COSADE 2013
Paris, France
May 7-8, 2013

Improved Algebraic Fault Analysis: A Case Study on Piccolo and Applications to Other Lightweight Block Ciphers

Fan (Terry) ZHANG[1], Xinjie ZHAO[2,3], Shize GUO[3], Tao WANG[2], Zhijie (Jerry) SHI[1]

1 University of Connecticut, USA
2 Ordnance Engineering College, China
3 The Institute of North Electronic Equipment, China

Outline

1. Algebraic Fault Analysis (AFA)
2. A Case Study of AFA on Piccolo
3. Applications to Other Lightweight Block Ciphers
4. Conclusion and Future Work

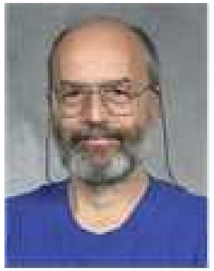
1 Algebraic Fault Analysis (AFA)

1.1 Motivations

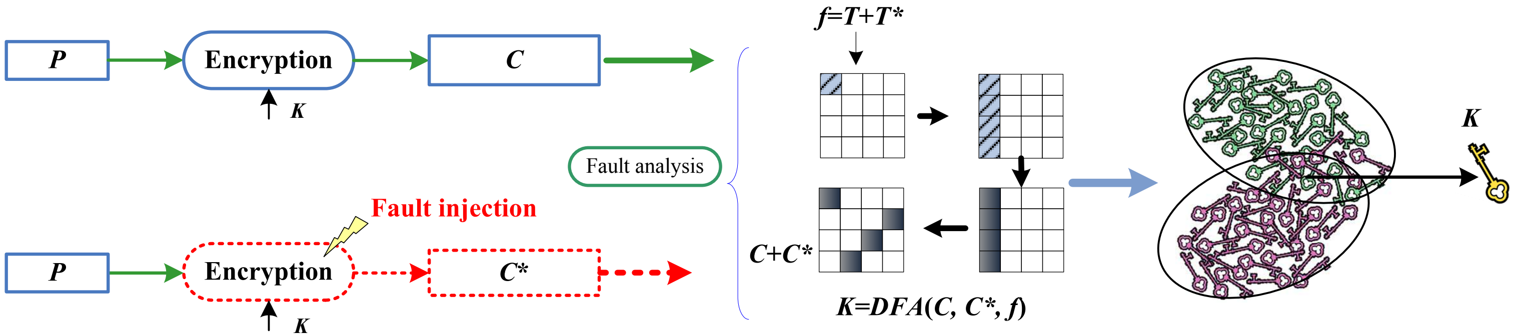
D. Boneh

Fault attack: "The cryptographic cipher has to be implemented on a device and deployed in the real world. The device performing the computations may introduce errors, which can enable a malicious adversary to inject and analyze faults for key recovery", with application to on RSA-CRT, 1996.

Differential Fault attack (DFA) on DES, 1997.



A. Shamir



DFA requires manually analysis fault propagation path, can we find out an automatic way for this?

1 Algebraic Fault Analysis (AFA)

1.1 Motivations

Inspired from Algebraic Side-Channel Attacks in 2009, AFA was proposed in 2010.

1 Inducing the faults

2 Constructing the equations for the cipher

3 Constructing the equations for the faults

4 Solving the equation system

1. AFA is an automatic and generic technique for fault attacks.

1.1 Motivations

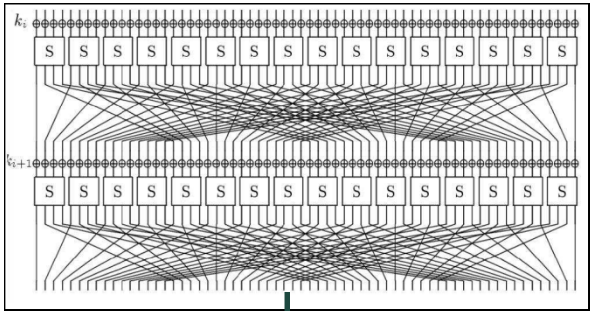
Our world need lightweight block ciphers, Many ciphers are proposed in recent years!

2007: PRESENT、 DESXL ;   2008: PUFFIN ;

2009: MIBS、KTANTAN ;   2010: PRINTCipher, GOST;

2011: Klein, Piccolo, LED;   2012: TWINE etc

lightweight design



Simple algebraic structure

lightweight implementation

Vulnerable to fault injection

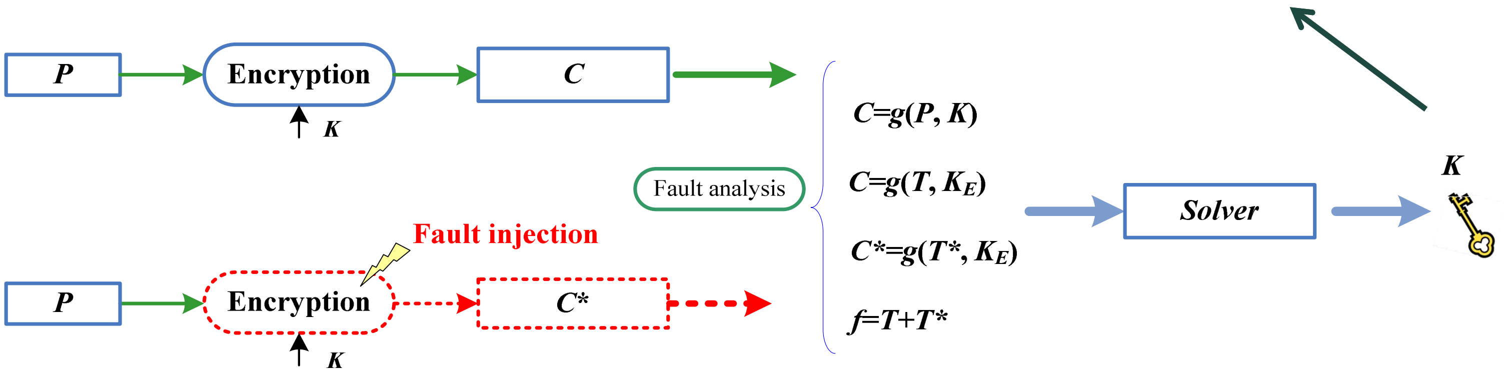
2. Lightweight block ciphers are vulnerable to AFA.

1.2 Algebraic Fault Attack (AFA)

First proposal, 2010

N. Courtois

Algebraic Fault Attack on DES with a single fault



1) 2 bits in the 13th round are altered
2) Guessing 24 key bits.

1) Each guess, 0.01 hour.
2) Full attack, $2^{17.35}$ hours.
3) 10 times faster than brute force.

Subsequent work till now

Mohamed applied AFA on Trivium in COSADE 2011.

Jovanovic applied AFA on LED in SCC 2012.

1 fault, 14.67 hours

1 Algebraic Fault Analysis (AFA)

1.2 Algebraic Fault Attack (AFA)

Open problem

- How to conduct more **efficient AFA**?
- What are the **advantages of AFA over DFA**?
- What are the threats of **AFA to lightweight block ciphers**?

2 A Case Study of AFA on Piccolo

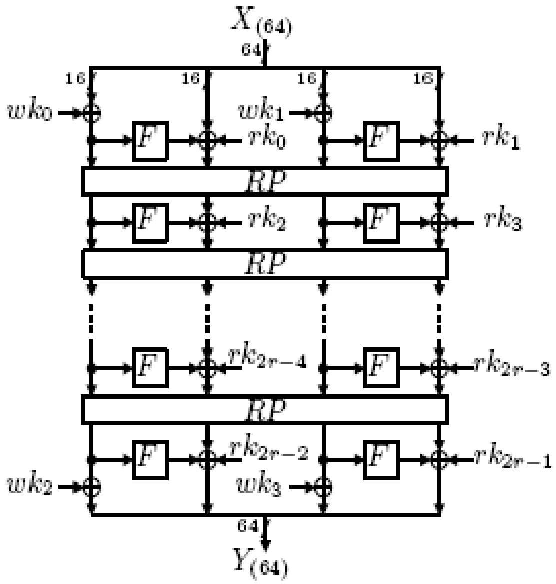
2.1 Overview of Piccolo

A lightweight block cipher introduced in CHES 2011

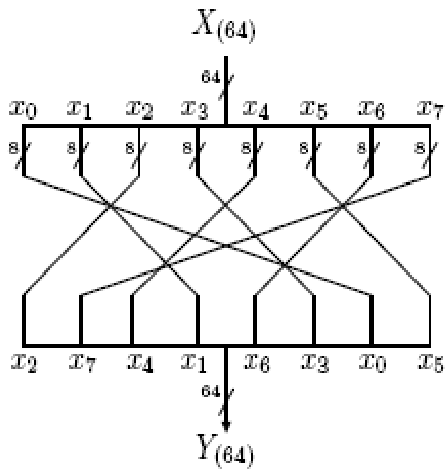
- 64-bit block cipher, small and fast like PRESENT
- uses the Feistel structure.
- uses 80 or 128 bit keys (Piccolo-80, Piccolo-128),
- uses 25 rounds for Piccolo-80 and 31 rounds for Piccolo-128
- simple key schedule which XORed key with many 16-bit constants

2.1 Overview of Piccolo

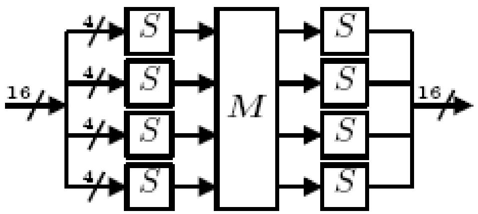
Specification of Piccolo



(a) Encryption function



(b) Round permutation



(c) F-function

$$M = \begin{pmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{pmatrix}.$$

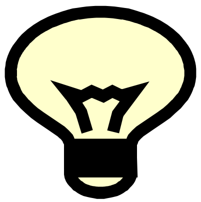
(d) MixColumn matrix

2 A Case Study of AFA on Piccolo

2.2 Related work

K. Jeong. Differential Fault Analysis on Block Cipher Piccolo. Cryptology ePrint Archive, available at http://eprint.iacr.org/2012/399.pdf, 2012.

◆random byte fault model in the penultimate (24th) round
◆DFA technique
◆Piccolo-80， six fault injections； Piccolo-128, eight fault injections

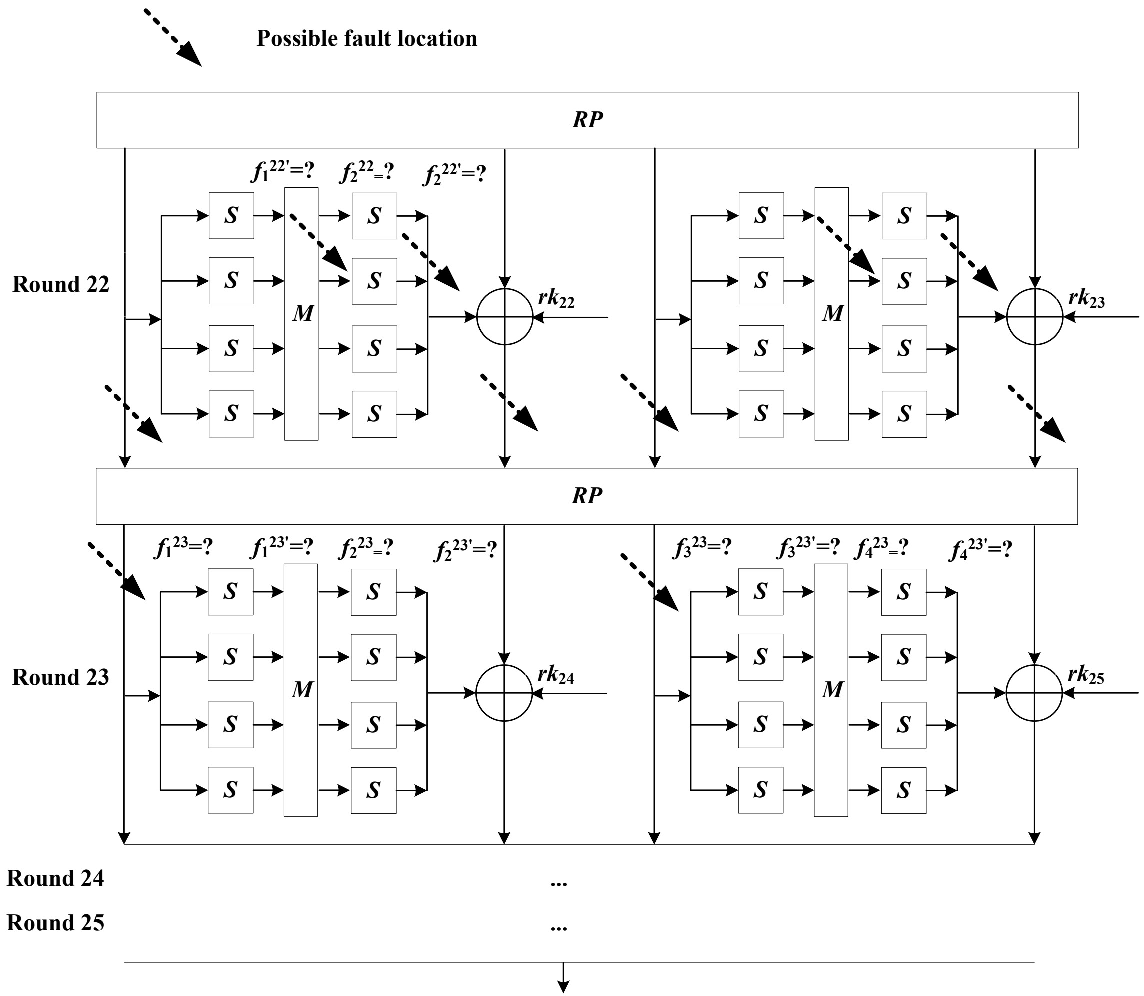


Can we break Piccolo with less fault injections use AFA?

2 A Case Study of AFA on Piccolo

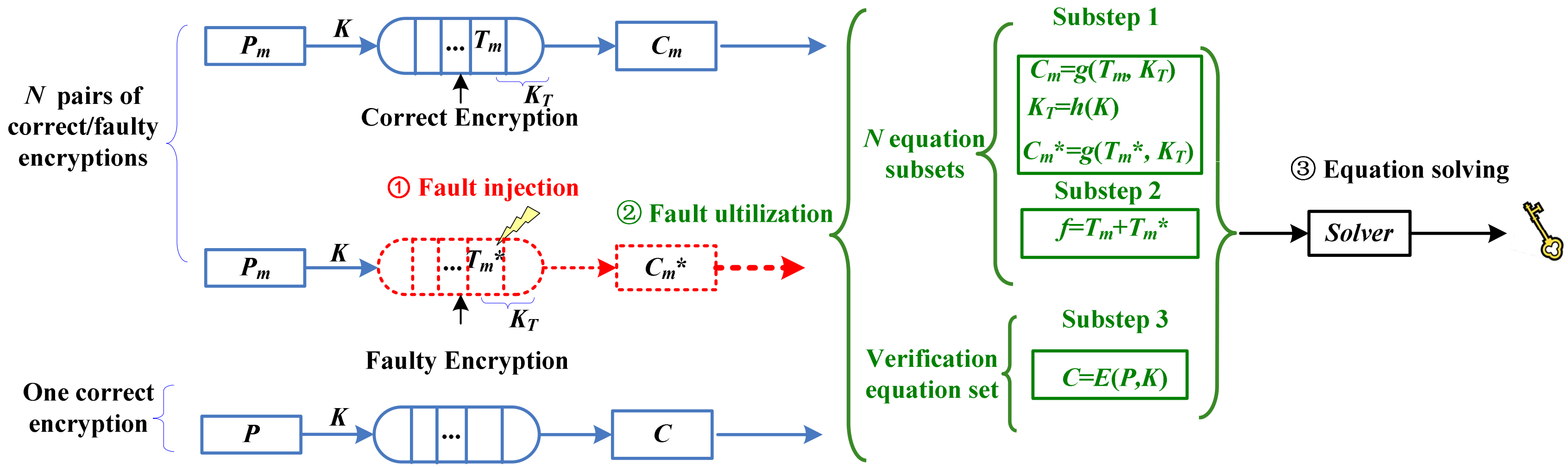
2.3 Fault model of our work

Single nibble fault injected to the 23-rd round, deeper than previous work



2.4 Improved AFA

1. Framework of AFA

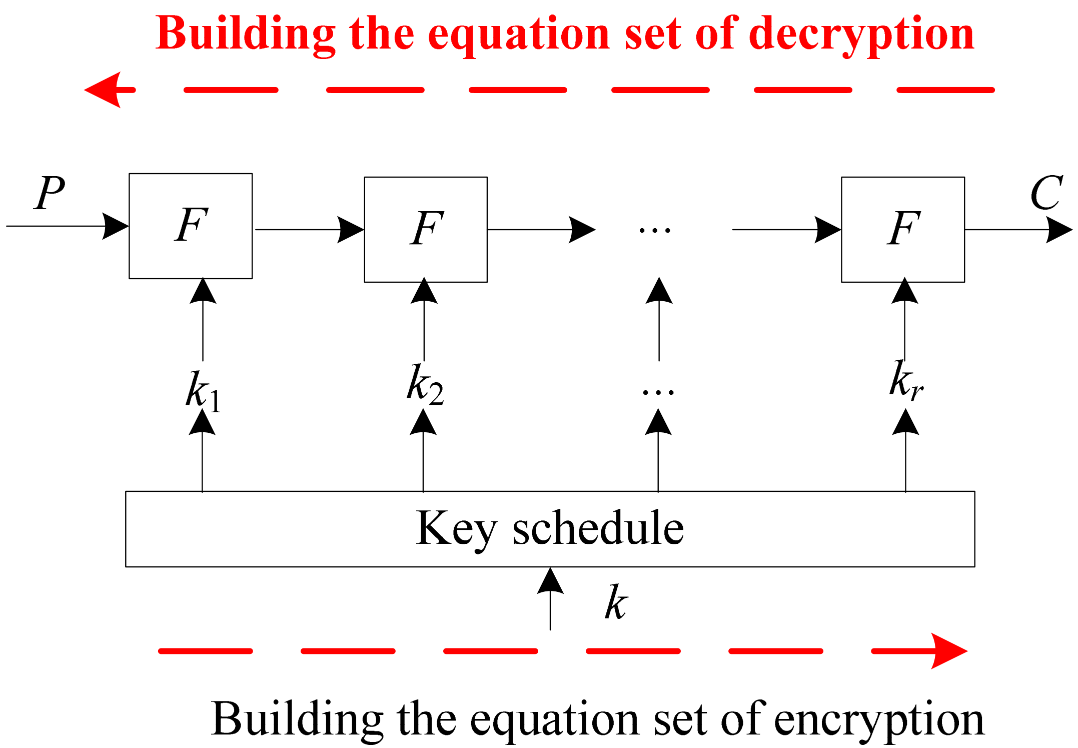


How to build the equation set for ciphers which is suitable for efficient AFA?

How to represent the fault model when the exact fault locations are unknown?

2.4 Improved AFA

1. build the equation set for the decryption of Piccolo.



2.4 Improved AFA

2. Build the equation set for fault when its location is unknown.

1 Correct state: $x_1||x_2||\ldots x_{16}$, faulty state injected with nibble (4-bit) fault: $y_1||y_2||\ldots y_{16}$

2 fault difference $Z = z_1 \,||\, z_2 \,||\ldots z_{16},\ \ z_i = x_i \oplus y_i,\ 1 \le i \le 16$

3 *Z can be divided into* four parts, $Z_1||Z_2||Z_3||Z_4$， $Z_i = z_{4i-3}||\, z_{4i-2}||\, z_{4i-1}||\, z_{4i}$ $(1 \le i \le 4)$

4 u_i denotes whether Z_i is injected with faults. $u_i=0$ means fault is injected to Z_i.

$$u_i = (1+z_{4i-3})(1+z_{4i-2})(1+z_{4i-1})(1+z_{4i}),\ 1 \le i \le 4$$

5 Only one nibble becomes faulty， only one of $u_0||u_1||u_2||u_3$ is zero.

$$(1+u_0) \vee (1+u_1) \vee (1+u_2) \vee (1+u_3) = 1,\ u_i \vee u_j = 1,\ 1 \le i < j \le 4$$

AFA does not need to deduce the accurate fault location as in DFA.

2.5 Experimental results

Attack Setup

01 Fault injection

Software simulation via VC++ 6.0 by modifying the source code of Piccolo.

02 Solver setup

CryptoMiniSAT v2.9.4..

03 Computing setup

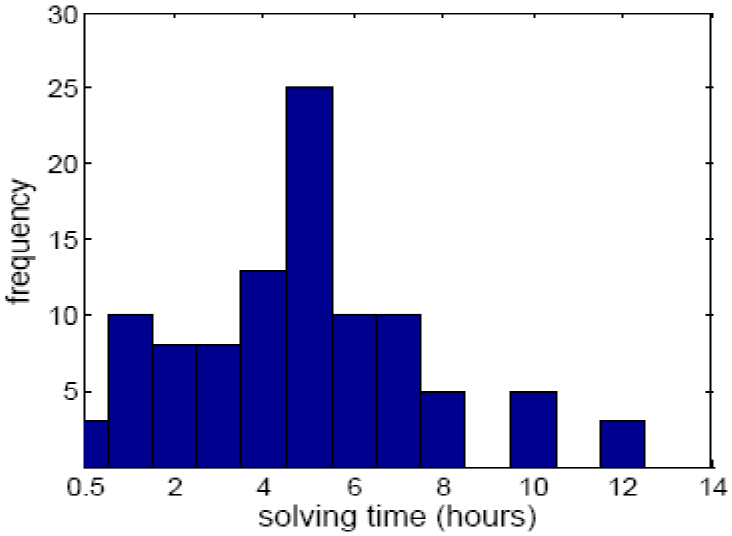
Intel(R) Core(TM) I7-2640M, 2.80 GHZ, 4G Memory, Windows XP 64-bit OS

2.5 Experimental results

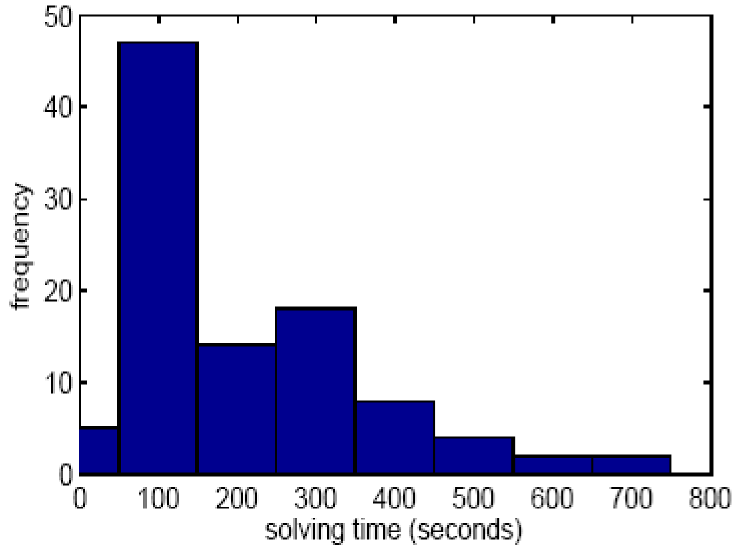
Single fault injection in the 23-th round

1) Full encryption set: 18,317 variables, 30,112 ANF equations, 580K script size, the solver can not output the solution within 48 hours

2) Full decryption set: 17,129 variables, 28,016 ANF equations, 553K script size , the attack can succeed.



(a) with one fault



(b) with two faults

One fault: 5 hours on average

Two faults: less than 700 seconds

2 A Case Study of AFA on Piccolo

2.5 Experimental results

Extensions

Different Fault Widths

One fault, 8-bit (6 hours), 16-bit (22 hours)

Different Fault Positions

Two faults, 22-nd round (5-6 hours)

Piccolo-128

Three faults, less than 0.5 hours

3.1 AFA on AES

Table 6. Results of AFA on AES

Attack	Block cipher	Fault model	Technique	Faults	Time
[29]	AES-128	n_w=8,n_c=7	DFA	1	2^{32} encryption
[38]	AES-128	n_w=8,n_c=7	DFA	1	50 minutes
[1]	AES-128	n_w=8,n_c=7	DFA	1	5 minutes
[8]	AES-128	n_w=8,n_c=7	AFA	1	1 second
This paper	AES-128	n_w=8,n_c=7	AFA	1	10 hours

our SAT-based AFA is less efficient than DFA in [29] and [38]

1) The algebraic structure of AES (especially the 8×8 S-box) is complicated.

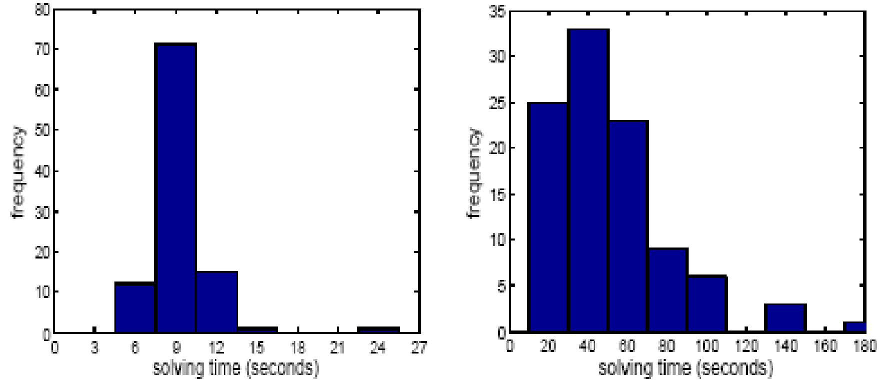
2) The second is that the solver used is not customized for fault attacks on AES, as in [8].

3.2 AFA on DES

Table 6. Results of AFA on DES

Attack	Block cipher	Fault model	Technique	Faults	Time
[5]	DES	n_w=1,n_c=14,15,16	DFA	3	—
[10]	DES	n_w=2,n_c=14	AFA	2	$2^{13.35}$ hours
[10]	DES	n_w=2,n_c=13	AFA	1	$2^{17.35}$ hours
[33]	DES	n_w=1,n_c=12	DFA	7	–
This paper	DES	n_w=1,n_c=12	AFA	1	10 seconds
[33]	DES	n_w=8,n_c=12	DFA	9	–
This paper	DES	n_w=8,n_c=12	AFA	1	60 seconds
[33]	DES	n_w=1,n_c=11	DFA	11	–
This paper	DES	n_w=1,n_c=11	AFA	1	3000 seconds

Single 1 bit or 8-bit fault injected to the left part of the DES internal state at the end of the 12-th round, a few minutes solving.



(a) DES,n_w=1,n_c=12, n_v=18329,n_a=104073,n_s=2021K

(b) DES,n_w=8,n_c=12, n_v=18350,n_a=105075,n_s=2025K

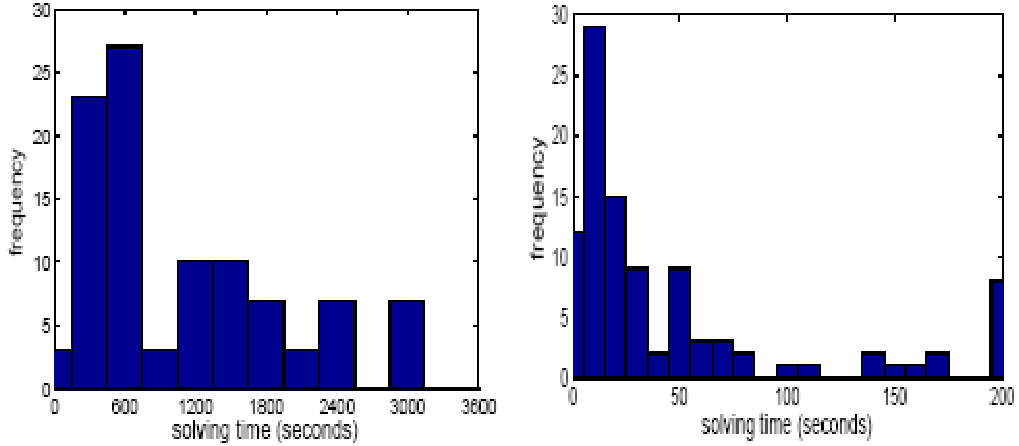
Click to Show the attack!

3.3 AFA on MIBS and LED

Table 6. Results of AFA on MIBS and LED

Attack	Block cipher	Fault model	Technique	Faults	Time
[39]	MIBS-64	n_w=4,n_c=30	DFA	1	60 seconds
This paper	MIBS-64	n_w=4,n_c=29	AFA	1	1100 seconds
[23]	LED-64	n_w=4,n_c=30	AFA	1	14.67 hours
This paper	LED-64	n_w=4,n_c=30	AFA	1	180 seconds

Single fault injection



(c) MIBS,n_w=4,n_c=29, n_v=13505,n_a=20514,n_s=426K

(d) LED,n_w=4,n_c=30, n_v=21131,n_a=35389,n_s=775K

More deeper fault model

More efficient AFA

3 Applications to Other Lightweight Block Ciphers

Lessons learned:

1) AFA requires the least number of faults.

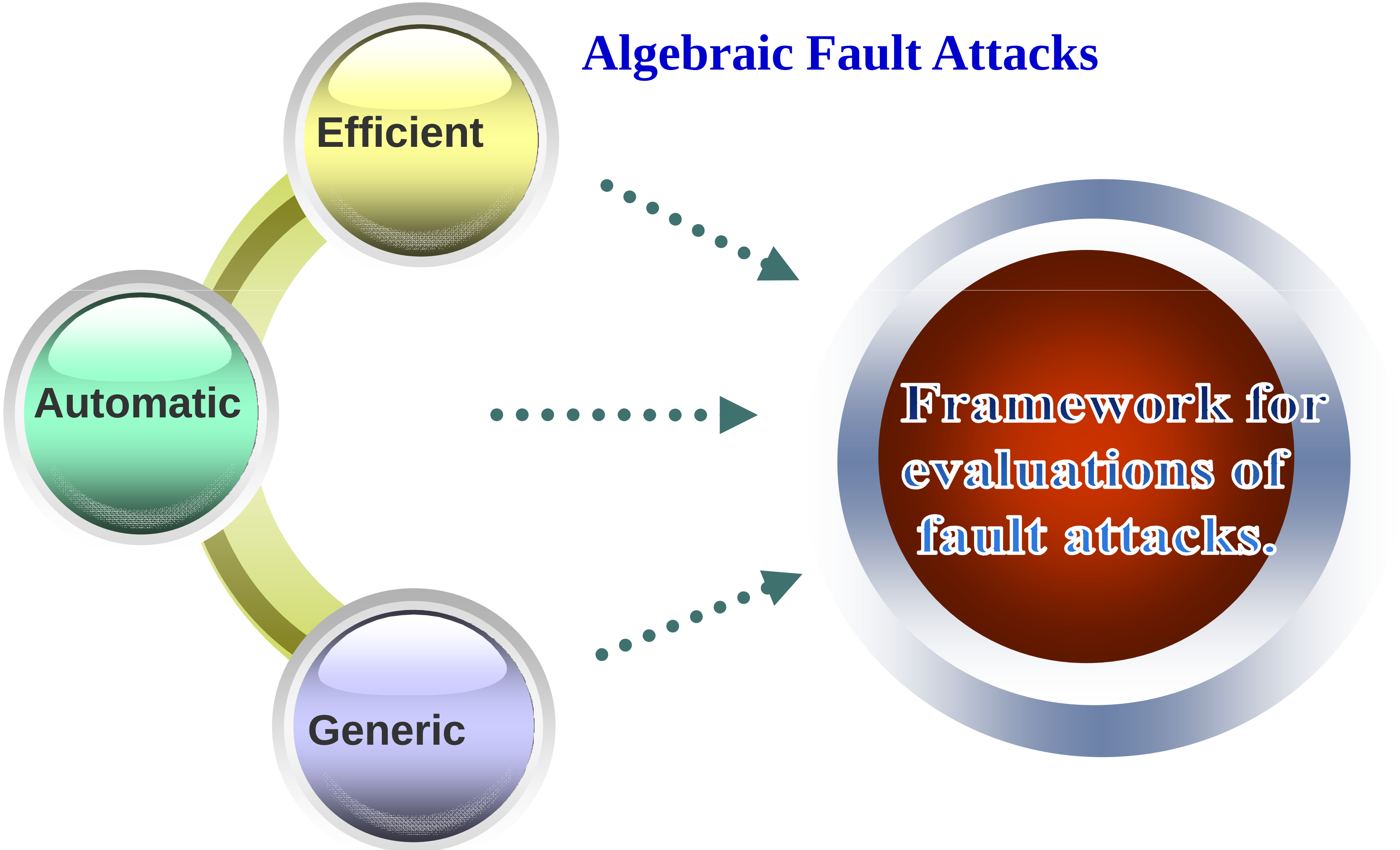
2) The efficiency of AFA depends on the algebraic structure of the cipher and the fault models.

3)The time is short for lightweight ciphers, and longer for block ciphers with complicated algebraic structures such as AES.

4) AFA can be used to improve DFA on lightweight block ciphers.

4 Conclusion and Future Work

4.1 Conclusion



4.2 future work

Improving AFA
- Optimize the equation set
- Optimize the solving strategy

Analyzing AFA
- What are the dependencies of AFA?
- When to use AFA, can AFA replace DFA?

Applying AFA
- Apply to more complicated ciphers
- Generate a universal evaluating tool

Defending AFA
- Design AFA resistant nonlinear function

Thanks!

Q & A

Email： fan.zhang@engineer.uconn.edu

zhaoxinjieem@163.com