus \mathrm{rk}_{23}^{4}[1]))\oplus \mathrm{rk}_{23}^{5}[1])\oplus \mathrm{rk}_{22}^{0}[1]))\oplus \mathrm{rk}_{22}^{1}[1]\oplus \mathrm{rk}_{22}^{2}[1]))\oplus \mathrm{rk}_{22}^{3}[1]\oplus \mathrm{rk}_{22}^{4}[1]))\oplus \mathrm{rk}_{22}^{5}[1] \tag{7}$$

同理，可恢复$x_{21}^{0}[2]$和$x_{21}^{0}[3]$的中间状态值。

步骤3 区分器筛选。攻击者使用2.7节中提出的区分器对步骤2获取的4 bit中间状态值进行计算，即可得到RK_{23}的48 bit和RK_{22}的24 bit。因此，通过重复步骤1~步骤3的操作，即多次注入半字节故障并倒推至倒数第3轮中间状态值，再通过区分器筛选，便可逐步获取RK_{23}和RK_{22}2轮轮密钥的全部信息。图4展示了注入故障位置与可恢复密钥比特对应关系。此外，为了方便后续的代数分析，现将筛选后剩余的候选密钥表示为$(\mathrm{RK}'_{22},\mathrm{RK}_{23})$的密钥对形式。

步骤4 代数关系构造。由2.4节的代数关系分析和该算法的密钥编排方案可知，最后2轮的轮密钥RK_{23}和RK_{22}间存在代数变换关系，满足该代数变换关系的候选密钥可能是正确的候选密钥，但不满足上述代数变换关系的一定不是正确密钥。同时，通过2.4节的分析可知，LEA的一轮轮密钥内部也存在代数变换关系。故攻击者可以利用式(4)的代数变换关系对被式(3)筛选后的候选密钥集合进行二次筛选。通过保留满足该代数变换关系的候选密钥，攻击者便可获得正确的RK_{23}。

步骤5 原始密钥恢复。攻击者根据步骤4中获取的RK_{23}，并利用密钥编排算法即可推导出RK_{0}，具体求解过程如算法3所示。

算法3 LEA原始密钥恢复过程

输入 RK_{23}，$\delta[\tau]$

输出 RK_{0}

1) for i = 23 to 0 do

2) $\mathrm{rk}_{23}^{0}=(\mathrm{rk}_{23}^{0}>>>1)\boxminus(\delta[\tau]<<<i)$

3) $\mathrm{rk}_{23}^{1}=(\mathrm{rk}_{23}^{1}>>>3)\boxminus(\delta[\tau]<<<(i+1))$

4) $\mathrm{rk}_{23}^{2}=(\mathrm{rk}_{23}^{2}>>>6)\boxminus(\delta[\tau]<<<(i+2))$

5) $\mathrm{rk}_{23}^{4}=(\mathrm{rk}_{23}^{4}>>>11)\boxminus(\delta[\tau]<<<(i+3))$

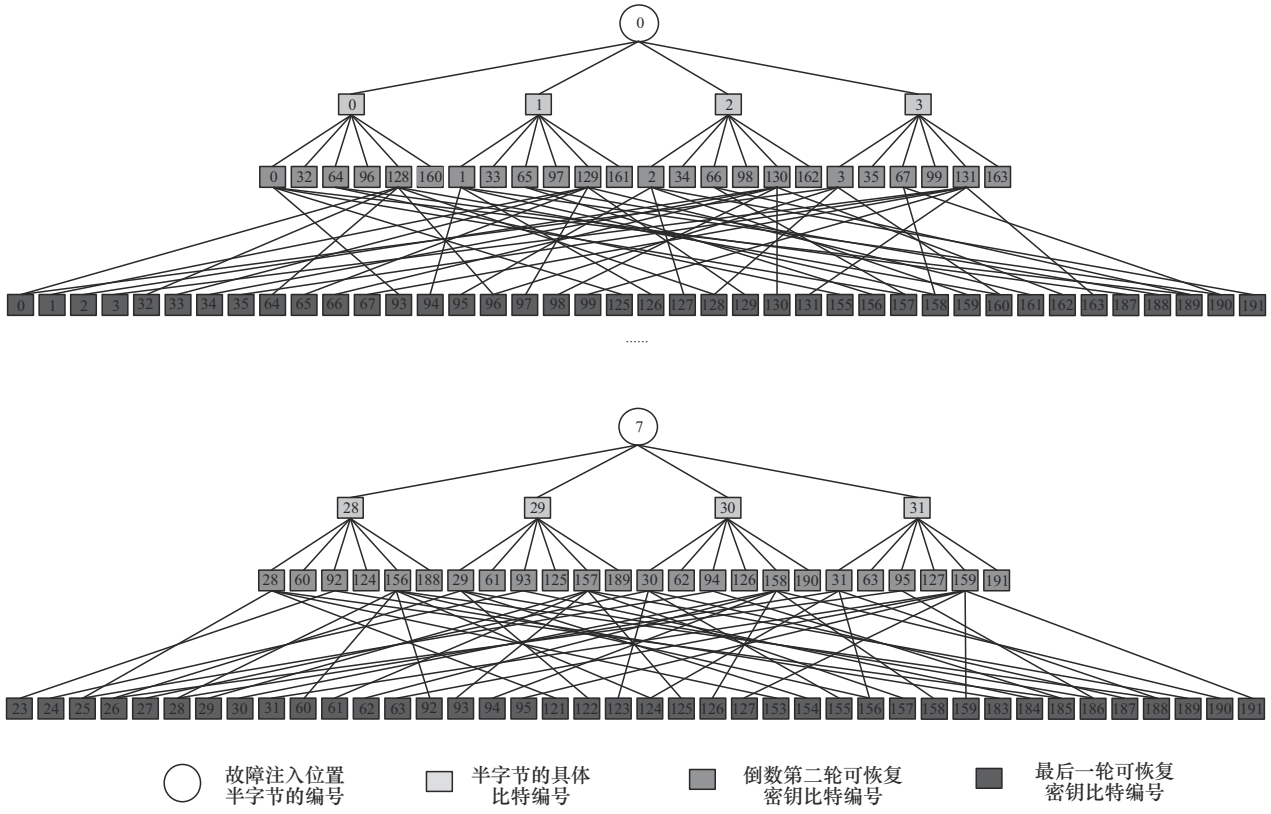

图4 注入故障位置与可恢复密钥比特对应关系

6)　$RK_i = rk_{23}^0||rk_{23}^1||rk_{23}^2||rk_{23}^1||rk_{23}^4||rk_{23}^1$

7) end for

8) return RK_0

攻击者再利用算法3中获得的RK_0，如式(8)所示。

$$RK_0 = rk_{23}^0||rk_{23}^1||rk_{23}^2||rk_{23}^1||rk_{23}^4||rk_{23}^1 \tag{8}$$

再结合RK_0和K的关系，便可恢复LEA的原始密钥，如式(9)所示。

$$K = K^0||K^1||K^2||K^3 = rk_{23}^0||rk_{23}^1||rk_{23}^2||rk_{23}^4 \tag{9}$$

2.7　区分器

2.7.1　经典区分器

1) 拟合优度

拟合优度通过评估由错误密文倒退得到的若干组中间状态值的实际分布与理论分布之间的统计一致性，为密码分析提供判别依据。最小拟合优度值对应的候选密钥即为正确轮密钥。拟合优度表达式为

$$GF = \sum_{a=0}^{1} \frac{(u_a - v_a)^2}{v_a} \tag{10}$$

其中，u_a表示实际观测到的中间状态值为a的个数，v_a表示理论预期的中间状态值为a的个数。

2) 极大似然估计

Fisher[38]提出了基于参数估计的极大似然估计方法。极大似然利用候选密钥对应的中间状态值的联合概率分布对正确轮密钥进行选取，其中，最大的ML值所对应的候选密钥即为正确轮密钥。极大似然估计的表达式为

$$ML = \prod_{z=1}^{m} \lg P_e(w_z) \tag{11}$$

其中，m表示注入的故障数量，w_z表示第z个故障产生的中间状态值，$P_e(w_z)$表示中间状态值为w_z出现的理论概率。

3) 汉明重量

Reed[39]提出了汉明重量概念即一个字符串中非零字符的数量。由式(5)可知，根据按位“与”操作的特性，故障注入后会扰乱中间状态0和1的平衡分布，使中间状态值中“0”出现的频率比“1”出现的频率明显增多。因此，攻击者可以计算中间状态的汉明重量值来区分候选密钥。最小汉明重量值对应的候选密钥即为正确轮密钥。汉明重量表达式为

$$HW = \sum_{z=1}^{m} H(w_z) \tag{12}$$

其中，m表示注入的故障数量，w_z表示由第z个故障产生的中间状态值，$H(w_z)$表示中间状态值为w_z的汉明重量值。

汉明重量、极大似然估计等区分器由Fuhr等[31]提出并结合统计分析攻击了分组密码AES。本文针对LEA选取上述经典区分器并结合统计分析攻击该算法时，实验表明，存在中间状态值的实际分布与理论分布拟合程度低，导致所需注入的故障数较多，破译该算法所需时间较长。因此，本文通过多轮实验对比不同区分器的筛选效果，最终提出并使用新型的Hellinger距离区分器、Hellinger距离–汉明重量区分器和Hellinger距离–极大似然区分器对LEA进行分析。实验结果显示，新型HD系列区分器可以将所需注入的故障数降至最低72个，破译所需时间降至最快100.3 s，更适用于在倒数第3轮注入故障时的LEA的分析。

2.7.2　新型区分器

1) Hellinger距离

Hellinger距离概念自提出以来，被广泛应用于信息论、统计学和机器学习等领域。在计算机科学领域，Hellinger距离常用于衡量实际观测到的数据分布与理论预期的分布之间的差异大小。Hellinger距离的表达式为

$$H(P_r,P_e) = \frac{1}{\sqrt{2}} \sqrt{\sum_{a=0}^{1} (\sqrt{P_r(a)} - \sqrt{P_e(a)})^2} \tag{13}$$

其中，$P_r(a)$表示a取0和1出现的实际概率，$P_e(a)$表示a取0和1出现的理论概率。Hellinger距离的值越小，则计算出的实际分布与理论分布之间越接近。在HD值取最小值时，对应的候选密钥即为正确密钥。

2) Hellinger距离–汉明重量

HD-HW区分器结合了Hellinger距离与汉明重量二者的优点。首先，使用Hellinger距离区分器缩小候选密钥搜索范围，将较大的Hellinger距离值对应的候选密钥暂存，筛选掉较小值对应的部分候选密钥。正确密钥所对应的中间状态值是剩余的候选密钥中使用汉明重量区分器计算而来的汉明重量值最小的一个。

3) Hellinger距离–极大似然

HD-ML区分器结合了Hellinger距离与极大似然估计二者的优点。首先，使用Hellinger距离区分

器缩小候选密钥搜索范围，将较大的Hellinger距离值对应的候选密钥筛除，暂存较小值对应的部分候选密钥。正确密钥所对应的中间状态值是剩余的候选密钥中使用极大似然区分器计算而来的ML值最大的一个。表4给出了本文中所使用到的区分器的取值范围和筛选过程。

3　实验分析

3.1　实验过程

本文使用CPU为Intel(R) Core(TM) i7–9750H的PC端配置，并使用C++编程语言通过软件模拟实现LEA的代数统计故障分析过程。实验使用固定的原始密钥和随机产生的明文，在该算法的倒数第3轮的首个半字节注入随机4 bit故障，收集产生的错误密文。将收集的错误密文和枚举的2^{18}个候选密钥，根据LEA的解密流程，分别倒推至注入故障位置得到2^{18}个中间状态值。再选取本文所提出的HD区分器，计算每组中间状态值与理论中间状态值的HD，筛选出较小HD值对应的候选密钥。随后，利用研究分析出的代数变换关系进一步过滤候选密钥，若剩余的候选密钥范围在2^8内且其中包含原始密钥，则判定实验成功。在10 000次实验中，HD区分器成功了9 992次，失败了8次，成功率可达到99.9%。破译所需时间最短为106.3 s，破译成功所需故障数最少为85个，其他区分器的实验过程同上述过程。

3.2　实验结果和指标

3.2.1　故障数

故障数是指攻击者使用区分器以最大概率恢复密钥所需的最少故障注入数量。因此，攻击者注入的故障数越少，攻击者所使用的攻击方法的有效性和威胁性越高。如图5所示，使用代数统计故障分析方法恢复128 bit原始密钥，GF、ML、HW、HD、HD-HW和HD-ML区分器分别需要220、180、152、85、78和72个故障，其中，新型单区分器所需故障数较少，新型区分器HD-ML所需故障数最少。

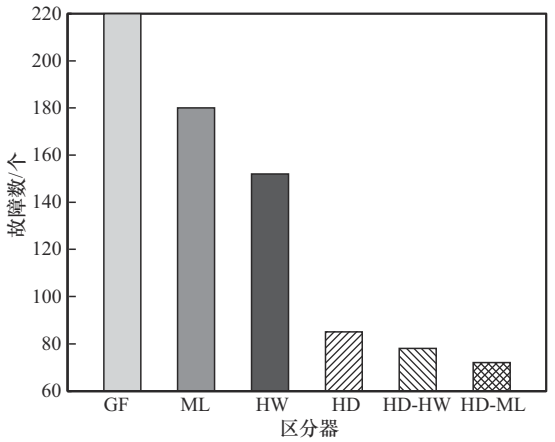

图5　代数统计故障分析方法使用各区分器恢复原始密钥的故障数

3.2.2　耗时

耗时是评估所有故障分析方法攻击效率的另一关键参数，指攻击者从故障注入到成功恢复原始密钥所花费的时间。攻击者用越短的耗时攻击密码算法，在实际应用环境下被检测到的风险就越低，同时反映攻击者所采用的攻击方法的高效性。图6展示了代数统计故障分析方法使用传统区分器和新型区分器恢复原始密钥的累积耗时，其中，横坐标为故障数，纵坐标为耗时。在GF、ML、HW、HD、HD-HW和HD-ML区分器以99.0%及以上的成功率恢复LEA原始密钥的情况下，耗时分别为506.2 s、390.4 s、221.1 s、106.3 s、105.6 s和100.3 s，其中，新型区分器比经典区分器的耗时更少。

表4　各区分器的取值范围和筛选过程

区分器	取值范围	筛选过程
GF	最小值	评估中间状态的出现概率，选择概率最小的统计样本
ML	最大值	评估中间状态的出现概率，选择概率最大的统计样本
HW	最小值	评估中间状态的汉明重量，选择汉明重量值最小的统计样本
HD	最小值	评估中间状态的出现概率，选择概率最小的统计样本
HD-HW	HD最小值 HW最小值	先使用HD区分器选择值最小的统计样本，再使用HW区分器选择出概率最小的统计样本
HD-ML	HD最小值 HW最大值	先使用HD区分器选择值最小的统计样本，再使用ML区分器选择出概率最大的统计样本

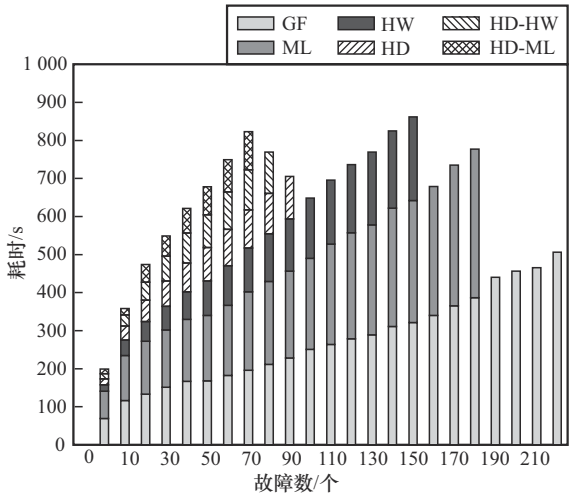

图6　代数统计故障分析使用各区分器恢复原始密钥的累积耗时

3.2.3　成功率

成功率是衡量攻击者攻击密码算法有效性的重要指标之一，指攻击者成功破译算法密钥的概率。攻击者破译密码的成功率越高，则所恢复的原始密钥置信度越高，反映了攻击实施的高可靠性。由于LEA中模加和异或操作的特性，且需要统计的中间状态值只有“0”和“1”2种情况。因此，单个中间状态值会对应多个候选密钥值，即所获得的候选密钥并非唯一确定。综上，本文实验将剩余候选密钥个数不超过256且其中包含正确密钥的情况计入成功率。图7为代数统计故障分析方法使用不同区分器在不同故障数下恢复原始密钥所对应的成功率，其中，横坐标为注入的故障数，纵坐标为破译原始密钥的成功率。如图7所示，GF、ML、HW、HD、HD-HW和HD-ML区分器均可破译LEA，成功率达到99.0%及以上。本文提出的新型HD、HD-HW和HD-ML区分器的表现均优于传统区分器。

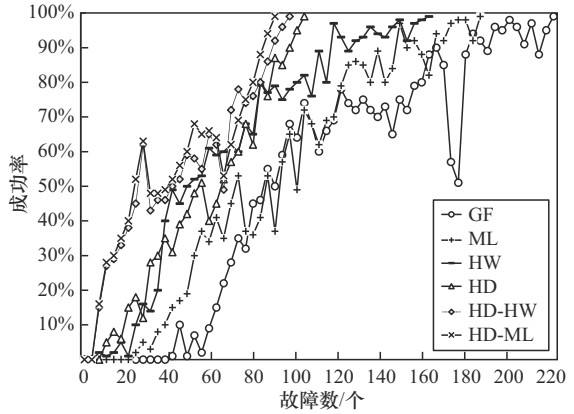

图7　代数统计故障分析方法使用各区分器恢复原始密钥的成功率

3.2.4　复杂度

时间复杂度、数据复杂度和存储复杂度是评估攻击方法效能及安全性的重要参数，分别指攻击密码算法时所需的时间、数据量和存储空间，同时反映了实施攻击的资源需求和效率。时间复杂度的表达式为

$$m2^n g \tag{14}$$

数据复杂度的表达式为

$$m2^n \tag{15}$$

存储复杂度的表达式为

$$m2^n s \tag{16}$$

其中，m为恢复原始密钥所需注入的故障数量，n为候选密钥的比特数，2^n为候选密钥的个数且$n = 18$，g为不同区分器本身的复杂度，s为所需存储的密文比特数。以HD为例，由实验数据可知，当HD区分器达到99.0%及以上成功率时，破译主密钥需要注入85个故障，因此，HD区分器的时间复杂度为

$$m2^n g = 85 \times 2^{18} \times 18 \approx 2^{28.59} \tag{17}$$

表5给出了3种经典区分器和3种新型区分器分别恢复LEA原始密钥成功率达到最大概率时所需的时间、存储和数据复杂度。由表中结果可知，本文提出的新型HD系列区分器相较于经典区分器中表现最优的HW区分器，均降低了时间、存储和数据复杂度，其中，HD-ML表现最优。

表5　区分器恢复128 bit原始密钥的复杂度分析

区分器	时间复杂度	存储复杂度	数据复杂度
GF	$2^{29.95}$	$2^{35.78}$	$2^{25.78}$
ML	$2^{29.66}$	$2^{35.49}$	$2^{25.49}$
HW	$2^{29.42}$	$2^{35.25}$	$2^{25.25}$
HD	$2^{28.59}$	$2^{34.42}$	$2^{24.42}$
HD-HW	$2^{28.49}$	$2^{34.32}$	$2^{24.32}$
HD-ML	$2^{28.34}$	$2^{34.17}$	$2^{24.17}$

综合以上所有指标，针对LEA，新提出的代数统计故障分析方法比现有统计故障分析方法表现更佳。新型HD系列区分器相较于已有区分器也更具优势。

4　结束语

本文针对LEA，在结合统计分析和代数分析的

基础上，提出了代数统计故障分析方法。与经典的统计故障分析方法相比，新型代数分析方法可以使故障注入轮数加深一轮，提出的新型区分器能够在耗时和故障数方面有所优化，并可以更低成本破译LEA。因此，代数统计故障分析方法对LEA的安全性构成威胁。在使用LEA时，建议在密码设备周围部署抗篡改传感器和环境异常监测电路，以减少其受到该类攻击的威胁。后续工作将尝试设计基于唯密文攻击假设的自动化攻击方法，对LEA等轻量级密码进行安全性研究，并重点探索代数统计故障分析与功耗分析等传统方法的协同机制，研究其降低综合成本的可行性。

参考文献：

[1] TIMKO D, SHARKO M, LI Y Y. Security analysis of wearable smart health devices and their companion apps[C]//Proceedings of the 2024 IEEE Security and Privacy Workshops (SPW). Piscataway: IEEE Press, 2024: 274-280.

[2] GAO Y M, ZHOU T Q, ZHENG W Y, et al. High-availability authentication and key agreement for Internet of things-based devices in industry 5.0[J]. IEEE Transactions on Industrial Informatics, 2024, 20(12): 13571-13579.

[3] LI W J, GLEERUP T, TAN J, et al. A security enhanced Android unlock scheme based on pinch-to-zoom for smart devices[J]. IEEE Transactions on Consumer Electronics, 2024, 70(1): 3985-3993.

[4] GUO Y, LI L, LIU B T. Shadow: a lightweight block cipher for IoT nodes[J]. IEEE Internet of Things Journal, 2021, 8(16): 13014-13023.

[5] 何乐生, 冯毅, 岳远康, 等. 针对物联网设备的旁路攻击及防御方法的研究[J]. 通信学报, 2025, 46(2): 166-175.
HE L S, FENG Y, YUE Y K, et al. Research on side-channel attacks and defense methods for IoT devices[J]. Journal on Communications, 2025, 46(2): 166-175.

[6] 肖冲, 唐明. 基于深度学习的侧信道分析综述[J]. 计算机学报, 2025, 48(3): 694-720.
XIAO C, TANG M. A survey on deep learning-based side-channel analysis[J]. Chinese Journal of Computers, 2025, 48(3): 694-720.

[7] 吴文玲, 王博琳. 新形态对称密码算法研究[J]. 密码学报, 2024, 11(1): 128-144.
WU W L, WANG B L. Research on new morphologic symmetric cryptographic algorithms[J]. Journal of Cryptologic Research, 2024, 11(1): 128-144.

[8] HONG D, LEE J K, KIM D C, et al. LEA: a 128-bit block cipher for fast encryption on common processors[C]//International Workshop on Information Security Applications. Berlin: Springer International Publishing, 2014: 3-27.

[9] ZHANG K, GUAN J, HU B. Zero correlation linear cryptanalysis on LEA family ciphers[J]. Journal of Communications, 2016, 11(7): 677-685.

[10] 李航, 任炯炯, 陈少真. 减轮LEA密码算法的积分攻击[J]. 电子学报, 2020, 48(1): 17-27.
LI H, REN J J, CHEN S Z. Integral attack on reduced-round LEA cipher[J]. Acta Electronica Sinica, 2020, 48(1): 17-27.

[11] 李艳俊, 李寅霜, 刘健, 等. 约减轮数分组密码LEA的差分分析[J]. 电子与信息学报, 2023, 45(10): 3737-3744.
LI Y J, LI Y S, LIU J, et al. Differential analysis of reduced rounds block cipher LEA[J]. Journal of Electronics & Information Technology, 2023, 45(10): 3737-3744.

[12] CHEN Y, BAO Z Z, YU H B. Differential-linear approximation semi-unconstrained searching and partition tree: application to LEA and speck[C]//International Conference on the Theory and Application of Cryptology and Information Security. Berlin: Springer, 2023: 223-255.

[13] BONEH D, DEMILLO R A, LIPTON R J. On the importance of checking cryptographic protocols for faults[C]//International Conference on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 1997: 37-51.

[14] 张国双, 陈晓, 王安, 等. 面向ACORN v3消息认证码的随机差分故障分析[J]. 密码学报, 2021, 8(3): 498-520.
ZHANG G S, CHEN X, WANG A, et al. Random differential fault attack for ACORN v3 message authentication code[J]. Journal of Cryptologic Research, 2021, 8(3): 498-520.

[15] PAL D, ALI M R, DAS A, et al. A cluster-based practical key recovery attack on reduced-round AES using impossible-differential cryptanalysis[J]. The Journal of Supercomputing, 2023, 79(6): 6252-6289.

[16] QIU Z Z, ZHANG F, FENG T X, et al. RAFA: redundancies-assisted algebraic fault analysis and its implementation on SPN block ciphers[J]. IACR Transactions on Cryptographic Hardware and Embedded Systems, 2023: 570-596.

[17] AHMADI S, AREF M R. Generalized meet in the middle cryptanalysis of block ciphers with an automated search algorithm[J]. IEEE Access, 2019, 8: 2284-2301.

[18] BEIGIZAD A A, SOLEIMANY H, ZAREI S, et al. Linked fault analysis[J]. IEEE Transactions on Information Forensics and Security, 2023, 19: 632-645.

[19] SALAM I, ALAWATUGODA J, MADUSHAN H. Statistical fault analysis of TinyJambu[J]. Discover Applied Sciences, 2024, 6(2): 55.

[20] PAPADIMITRIOU A, NOMIKOS K, PSARAKIS M, et al. You can detect but you cannot hide: fault assisted side channel analysis on protected software-based block ciphers[C]//Proceedings of the 2020 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT). Piscataway: IEEE Press, 2020: 1-6.

[21] LIM S, LEE J, HAN D G. Improved differential fault attack on LEA by algebraic representation of modular addition[J]. IEEE Access, 2020, 8: 212794-212802.

[22] 张金煜, 张雨希, 李玮. 轻量级密码LEA的唯密文故障分析[J]. 东华大学学报(自然科学版), 2023, 49(6): 135-141.
ZHANG J Y, ZHANG Y X, LI W. Ciphertext-only fault analysis on the LEA lightweight cipher[J]. Journal of Donghua University (Natural Science), 2023, 49(6): 135-141.

[23] BALASCH J, GIERLICHS B, VERBAUWHEDE I. An in-depth and black-box characterization of the effects of clock glitches on 8-bit MCUs[C]//Proceedings of the 2011 Workshop on Fault Diagnosis and Tolerance in Cryptography. Piscataway: IEEE Press, 2011: 105-114.

[24] COURTOIS N, KLIMOV A, PATARIN J, et al. Efficient algorithms for

solving overdefined systems of multivariate polynomial equations[C]// International Conference on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 2000: 392-407.

[25] COURTOIS N T, PIEPRZYK J. Cryptanalysis of block ciphers with overdefined systems of equations[C]//International Conference on the Theory and Application of Cryptology and Information Security. Berlin: Springer, 2002: 267-287.

[26] BARD G V, COURTOIS N T, NAKAHARA JR J, et al. Algebraic, AIDA/cube and side channel analysis of KATAN family of block ciphers[C]//Progress in Cryptology-INDOCRYPT 2010. Berlin: Springer, 2010: 176-196.

[27] 关杰, 张中亚 . 5 轮 Salsa20 的代数-截断差分攻击[J]. 软件学报, 2013, 24(5): 1111-1126.
GUAN J, ZHANG Z Y. Algebraic truncated differential cryptanalysis of 5-round Salsa20[J]. Journal of Software, 2013, 24(5): 1111-1126.

[28] 黄静, 赵新杰, 张帆, 等 . PRESENT 代数故障攻击的改进与评估[J]. 通信学报, 2016, 37(8): 144-156.
HUANG J, ZHAO X J, ZHANG F, et al. Improvement and evaluation for algebraic fault attacks on PRESENT[J]. Journal on Communications, 2016, 37(8): 144-156.

[29] LE D P, YEO S L, KHOO K. Algebraic differential fault analysis on SIMON block cipher[J]. IEEE Transactions on Computers, 2019, 68(11): 1561-1572.

[30] LI Z N, WU B F, LIN D D. Algebraic-differential attacks on a family of arithmetization-oriented symmetric ciphers[J]. Journal of Systems Science and Complexity, 2023, 36(6): 2681-2702.

[31] FUHR T, JAULMES E, LOMNÉ V, et al. Fault attacks on AES with faulty ciphertexts only[C]//Proceedings of the 2013 Workshop on Fault Diagnosis and Tolerance in Cryptography. Piscataway: IEEE Press, 2013: 108-118.

[32] NOZAKI Y, ASAHI K, YOSHIKAWA M. Statistical fault analysis for a lightweight block cipher TWINE[C]//Proceedings of the 2015 IEEE 4th Global Conference on Consumer Electronics (GCCE). Piscataway: IEEE Press, 2015: 477-478.

[33] HARMOUCH Y, KOUCH R E. A statistical analysis for high-speed stream ciphers[C]//Innovations in Bio-Inspired Computing and Applications. Berlin: Springer, 2018: 339-349.

[34] 李玮, 吴益鑫, 谷大武, 等 . SIMON 轻量级密码算法的唯密文故障分析[J]. 通信学报, 2019, 40(11): 122-137.
LI W, WU Y X, GU D W, et al. Ciphertext-only fault analysis of the SIMON lightweight cipher[J]. Journal on Communications, 2019, 40(11): 122-137.

[35] 李玮, 朱晓铭, 谷大武, 等 . PRESENT 轻量级密码的中间相遇统计故障分析[J]. 计算机学报, 2023, 46(2): 353-370.
LI W, ZHU X M, GU D W, et al. Meet-in-the-middle statistical fault analysis of the PRESENT lightweight cryptosystem[J]. Chinese Journal of Computers, 2023, 46(2): 353-370.

[36] JAP D, BREIER J. Differential fault attack on LEA[C]//Information and Communication Technology. Berlin: Springer, 2015: 265-274.

[37] PARK M, KIM J. Differential fault analysis of the block cipher LEA[J]. Journal of the Korea Institute of Information Security & Cryptology, 2014, 24(6): 1117-1127.

[38] FISHER R A. On the mathematical foundations of theoretical statistics[C]// Breakthroughs in Statistics: Foundations and Basic Theory. ACM Press: New York, 1992: 11-44.

[39] REED I. A class of multiple-error-correcting codes and the decoding scheme[J]. Transactions of the IRE Professional Group on Information Theory, 1954, 4(4): 38-49.

[作者简介]

李玮（1980– ），女，安徽寿县人，博士，东华大学教授、博士生导师，主要研究方向为对称密码的设计与分析。

张爱琳（2001– ），女，吉林四平人，东华大学硕士生，主要研究方向为轻量级分组密码的故障分析。

王弈（1974– ），女，浙江宁波人，博士，华东政法大学教授、博士生导师，主要研究方向为信息安全和信息法学。

谷大武（1970– ），男，河南漯河人，博士，上海交通大学教授、博士生导师，主要研究方向为密码学和计算机安全。

李诗雅（2002– ），女，河南柘城人，东华大学硕士生，主要研究方向为对称密码的故障分析。