

# Revisiting Differential-Linear Attacks via a Boomerang Perspective

Applications to AES, Ascon, CLEFIA, SKINNY, PRESENT, KNOT, TWINE, WARP, LBlock, Simeck, and SERPENT

**Hosein Hadipour**

Patrick Derbez

Maria Eichlseder

CRYPTO 2024 - California, USA



# Research Gap and Our Contributions



## Research Gap

- 🕒 How to analytically estimate the correlation of DL distinguishers?
- 🕒 How to (efficiently) find good DL distinguishers?



## Contributions

- ✅ Generalizing the DLCT framework [Bar+19] for analytical correlation estimation.
- ✅ Introducing an efficient method to search for DL distinguishers applicable to:
  - Strongly aligned SPN primitives: AES, SKINNY
  - Weakly aligned SPN primitives: Ascon, SERPENT, KNOT, PRESENT
  - Feistel structures: CLEFIA, TWINE, LBlock, LBlock-s, WARP
  - AndRX designs: Simeck

# Research Gap and Our Contributions



## Research Gap

- ❑ How to analytically estimate the correlation of DL distinguishers?
- ❑ How to (efficiently) find good DL distinguishers?



## Contributions

- ✔ Generalizing the DLCT framework [Bar+19] for analytical correlation estimation.
- ✔ Introducing an efficient method to search for DL distinguishers applicable to:
  - Strongly aligned SPN primitives: AES, SKINNY
  - Weakly aligned SPN primitives: Ascon, SERPENT, KNOT, PRESENT
  - Feistel structures: CLEFIA, TWINE, LBlock, LBlock-s, WARP
  - AndRX designs: Simeck

# Outline

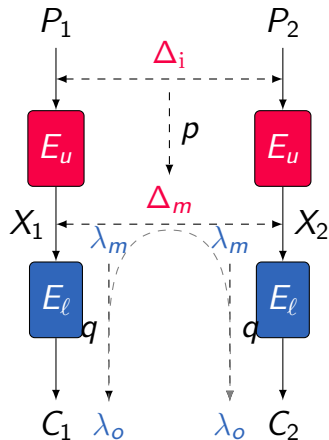
- 1 Background
- 2 Generalized DLCT Framework
- 3 Differential-Linear Switches and Deterministic Trails
- 4 Automatic Tools to Search for DL Distinguishers
- 5 Contributions and Future Works

# Background



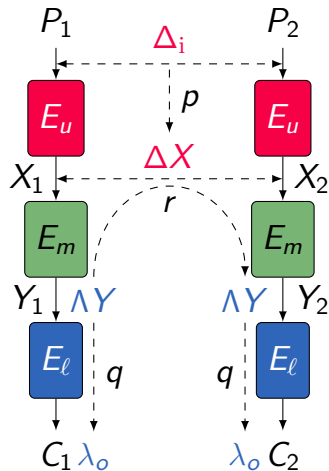
# Differential-Linear (DL) Attack [LH94]

- $\mathbb{P}(\Delta_i \xrightarrow{E_u} \Delta_m) = p$
- $\mathbb{C}(\lambda_m \xrightarrow{E_\ell} \lambda_o) = q$
- Assumptions ( $\Delta X = X_1 \oplus X_2$ ):
  1.  $E_u$ , and  $E_\ell$  are statistically independent
  2.  $\mathbb{P}(\lambda_m \cdot \Delta X = 0) = 1/2$  when  $\Delta X \neq \Delta_m$
- $\mathbb{C}(\lambda_o \cdot \Delta C) = (-1)^{\lambda_m \cdot \Delta_m} \cdot pq^2 = \pm pq^2$



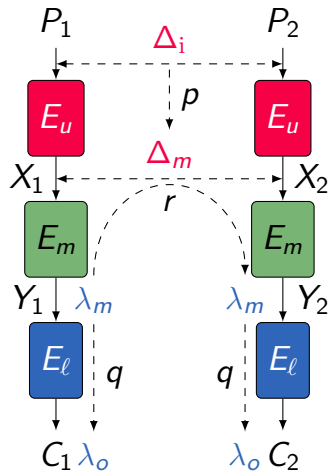
# Sandwich Framework for DL Attack [BLN14; DKS14; Bar+19]

- $\mathbb{R}(\Delta X, \Lambda Y) = \mathbb{C}(\Lambda Y \cdot E_m(X) \oplus \Lambda Y \cdot E_m(X \oplus \Delta X))$
- $\mathbb{C}(\lambda_o \cdot \Delta C) = \sum_{\Delta X, \Lambda Y} \mathbb{P}(\Delta_i, \Delta X) \cdot \mathbb{R}(\Delta X, \Lambda Y) \cdot \mathbb{C}^2(\Lambda Y, \lambda_o)$
- $\mathbb{P}(\Delta_i \xrightarrow{E_u} \Delta_m) = p$
- $\mathbb{R}(\Delta_m, \lambda_m) = r$
- $\mathbb{C}(\lambda_m \xrightarrow{E_\ell} \lambda_o) = q$
- $\mathbb{C}(\lambda_o \cdot \Delta C) \approx prq^2$



# Sandwich Framework for DL Attack [BLN14; DKS14; Bar+19]

- $\mathbb{R}(\Delta X, \Lambda Y) = \mathbb{C}(\Lambda Y \cdot E_m(X) \oplus \Lambda Y \cdot E_m(X \oplus \Delta X))$
- $\mathbb{C}(\lambda_o \cdot \Delta C) = \sum_{\Delta X, \Lambda Y} \mathbb{P}(\Delta_i, \Delta X) \cdot \mathbb{R}(\Delta X, \Lambda Y) \cdot \mathbb{C}^2(\Lambda Y, \lambda_o)$
- $\mathbb{P}(\Delta_i \xrightarrow{E_u} \Delta_m) = p$
- $\mathbb{R}(\Delta_m, \lambda_m) = r$
- $\mathbb{C}(\lambda_m \xrightarrow{E_\ell} \lambda_o) = q$
- $\mathbb{C}(\lambda_o \cdot \Delta C) \approx prq^2$





# Differential-Linear Connectivity Table (DLCT) [Bar+19]

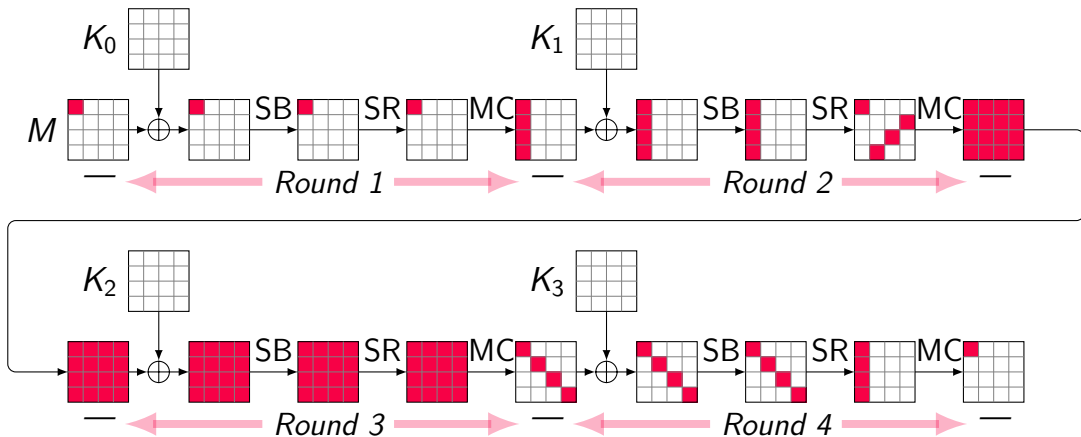


$$\text{DLCT}_b(\Delta_i, \lambda_o) = \{x \in \mathbb{F}_2^n : \lambda_o \cdot S(x) \oplus \lambda_o \cdot S(x \oplus \Delta_i) = b\}$$

$$\text{DLCT}(\Delta_i, \lambda_o) = |\text{DLCT}_0(\Delta_i, \lambda_o)| - |\text{DLCT}_1(\Delta_i, \lambda_o)|$$

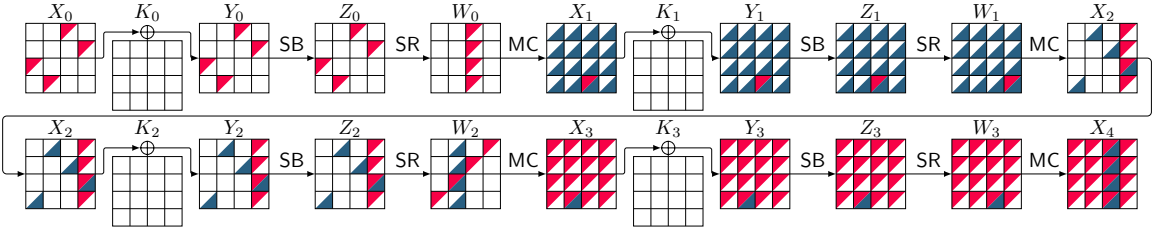
$$\mathbb{C}_{\text{DLCT}}(\Delta_i, \lambda_o) = 2^{-n} \cdot \text{DLCT}(\Delta_i, \lambda_o)$$

# Security of AES Against Differential/Linear Attacks



$$\mathbb{P}_{4 \text{ rounds}} \leq 2^{-150}, \quad \mathbb{C}_{4 \text{ rounds}}^2 \leq 2^{-150}$$

# A 4-round DL Distinguisher for AES



---

$r_u = 1, r_m = 3, r_\ell = 0, \quad p = 2^{-24.00}, \quad r = 2^{-7.66}, q^2 = 1, \quad \mathbb{C} = prq^2 = 2^{-31.66}$	
$\Delta X_0$ 00005200000000f58f000000007b0000	$\Delta X_1$ 000000000000000000000000000000b400
$\Gamma X_4$ 0032000000ab0000000660000000980000	-

---

$2^{63.32}$  v.s.  $2^{150}$

# Generalized DLCT Framework



# Upper Differential-Linear Connectivity Table (UDLCT)



$$\text{UDLCT}_b(\Delta_i, \Delta_o, \lambda_o) = \{x \in \mathbb{F}_2^n : S(x) \oplus S(x \oplus \Delta_i) = \Delta_o \text{ and } \lambda_o \cdot \Delta_o = b\}$$

$$\text{UDLCT}(\Delta_i, \Delta_o, \lambda_o) = |\text{UDLCT}_0(\Delta_i, \Delta_o, \lambda_o)| - |\text{UDLCT}_1(\Delta_i, \Delta_o, \lambda_o)|$$

$$\mathbb{C}_{\text{UDLCT}}(\Delta_i, \Delta_o, \lambda_o) = 2^{-n} \cdot \text{UDLCT}(\Delta_i, \Delta_o, \lambda_o)$$

# Lower Differential-Linear Connectivity Table (LDLCT)



$$\text{LDLCT}_b(\Delta_i, \lambda_i, \lambda_o) = \{x \in \mathbb{F}_2^n : \lambda_i \cdot \Delta_i \oplus \lambda_o \cdot S(x) \oplus \lambda_o \cdot S(x \oplus \Delta_i) = b\}$$

$$\text{LDLCT}(\Delta_i, \lambda_i, \lambda_o) = |\text{LDLCT}_0(\Delta_i, \lambda_i, \lambda_o)| - |\text{LDLCT}_1(\Delta_i, \lambda_i, \lambda_o)|$$

$$\mathbb{C}_{\text{LDLCT}}(\Delta_i, \lambda_i, \lambda_o) = 2^{-n} \cdot \text{LDLCT}(\Delta_i, \lambda_i, \lambda_o)$$

# Extended Differential-Linear Connectivity Table (EDLCT)

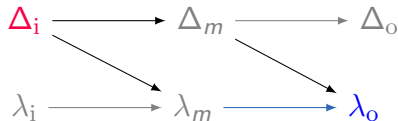


$$\text{EDLCT}_b(\Delta_i, \Delta_o, \lambda_i, \lambda_o) = \{x \in \mathbb{F}_2^n : S(x) \oplus S(x \oplus \Delta_i) = \Delta_o \text{ and } \lambda_i \cdot \Delta_i \oplus \lambda_o \cdot \Delta_o = b\}$$

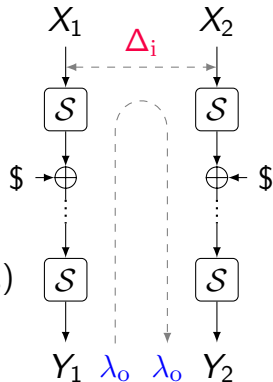
$$\text{EDLCT}(\Delta_i, \Delta_o, \lambda_i, \lambda_o) = |\text{EDLCT}_0(\Delta_i, \Delta_o, \lambda_i, \lambda_o)| - |\text{EDLCT}_1(\Delta_i, \Delta_o, \lambda_i, \lambda_o)|$$

$$\mathbb{C}_{\text{EDLCT}}(\Delta_i, \Delta_o, \lambda_i, \lambda_o) = 2^{-n} \cdot \text{EDLCT}(\Delta_i, \Delta_o, \lambda_i, \lambda_o)$$

# Double Differential-Linear Connectivity Table (DDLCT)



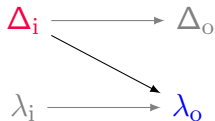
$$\text{DDLCT}(\Delta_i, \lambda_o) = 2^{-n} \sum_{\Delta_m} \sum_{\lambda_m} \text{UDLCT}(\Delta_i, \Delta_m, \lambda_m) \cdot \text{LDLCT}(\Delta_m, \lambda_m, \lambda_o)$$



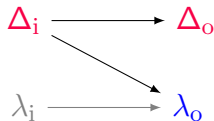


# Generalized DLCT Framework (GBCT)

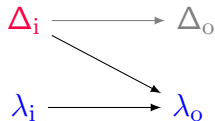
- How to formulate the correlation for more than 1 round?



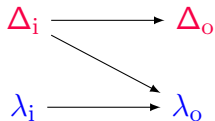
DLCT ( $\Delta_i, \lambda_o$ )



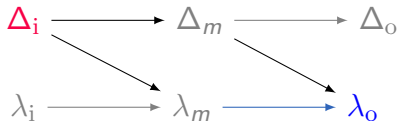
UDLCT ( $\Delta_i, \Delta_o, \lambda_o$ )



LDLCT ( $\Delta_i, \lambda_i, \lambda_o$ )

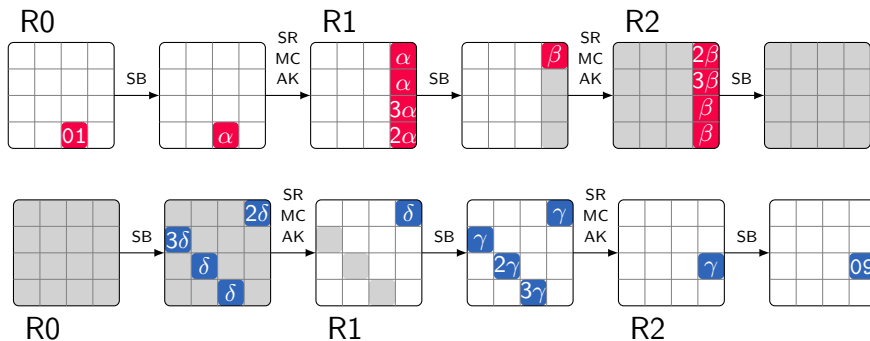


EDLCT ( $\Delta_i, \Delta_o, \lambda_i, \lambda_o$ )



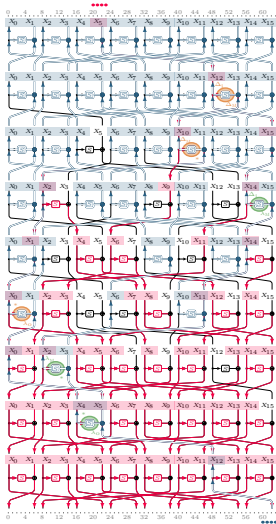
DDLCT ( $\Delta_i, \lambda_o$ )

# Application of the Generalized DLCT Tables - AES ( - differential - linear)



$$\sum_{\alpha, \beta, \gamma, \delta} \mathbb{C}_{\text{UDLCT}}(1, \alpha, \delta) \cdot \mathbb{C}_{\text{EDLCT}}(\alpha, \beta, \delta, \gamma) \cdot \mathbb{C}_{\text{LDLCT}}(\beta, \gamma, 9) = -2^{-7.94}$$

# Application of the Generalized DLCT Tables - TWINE ( - differential - linear)



$$\begin{aligned}\mathbb{C}(\Delta_i, \lambda_o) &= \sum_{\Delta_m} \mathbb{P}_{\text{DDT}}(\Delta_i, \Delta_m) \cdot \mathbb{C}_{\text{DDLCT}}(\Delta_m, \lambda_o) \\ &= \sum_{\lambda_m} \mathbb{C}_{\text{DDLCT}}(\Delta_i, \lambda_m) \cdot \mathbb{C}_{\text{LAT}}^2(\lambda_m, \lambda_o).\end{aligned}$$

$$\mathbb{C}_{\text{tot}}(\Delta_i, \lambda_o) = \mathbb{C}^2(\Delta_i, \lambda_o).$$

Input/Output Differences/Linear-mask	Formula	Exp. Correlation
$(\Delta_i, \lambda_o) = (0xb4, 0x67)$	$-2^{-7.66}$	$-2^{-7.64}$
$(\Delta_i, \lambda_o) = (0x02, 0x02)$	$-2^{-7.92}$	$-2^{-7.93}$
$(\Delta_i, \lambda_o) = (0x55, 0x55)$	$-2^{-7.99}$	$-2^{-7.98}$
$(\Delta_i, \lambda_o) = (0xbf, 0xef)$	$-2^{-8.05}$	$-2^{-8.06}$
$(\Delta_i, \lambda_o) = (0xfe, 0x06)$	$-2^{-8.26}$	$-2^{-8.25}$
$(\Delta_i, \lambda_o) = (0x4b, 0x1a)$	$-2^{-8.43}$	$-2^{-8.44}$

# Differential-Linear Switches and Deterministic Trails



# Cell-Wise and Bit-Wise Switches

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
S(x)	4	0	a	7	b	e	1	d	9	f	6	8	5	2	c	3

$\Delta \backslash \lambda$	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16
1	16	0	0	0	-16	0	0	0	0	0	0	0	0	0	0	0
2	16	-8	-8	0	0	0	8	-8	0	-8	0	8	0	0	0	0
3	16	0	-8	-8	0	-8	8	0	0	0	0	0	0	-8	0	8
4	16	0	-8	0	0	0	-8	0	-16	0	8	0	0	0	8	0
5	16	0	-8	0	0	0	-8	0	0	0	8	0	-16	0	8	0
6	16	-8	8	-8	0	0	-8	0	0	-8	0	0	0	0	0	8
7	16	0	8	0	0	-8	-8	-8	0	0	0	8	0	-8	0	0
8	16	0	0	0	-16	0	0	0	-16	0	0	0	16	0	0	0
9	16	-8	0	-8	16	-8	0	-8	0	8	0	-8	0	8	0	-8
a	16	0	0	8	0	8	0	0	0	0	-8	0	0	-8	-8	-8
b	16	8	0	0	0	0	0	8	0	-8	-8	-8	0	0	-8	0
c	16	0	0	-8	0	0	0	-8	16	0	0	-8	0	0	0	-8
d	16	-8	0	0	0	-8	0	0	0	8	0	0	-16	8	0	0
e	16	0	0	0	0	8	0	8	0	0	-8	-8	0	-8	-8	0
f	16	8	0	8	0	0	0	0	0	-8	-8	0	0	0	-8	-8

- Cell-wise switches:  
 $\text{DLCT}(\Delta_i, 0) = \text{DLCT}(0, \lambda_o) = 2^n$  for all  $\Delta_i, \lambda_o$
- Bit-wise switches:  
 $\text{DLCT}(\Delta_i, \lambda_o) = \pm 2^n$  for  $\Delta_i, \lambda_o \neq 0$ 
  - Example:  $\mathbb{C}(9, 4) = \frac{16}{16}$

# Bit-Wise Switches and Deterministic Trails

$x$	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$S(x)$	4	0	a	7	b	e	1	d	9	f	6	8	5	2	c	3

$\Delta \backslash \lambda$	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16
1	16	0	0	0	-16	0	0	0	0	0	0	0	0	0	0	0
2	16	-8	-8	0	0	0	8	-8	0	-8	0	8	0	0	0	0
3	16	0	-8	-8	0	-8	8	0	0	0	0	0	0	-8	0	8
4	16	0	-8	0	0	0	-8	0	-16	0	8	0	0	0	8	0
5	16	0	-8	0	0	0	-8	0	0	0	8	0	-16	0	8	0
6	16	-8	8	-8	0	0	-8	0	0	-8	0	0	0	0	0	8
7	16	0	8	0	0	-8	-8	-8	0	0	0	8	0	-8	0	0
8	16	0	0	0	-16	0	0	0	-16	0	0	0	16	0	0	0
9	16	-8	0	-8	16	-8	0	-8	0	8	0	-8	0	8	0	-8
a	16	0	0	8	0	8	0	0	0	0	-8	0	0	-8	-8	-8
b	16	8	0	0	0	0	0	8	0	-8	-8	-8	0	0	-8	0
c	16	0	0	-8	0	0	0	-8	16	0	0	-8	0	0	0	-8
d	16	-8	0	0	0	-8	0	0	0	8	0	0	-16	8	0	0
e	16	0	0	0	0	8	0	8	0	0	-8	-8	0	-8	-8	0
f	16	8	0	8	0	0	0	0	0	-8	-8	0	0	0	-8	-8

$$\Delta_i = (0, 0, 0, 1) \xrightarrow{S} \Delta_o = (?, 1, ?, ?)$$

$$\Delta_i = (0, 1, 0, 0) \xrightarrow{S} \Delta_o = (1, ?, ?, ?)$$

$$\Delta_i = (1, 0, 0, 0) \xrightarrow{S} \Delta_o = (1, 1, ?, ?)$$

$$\Delta_i = (1, 0, 0, 1) \xrightarrow{S} \Delta_o = (?, 0, ?, ?)$$

$$\Delta_i = (1, 1, 0, 0) \xrightarrow{S} \Delta_o = (0, ?, ?, ?)$$

$$\lambda_i = (1, ?, ?, 1) \xleftarrow{S} \lambda_o = (0, 1, 0, 0)$$

$$\lambda_i = (1, 1, ?, ?) \xleftarrow{S} \lambda_o = (1, 0, 0, 0)$$

$$\lambda_i = (0, ?, ?, ?) \xleftarrow{S} \lambda_o = (1, 1, 0, 0)$$

# Automatic Tools to Search for DL Distinguishers

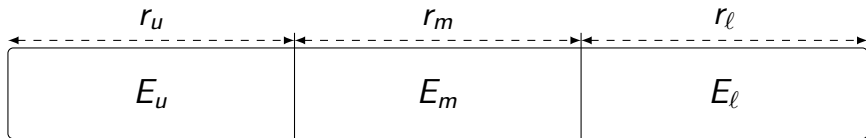


# Overview of Our Method to Search for Distinguishers

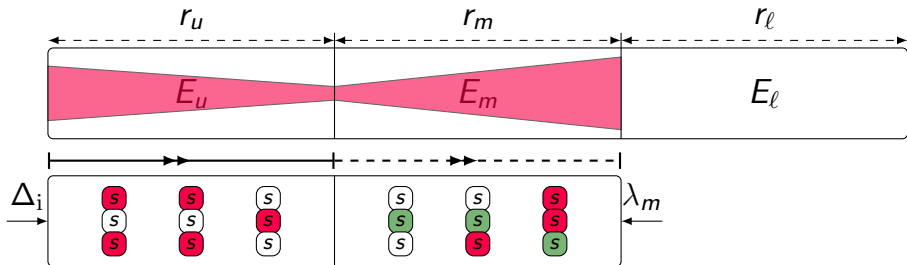
$E$



# Overview of Our Method to Search for Distinguishers

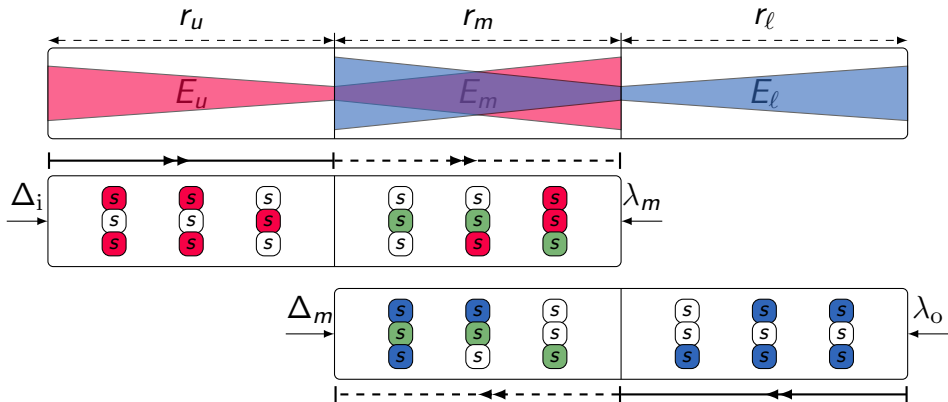


# Overview of Our Method to Search for Distinguishers



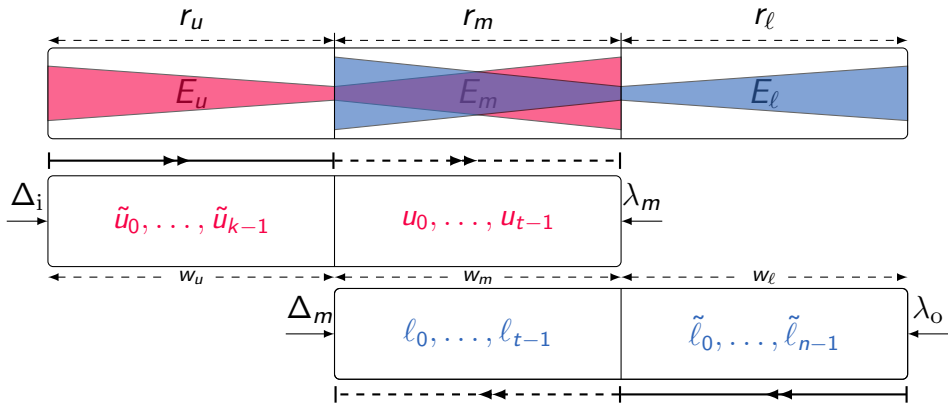
■ differentially active S-box  
 ■ linearly active S-box  
 ■ common active S-box

# Overview of Our Method to Search for Distinguishers



■ differentially active S-box  
 ■ linearly active S-box  
 ■ common active S-box

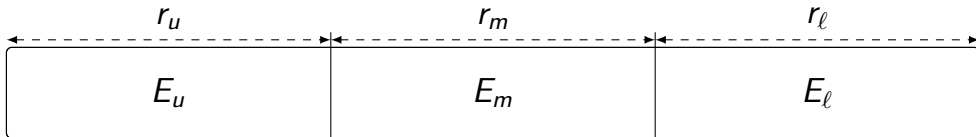
# Overview of Our Method to Search for Distinguishers



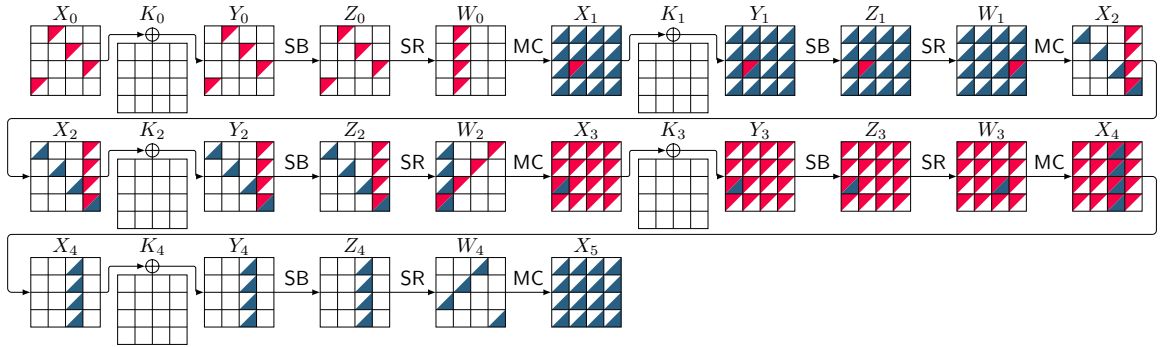
$$\min \left( \sum_{i=0}^{k-1} w_u \cdot \tilde{u}_i + \sum_{j=0}^{t-1} w_m \cdot \text{bool2int}(\ell_j + u_j = 2) + \sum_{k=0}^{n-1} w_l \cdot \tilde{\ell}_k \right)$$

# Usage of Our Tool

```
python3 attack.py -RU 6 -RM 10 -RL 6
```



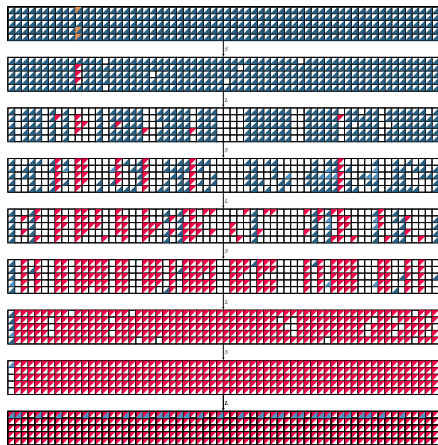
# Results: A 5-round DL Distinguisher for AES



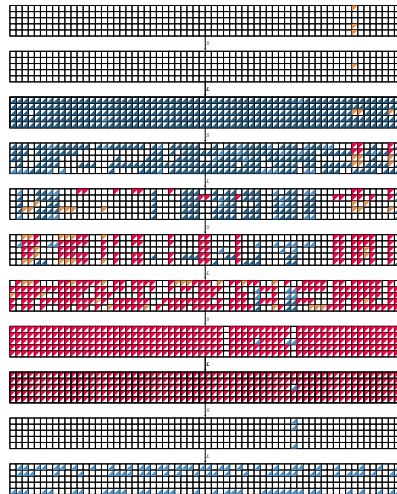
$$r_0 = 1, r_m = 3, r_1 = 1, p = 2^{-24.00}, r = 2^{-7.66}, q^2 = 2^{-24.00}, prq^2 = 2^{-55.66}$$

$\Delta X_0$  001c00000000e200000000dfb3000000     $\Delta X_1$  00000000000000000000f7000000000000  
 $\Gamma X_4$  0000000000000000000067000000000000     $\Gamma X_5$  21d3814d93b1ef228e923507f67383fd

# Results: Application to Ascon-p( active difference unknown difference active mask unknown mask )



$C = 1$



$C = 2^{-4.33}$

# Contributions and Future Works





# Contributions and Future Works

## ■ Contributions

- 💎 We generalized the DLCT framework from one S-box layer to multiple rounds
- 💎 We proposed an automatic tool for finding optimum DL distinguishers
- 💎 We applied our tool to almost any design paradigm

## ■ Future works

- A** Extending the application of our tool to other primitives, e.g., ARX
- A** Extending our tool to a unified model for finding complete attack (key recovery)

🐙: <https://github.com/hadipourh/DL>

📄: <https://ia.cr/2024/255>

# Bibliography I

- [Bar+19] Achiya Bar-On et al. **DLCT: A New Tool for Differential-Linear Cryptanalysis**. EUROCRYPT 2019. Vol. 11476. LNCS. Springer, 2019, pp. 313–342. DOI: [10.1007/978-3-030-17653-2\\_11](https://doi.org/10.1007/978-3-030-17653-2_11).
- [BLN14] Céline Blondeau, Gregor Leander, and Kaisa Nyberg. **Differential-Linear Cryptanalysis Revisited**. FSE 2014. Ed. by Carlos Cid and Christian Rechberger. Vol. 8540. LNCS. Springer, 2014, pp. 411–430. DOI: [10.1007/978-3-662-46706-0\\_21](https://doi.org/10.1007/978-3-662-46706-0_21).
- [DIK08] Orr Dunkelman, Sebastiaan Indesteege, and Nathan Keller. **A Differential-Linear Attack on 12-Round Serpent**. INDOCRYPT 2008. Ed. by Dipanwita Roy Chowdhury, Vincent Rijmen, and Abhijit Das. Vol. 5365. LNCS. Springer, 2008, pp. 308–321. DOI: [10.1007/978-3-540-89754-5\\_24](https://doi.org/10.1007/978-3-540-89754-5_24).

## Bibliography II

- [DKS14] Orr Dunkelman, Nathan Keller, and Adi Shamir. **A Practical-Time Related-Key Attack on the KASUMI Cryptosystem Used in GSM and 3G Telephony.** *J. Cryptol.* 27.4 (2014), pp. 824–849. DOI: [10.1007/s00145-013-9154-9](https://doi.org/10.1007/s00145-013-9154-9).
- [HNE22] Hosein Hadipour, Marcel Nageler, and Maria Eichlseder. **Throwing Boomerangs into Feistel Structures Application to CLEFIA, WARP, LBlock, LBlock-s and TWINE.** *IACR Trans. Symmetric Cryptol.* 2022.3 (2022), pp. 271–302. DOI: [10.46586/TOSC.V2022.I3.271-302](https://doi.org/10.46586/TOSC.V2022.I3.271-302).
- [LH94] Susan K. Langford and Martin E. Hellman. **Differential-Linear Cryptanalysis.** CRYPTO '94. Vol. 839. Springer, 1994, pp. 17–25. DOI: [10.1007/3-540-48658-5\\_3](https://doi.org/10.1007/3-540-48658-5_3).

## Bibliography III

- [ZWH24] Yanyan Zhou, Senpeng Wang, and Bin Hu. **MILP/MIQCP-Based Fully Automatic Method of Searching for Differential-Linear Distinguishers for SIMON-Like Ciphers.** *IET Information Security* 2024 (2024). DOI: [10.1049/2024/8315115](https://doi.org/10.1049/2024/8315115).

# Properties of Generalized DLCT Tables - I

- $\text{DLCT}(\Delta_i, \lambda_o) = \sum_{\Delta_o} \text{UDLCT}(\Delta_i, \Delta_o, \lambda_o)$
- $\text{UDLCT}(\Delta_i, \Delta_o, \lambda_o) = (-1)^{\Delta_o \cdot \lambda_o} \text{DDT}(\Delta_i, \Delta_o)$
- $\text{LDLCT}(\Delta_i, \lambda_i, \lambda_o) = (-1)^{\Delta_i \cdot \lambda_i} \text{DLCT}(\Delta_i, \lambda_o)$
- $\text{EDLCT}(\Delta_i, \Delta_o, \lambda_i, \lambda_o) = (-1)^{\lambda_i \cdot \Delta_i \oplus \lambda_o \cdot \Delta_o} \text{DDT}(\Delta_i, \Delta_o)$
- $\text{LDLCT}(\Delta_i, \lambda_i, \lambda_o) = \sum_{\Delta_o} \text{EDLCT}(\Delta_i, \Delta_o, \lambda_i, \lambda_o)$
- $\sum_{\Delta_i} \text{LDLCT}(\Delta_i, \lambda_i, \lambda_o) = \text{LAT}^2(\lambda_i, \lambda_o)$

## Properties of Generalized DLCT Tables - II

- $\text{DDLCT}(\Delta_i, \lambda_o) = 2^{-n} \cdot \sum_{\Delta_m} \sum_{\lambda_m} \text{UDLCT}(\Delta_i, \Delta_m, \lambda_m) \cdot \text{LDLCT}(\Delta_m, \lambda_m, \lambda_o)$

$$\begin{aligned}\text{DDLCT}(\Delta_i, \lambda_o) &= \sum_{\Delta_m} \text{DDT}(\Delta_i, \Delta_m) \cdot \text{DLCT}(\Delta_m, \lambda_o) \\ &= 2^{-n} \sum_{\lambda_m} \text{DLCT}(\Delta_i, \lambda_m) \cdot \text{LAT}^2(\lambda_m, \lambda_o).\end{aligned}$$

## Example: Distinguishers for up to 17 Rounds of TWINE

- Comparing the data complexity of best boomerang and DL distinguishers

# Rounds	Boomerang [HNE22]	Differential-Linear	Gain
5	1	1	1
7	$2^{3.20}$	1	$2^{3.20}$
13	$2^{34.32}$	$2^{27.16}$	$2^{7.16}$
14	$2^{42.25}$	$2^{31.28}$	$2^{10.97}$
15	$2^{51.03}$	$2^{38.98}$	$2^{12.05}$
16	$2^{58.04}$	$2^{47.28}$	$2^{10.76}$
<b>17</b>	-	$2^{59.24}$	-

## Results: Distinguishers for up to 17 Rounds of LBlock

- Comparing the data complexity of best boomerang and DL distinguishers

# Rounds	Boomerang [HNE22]	Differential-Linear	Gain
5	1	1	1
7	$2^{2.97}$	1	$2^{2.97}$
13	$2^{30.28}$	$2^{23.78}$	$2^{6.50}$
14	$2^{38.86}$	$2^{30.34}$	$2^{8.52}$
15	$2^{46.90}$	$2^{38.26}$	$2^{8.64}$
16	$2^{57.16}$	$2^{46.26}$	$2^{10.90}$
<b>17</b>	-	$2^{58.30}$	-





## Results: Distinguishers for up to 8 Rounds of CLEFIA

- Comparing the data complexity of best boomerang and DL distinguishers


# Rounds	Boomerang [HNE22]	Differential-Linear	Gain
3	1	1	1
4	$2^{6.32}$	1	$2^{6.32}$
5	$2^{12.26}$	$2^{5.36}$	$2^{6.90}$
6	$2^{22.45}$	$2^{14.14}$	$2^{8.31}$
7	$2^{32.67}$	$2^{23.50}$	$2^{9.17}$
8	$2^{76.03}$	$2^{66.86}$	$2^{9.17}$


## Results: Application to SERPENT


- : Experimentally verified


Cipher	#R	$\mathbb{C}$		Ref.
SERPENT	3	<b><math>2^{-0.68}</math></b>	✓	This work
	4	$2^{-12.75}$		[DIK08]
	4	<b><math>2^{-5.54}</math></b>	✓	This work
	5	$2^{-16.75}$		[DIK08]
	5	<b><math>2^{-11.10}</math></b>	✓	This work
	8	$2^{-39.18}$		This work
	9	$2^{-56.50}$		[DIK08]
	9	<b><math>2^{-50.95}</math></b>		This work

# Results: Application to Simeck

- : Experimentally verified

Cipher	#R	$\mathbb{C}$		Ref.
Simeck-32	7	<b>1</b>	✓	This work
	14	$2^{-16.63}$		[ZWH24]
	14	<b><math>2^{-13.92}</math></b>	✓	This work

Cipher	#R	$\mathbb{C}$		Ref.
Simeck-48	8	<b>1</b>	✓	This work
	17	$2^{-22.37}$		[ZWH24]
	17	<b><math>2^{-13.89}</math></b>	✓	This work
	18	$2^{-24.75}$		[ZWH24]
	18	<b><math>2^{-15.89}</math></b>		This work
	<b>19</b>	<b><math>2^{-17.89}</math></b>		This work
	<b>20</b>	<b><math>2^{-21.89}</math></b>		This work

Cipher	#R	$\mathbb{C}$		Ref.
Simeck-64	10	<b>1</b>	✓	This work
	24	$2^{-38.13}$		[ZWH24]
	24	<b><math>2^{-25.14}</math></b>		This work
	25	$2^{-41.04}$		[ZWH24]
	25	<b><math>2^{-27.14}</math></b>		This work
	<b>26</b>	<b><math>2^{-30.35}</math></b>		This work