Automatic Search for Differential Trails in ARX Ciphers

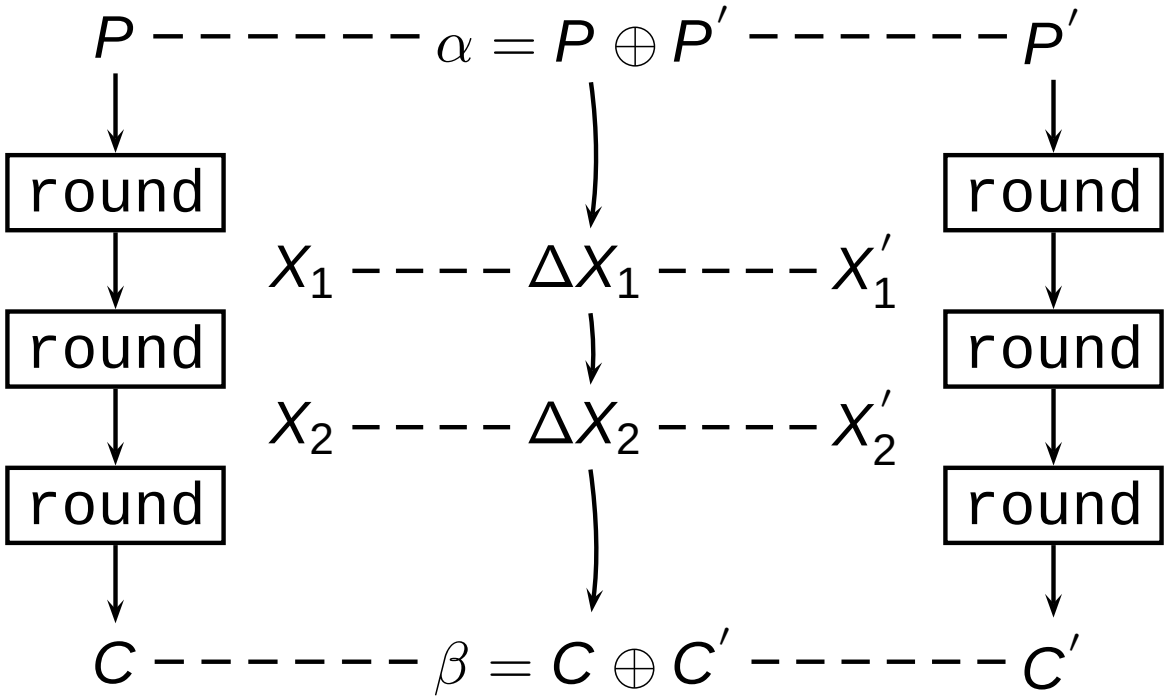
A. Biryukov V. Velichkov

Laboratory of Algorithmics, Cryptology and Security (LACS)
University of Luxembourg

RSA Conference Cryptographers' Track – 2014
February 24-28, San Francisco, USA

Outline

Differential Cryptanalysis (DC) [Biham,Shamir,1991]



Differentials, Trails and Probabilities

- Differential for r rounds:

$$(\alpha, \beta) \ .$$

- Differential trail (characteristic) for r rounds:

$$(\alpha = \Delta X_0,\ \Delta X_1\ \ldots\ \Delta X_{r-1},\ \Delta X_r = \beta) \ .$$

- Differential Probability (DP) of a single round:

$$\mathrm{DP}(\alpha \xrightarrow{F_K} \beta) = \frac{\#\{X, K : F_K(X) \oplus F_K(X \oplus \alpha) = \beta\}}{\#\{X, K\}} \ .$$

- DP of a trail $^{(*)}$:

$$\mathrm{DP}(\Delta X_0, \Delta X_1, \ldots, \Delta X_r) = \prod_{i=1}^{r} \mathrm{DP}(\Delta X_{i-1} \xrightarrow{F_{K_i}} \Delta X_i) \ .$$

$^{(*)}$ Under certain assumptions: Markov cipher, independent round keys, etc.

Difference Distribution Table (DDT)

α, β	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	16	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1	.	.	.	2	.	.	.	2	.	2	4	.	4	2	.	.
2	.	.	.	2	.	6	2	2	.	2	.	.	.	.	2	.
3	.	.	2	.	2	.	.	.	.	4	2	.	2	.	.	4
4	.	.	.	2	.	.	6	.	.	2	.	4	2	.	.	.
5	.	4	.	.	.	2	2	.	.	.	4	.	2	.	.	2
6	.	.	.	4	.	4	.	.	.	.	.	.	2	2	2	2
7	.	.	2	2	2	.	2	.	.	2	2	.	.	.	.	4
8	.	.	.	.	.	.	2	2	.	.	.	4	.	4	2	2
9	.	2	.	.	2	.	.	4	2	.	2	2	2	.	.	.
A	.	2	2	.	.	.	.	.	6	.	.	2	.	.	4	.
B	.	.	8	.	.	2	.	2	.	.	.	.	.	2	.	2
C	.	2	.	.	2	2	2	.	.	.	.	2	.	6	.	.
D	.	4	.	.	.	.	.	4	2	.	2	.	2	.	2	.
E	.	.	2	4	2	.	.	.	6	.	.	.	.	.	2	.
F	.	2	.	.	6	.	.	.	.	4	.	2	.	.	2	.

Searching for the Best Trail

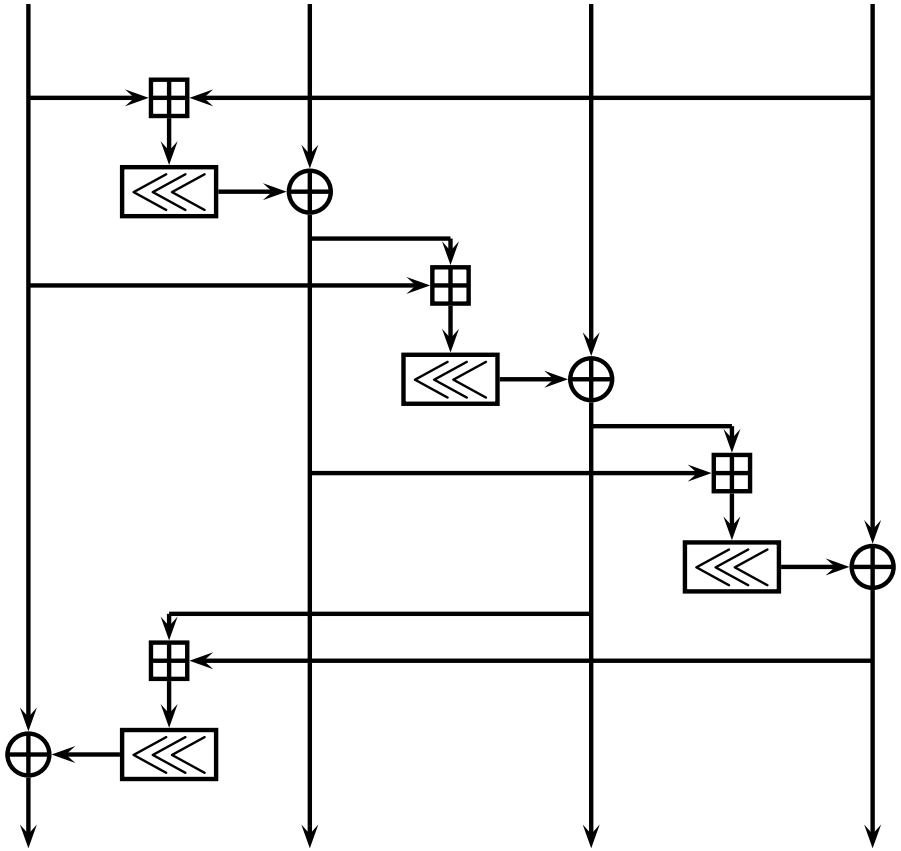
Matsui's branch-and-bound algorithm:

> Mitsuru Matsui, *On Correlation Between the Order of S-boxes and the Strength of DES*, EUROCRYPT'94.

- Find the best trails for up to 16 rounds of DES.
- Best $=$ maximum probability.
- Lower bound on the prob. of the best differential.
- Indication of the strength against DC; first step in a DC attack.
- **Problem:** not applicable to ciphers without S-boxes such as ARX.

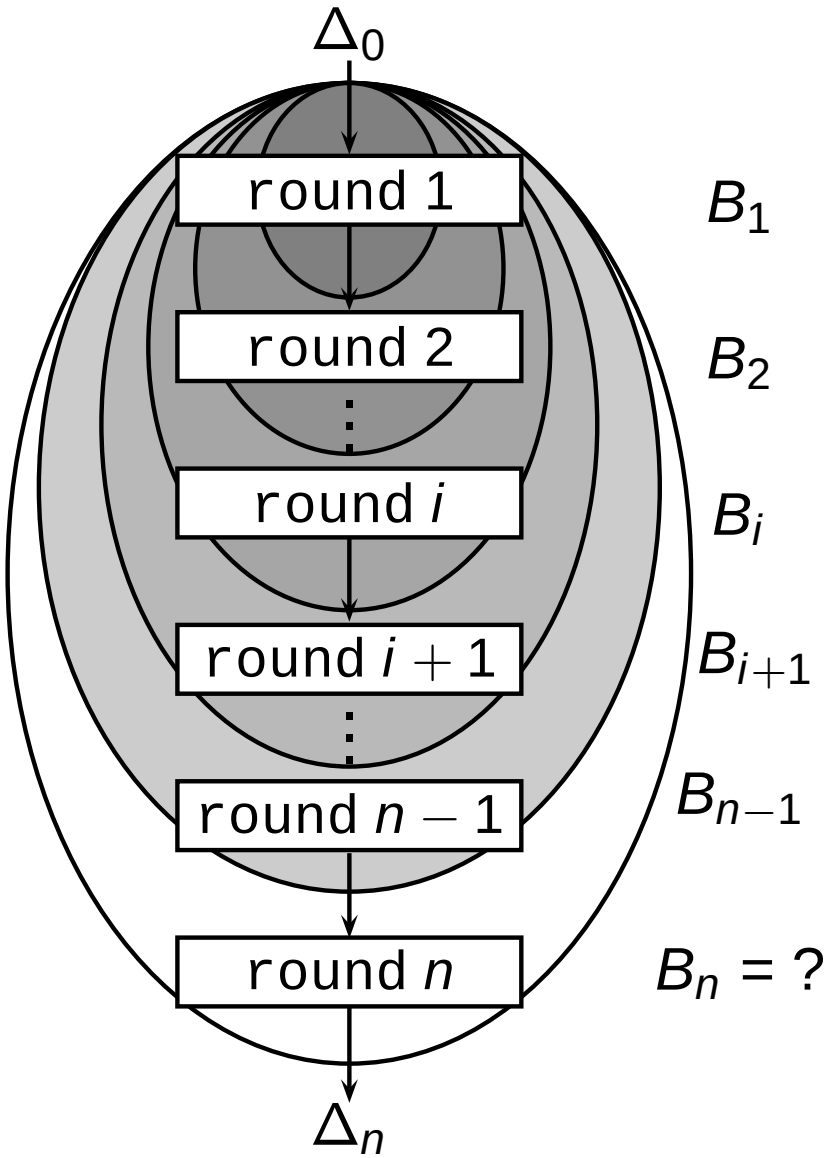
Ciphers Based on Addition, Rotation, XOR (ARX)

In ARX ADD and XOR provide non-linearity similarly to an S-box



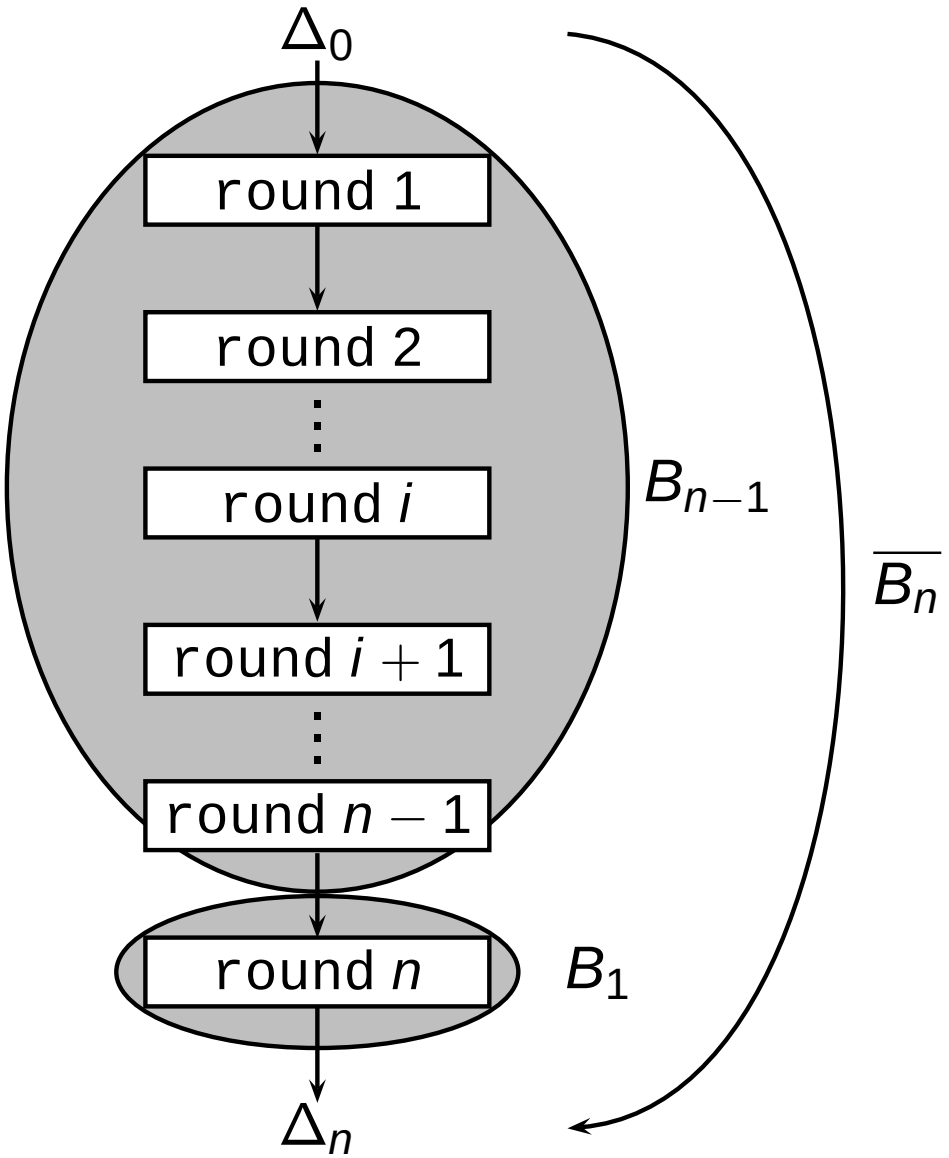
Examples: FEAL, TEA/XTEA, Salsa20, Threefish, etc.

Outline



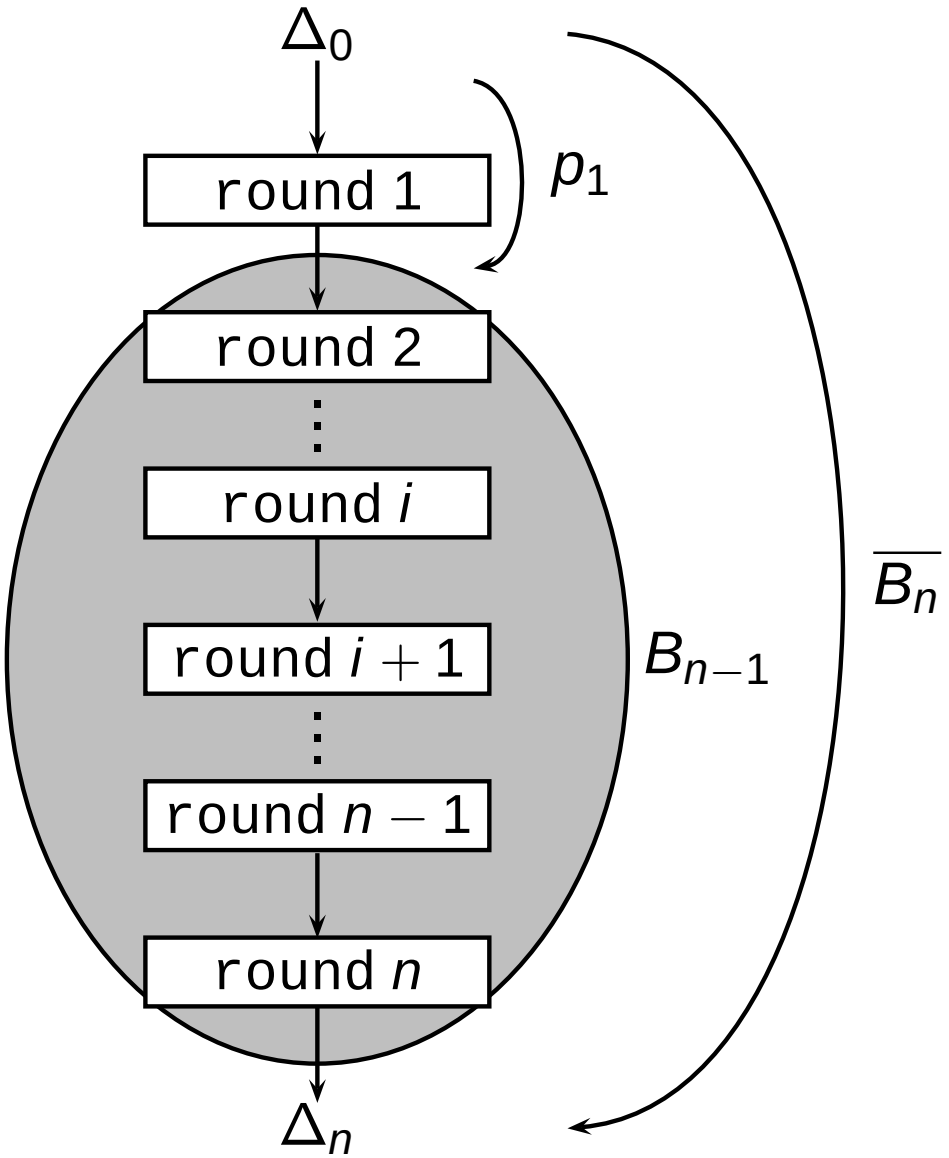
Input: best p for $n-1$ rounds:
$B_1, B_2, \ldots, B_{n-1}$

Output: best p for n rounds:
B_n

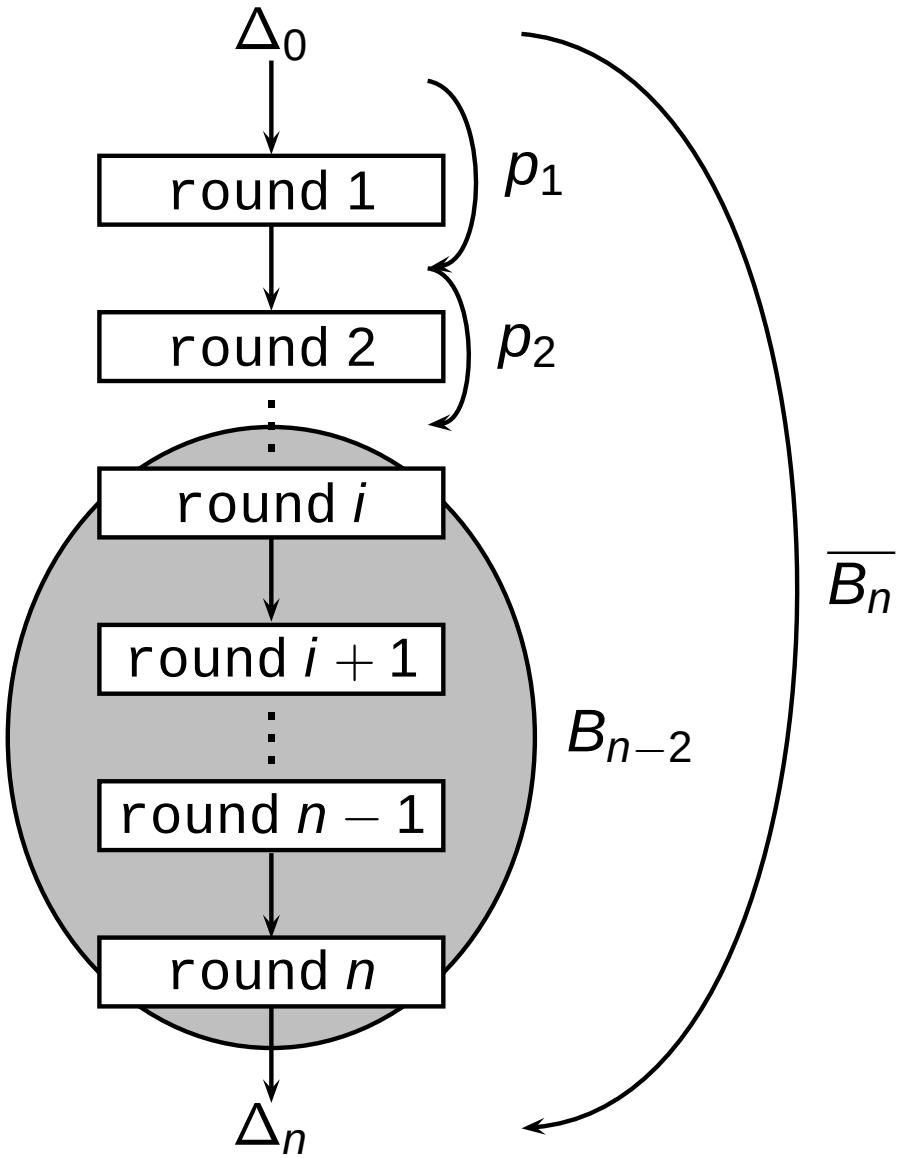


init bound:

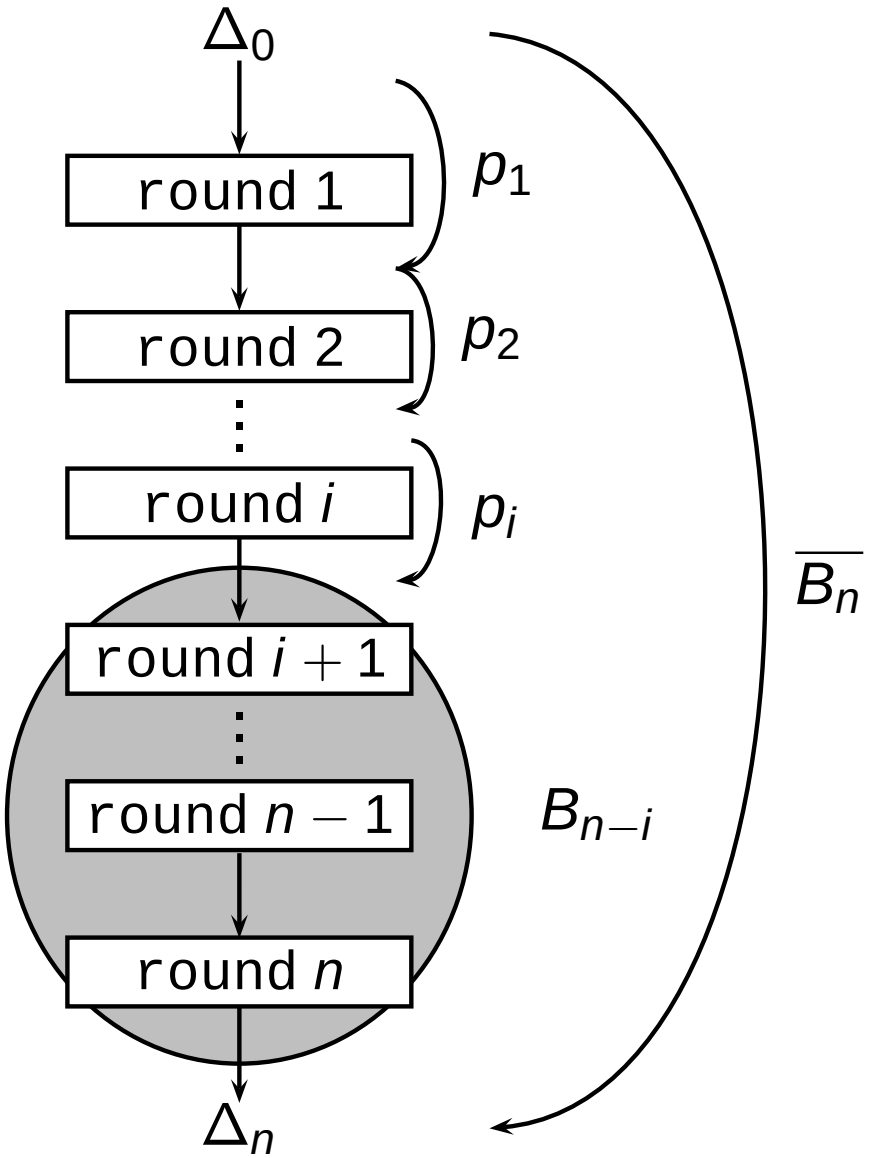
$$\overline{B_n} \leftarrow B_{n-1}B_1$$



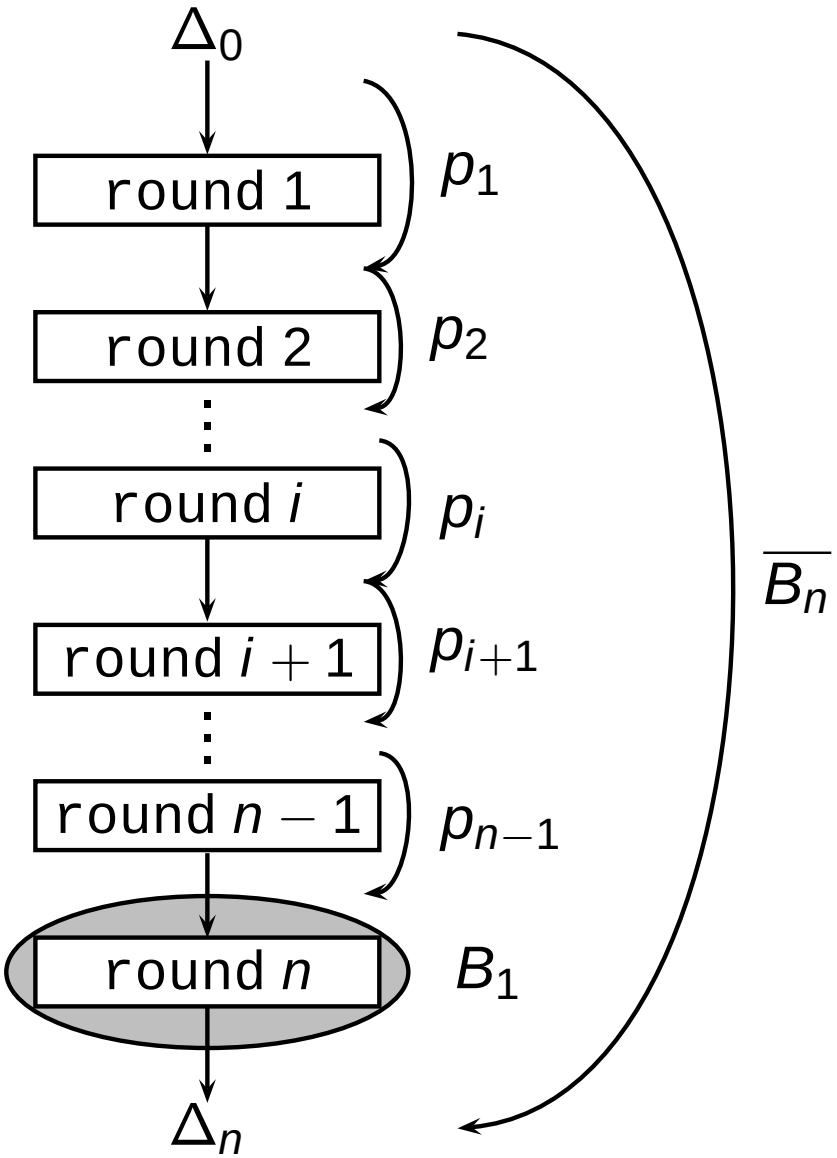
for all Δ_0 :
  if $p_1 B_{n-1} \geq \overline{B_n}$:
    call round 2



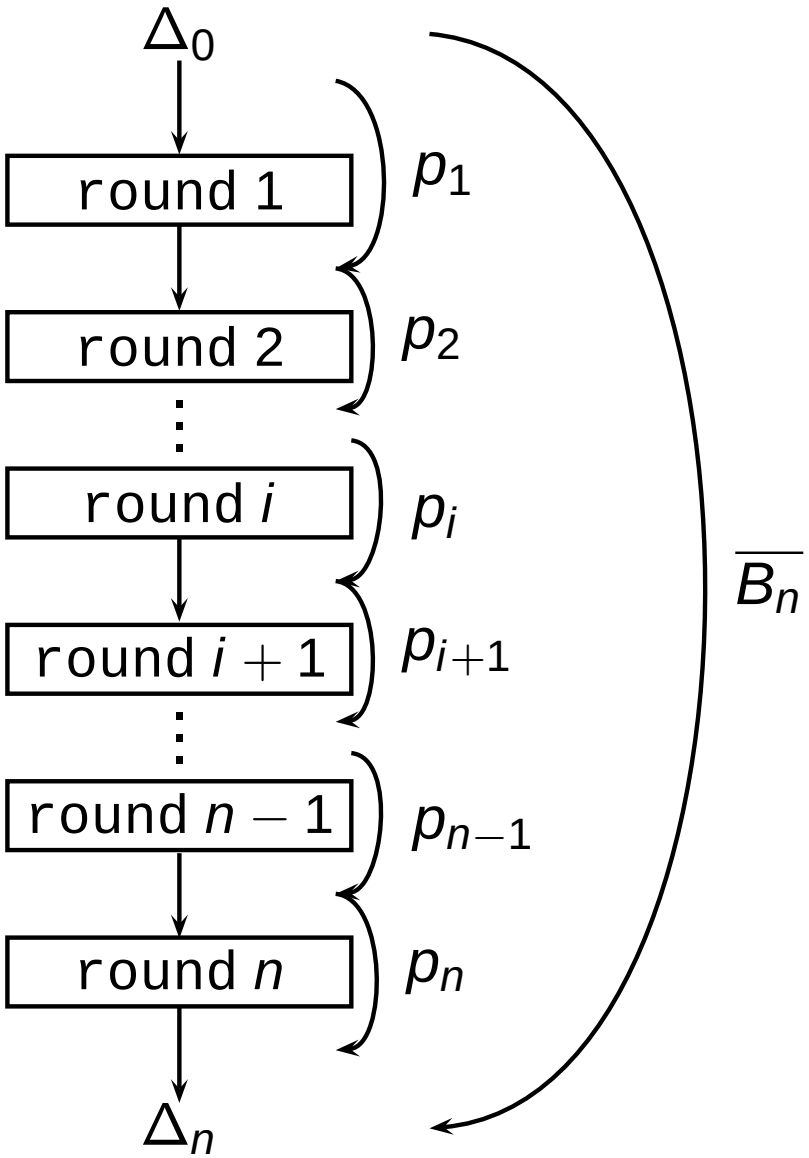
if $p_1 p_2 B_{n-2} \geq \overline{B_n}$:
call round 3



if $p_1 p_2 \ldots p_i B_{n-i} \geq \overline{B_n}$:
call round $i+1$

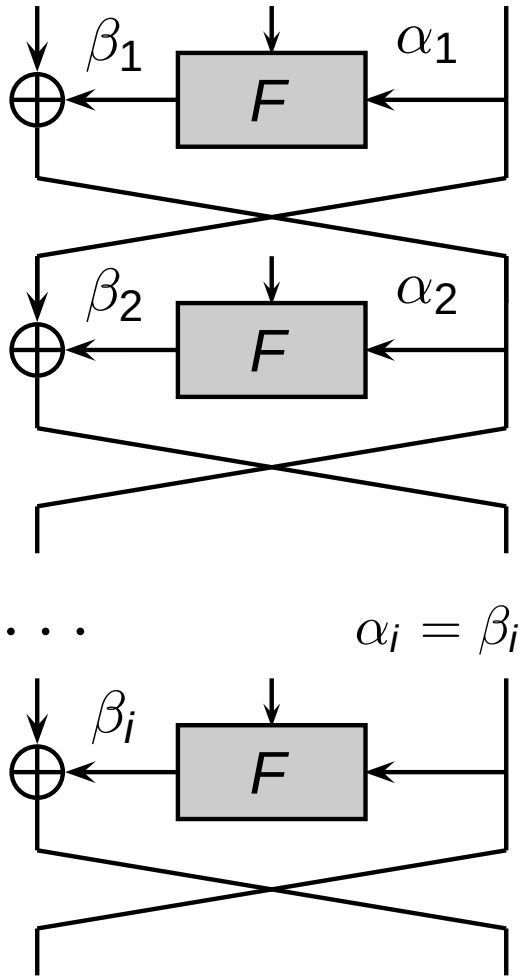


if $p_1 p_2 \ldots p_{n-1} B_1 \geq \overline{B_n}$:
call round n



if $p = p_1 p_2 \ldots p_{n-1} p_n \geq \overline{B_n}$:
update bound:
$\overline{B_n} \leftarrow p$

Application to DES



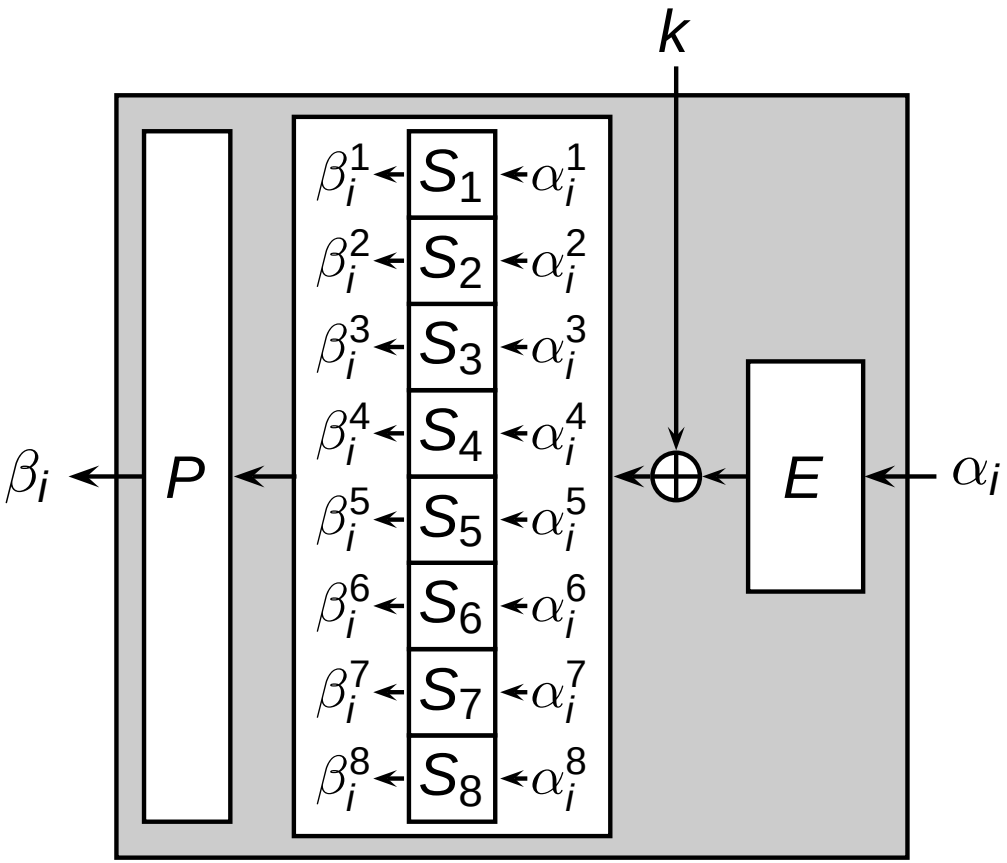
round 1
for all α_1 :
$\beta_1 : p_1 = \max_\beta p(\alpha_1 \to \beta)$
if $p_1 B_{n-1} \geq \overline{B_n}$:
call round 2

round 2
for all α_2, β_2 :
$p_2 = p(\alpha_2 \to \beta_2)$
if $p_1 p_2 B_{n-2} \geq \overline{B_n}$
call round 3

...

round i
$\alpha_i = \beta_{i-1} \oplus \alpha_{i-2}$
for all $\beta_i : p_i = p(\alpha_i \to \beta_i)$:
if $p_1 p_2 \ldots p_i B_{n-i} \geq \overline{B_n}$:
call round $i+1$

Divide-and-Conquer the Input to the S-box Layer

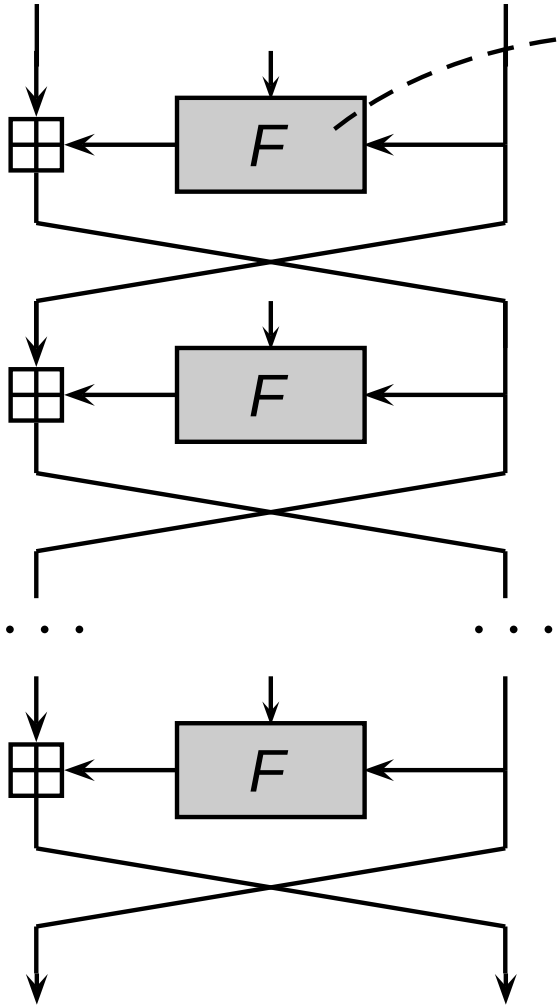


round 2
for $j : 1 \leq j \leq 8$:
  for all α_2^j, β_2^j :
    $p^j = p(\alpha_2^j \rightarrow \beta_2^j)$
    $p_2 = p^1 p^2 \ldots p^j$
    if $p_1 p_2 B_{n-2} \geq \overline{B_n}$
      $j = j + 1$
    if $j > 8$
      call round 3

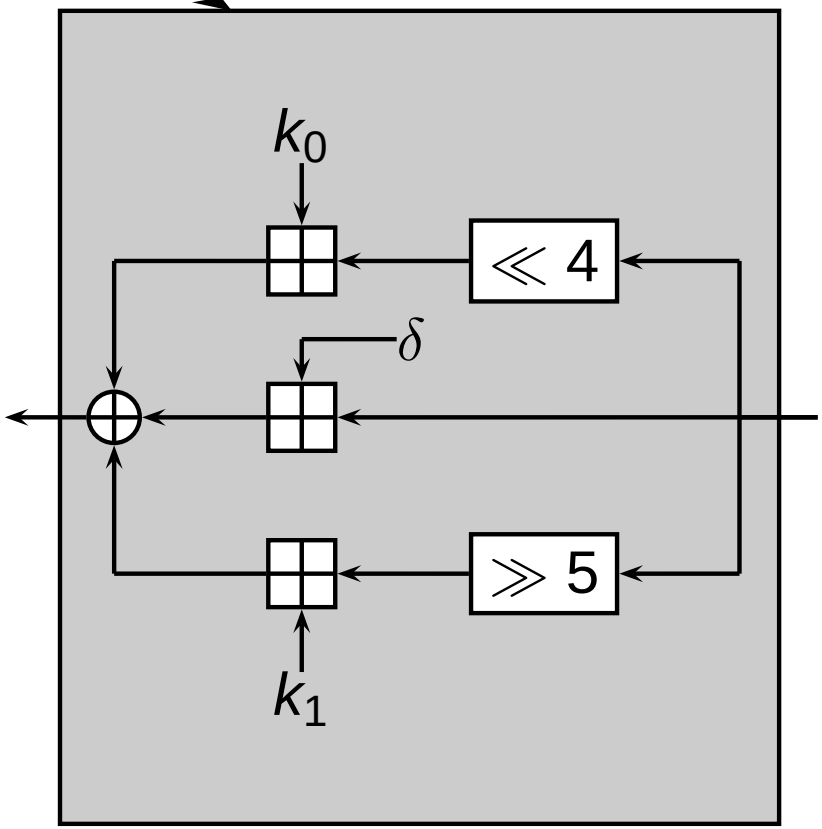
Note: p is computed using the DDT of the S-boxes.

Outline

Application to ARX



64 Rounds



Block Cipher TEA

Application to ARX: Problems and Solutions

Problems: ☹

1. No S-boxes $\Longrightarrow$ divide-and-conquer trick does not work.
2. Infeasible to compute full DDT for ADD or XOR.

Solutions: ☺

1. Partial difference distribution table (pDDT).
2. The country roads and highways analogy.

The Catch?

- Not guaranteed to find the (provably) best trail.

Partial Difference Distribution Table (pDDT)

Definition

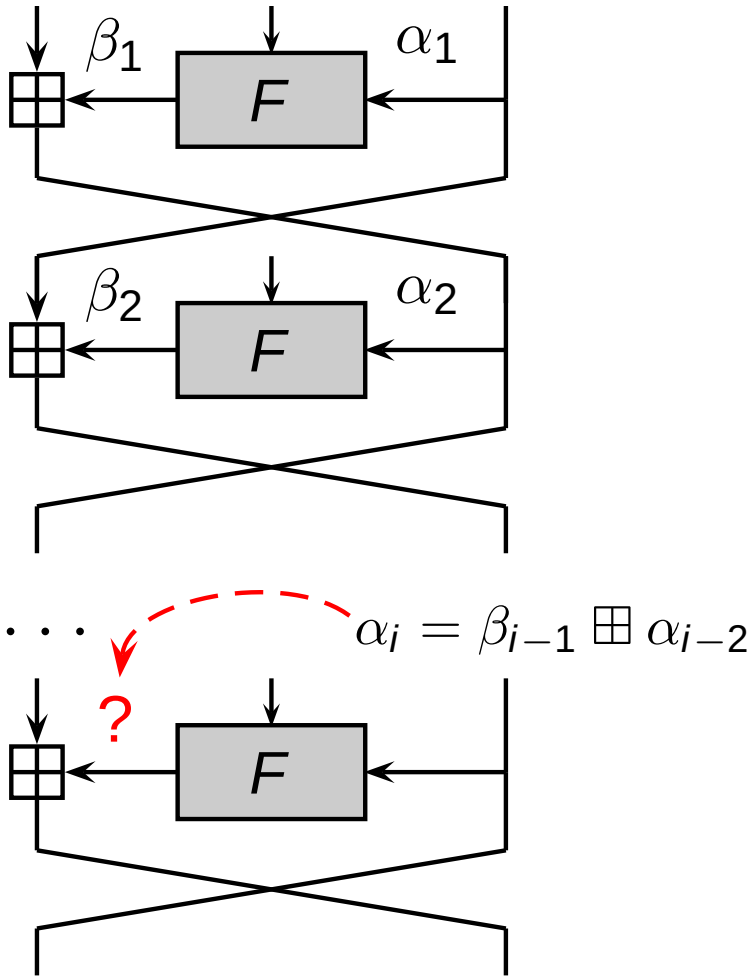
A partial difference distribution table (pDDT) D for the non-linear mapping S is a DDT that contains differentials $(\alpha \xrightarrow{S} \beta)$ with probabilities larger than or equal to a fixed threshold $p_{\text{thres}} > 0$:

$$(\alpha \xrightarrow{S} \beta) \in D \iff p(\alpha \xrightarrow{S} \beta) \geq p_{\text{thres}} \quad .$$

Definition

A pDDT is said to be complete (resp. incomplete) if it contains all (resp. not all) differentials that have probability $\geq p_{\text{thres}}$.

Problem: for given α_i no transition in the pDDT



round 1
for all $\alpha_1, \beta_1 \in \text{pDDT}$:
$p_1 = p(\alpha_1 \to \beta_1)$
if $p_1 B_{n-1} \geq \overline{B_n}$:
call round 2

round 2
for all $\alpha_2, \beta_2 \in \text{pDDT}$:
$p_2 = p(\alpha_2 \to \beta_2)$
if $p_1 p_2 B_{n-2} \geq \overline{B_n}$
call round 3

$\cdots$

round i
$\alpha_i = \beta_{i-1} \oplus \alpha_{i-2}$
$\nexists \beta : (\alpha_i, \beta) \in \text{pDDT}$

The Highways and Country Roads Analogy

Highways and Country Roads

Definition (Highway)

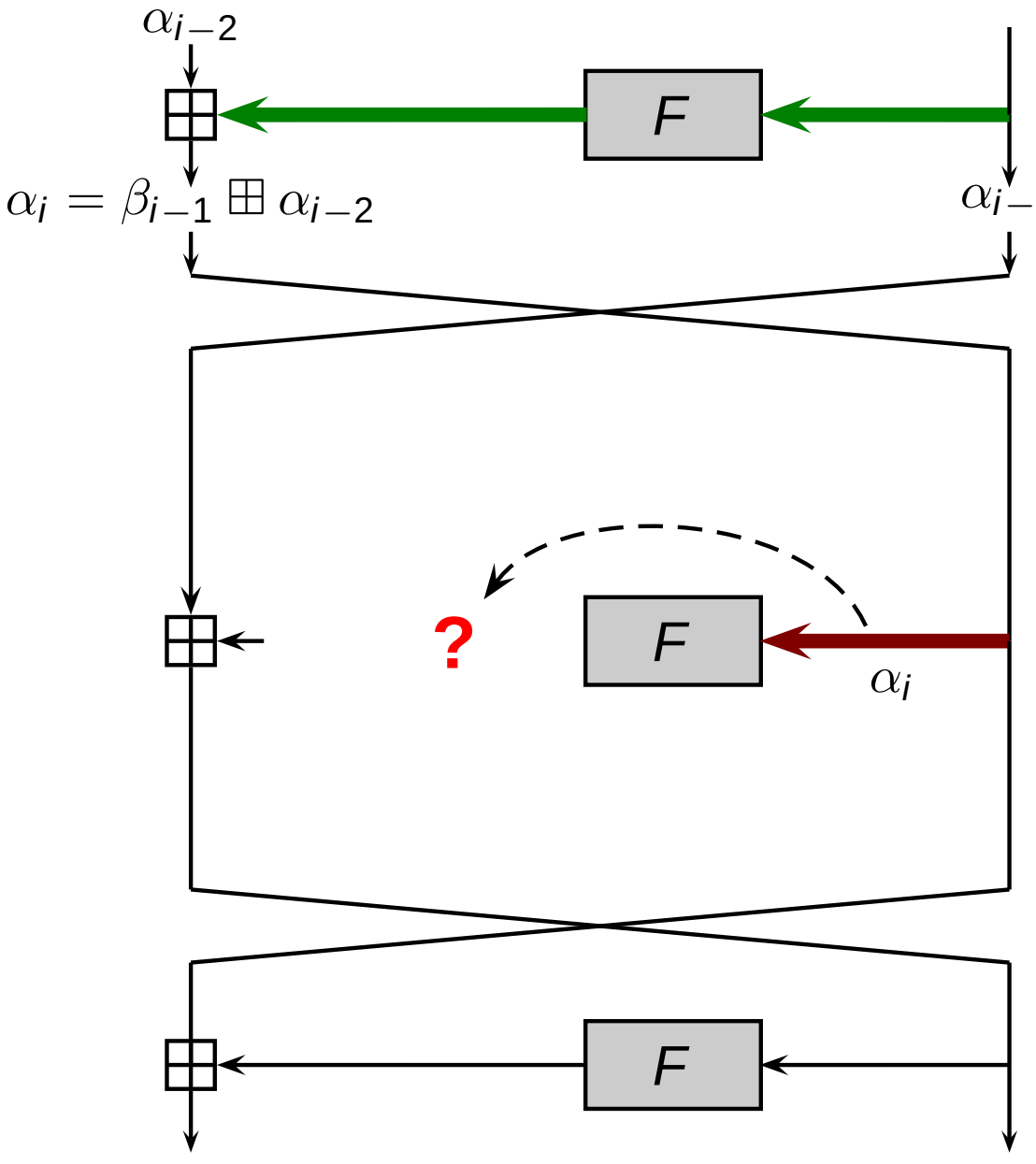
A highway is a transition $(\alpha \to \beta)$ such that $p(\alpha \to \beta) \geq p_{\text{thres}}$ for some fixed probability threshold p_{thres}.

Definition (Country road)

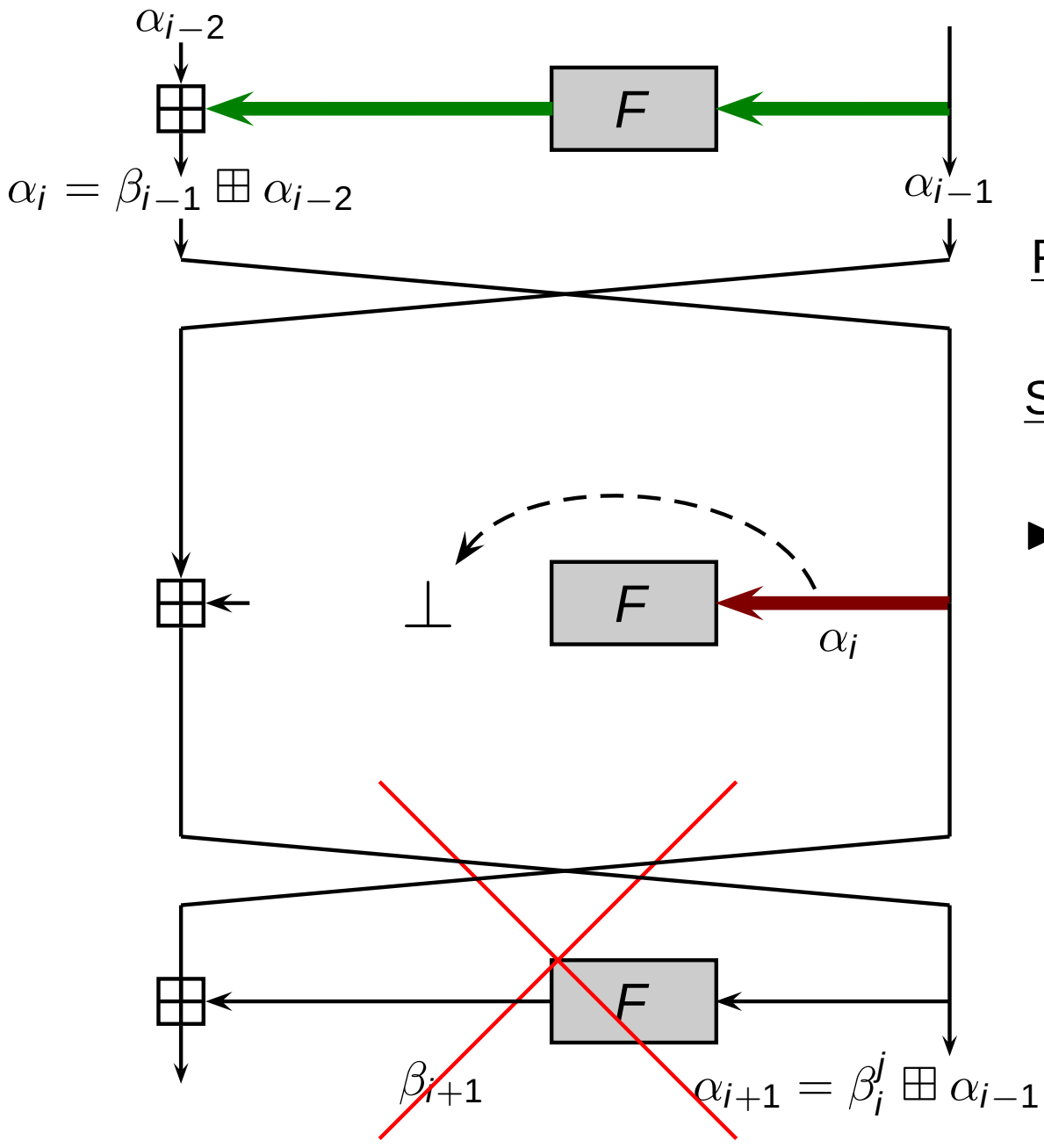
All transitions that are not highways are country roads.

Remark

All transitions in a pDDT are highways.



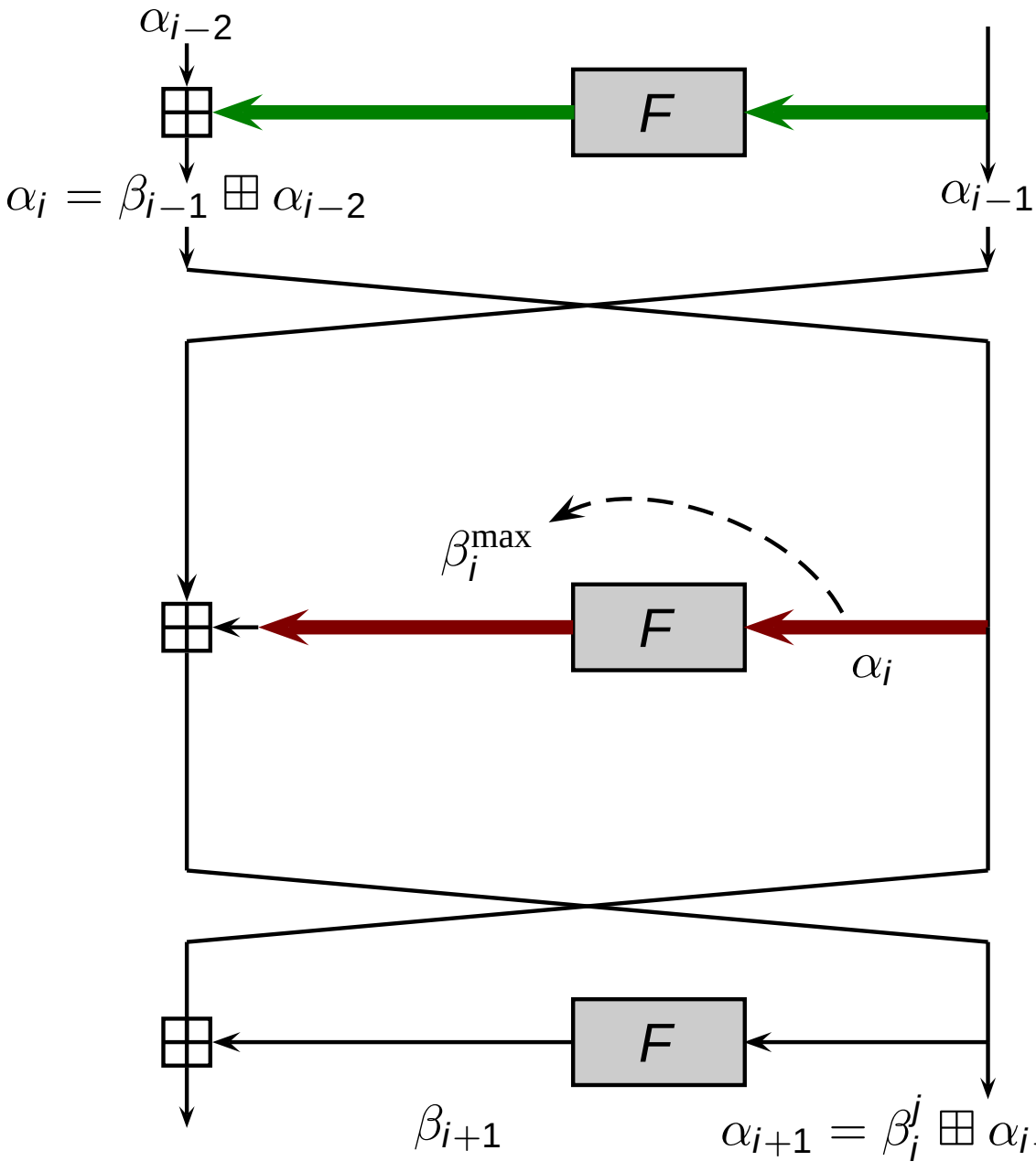
Problem: $\nexists \beta : (\alpha_i, \beta) \in D$



Problem: $\nexists \beta : (\alpha_i, \beta) \in D$

Solution 1: do nothing

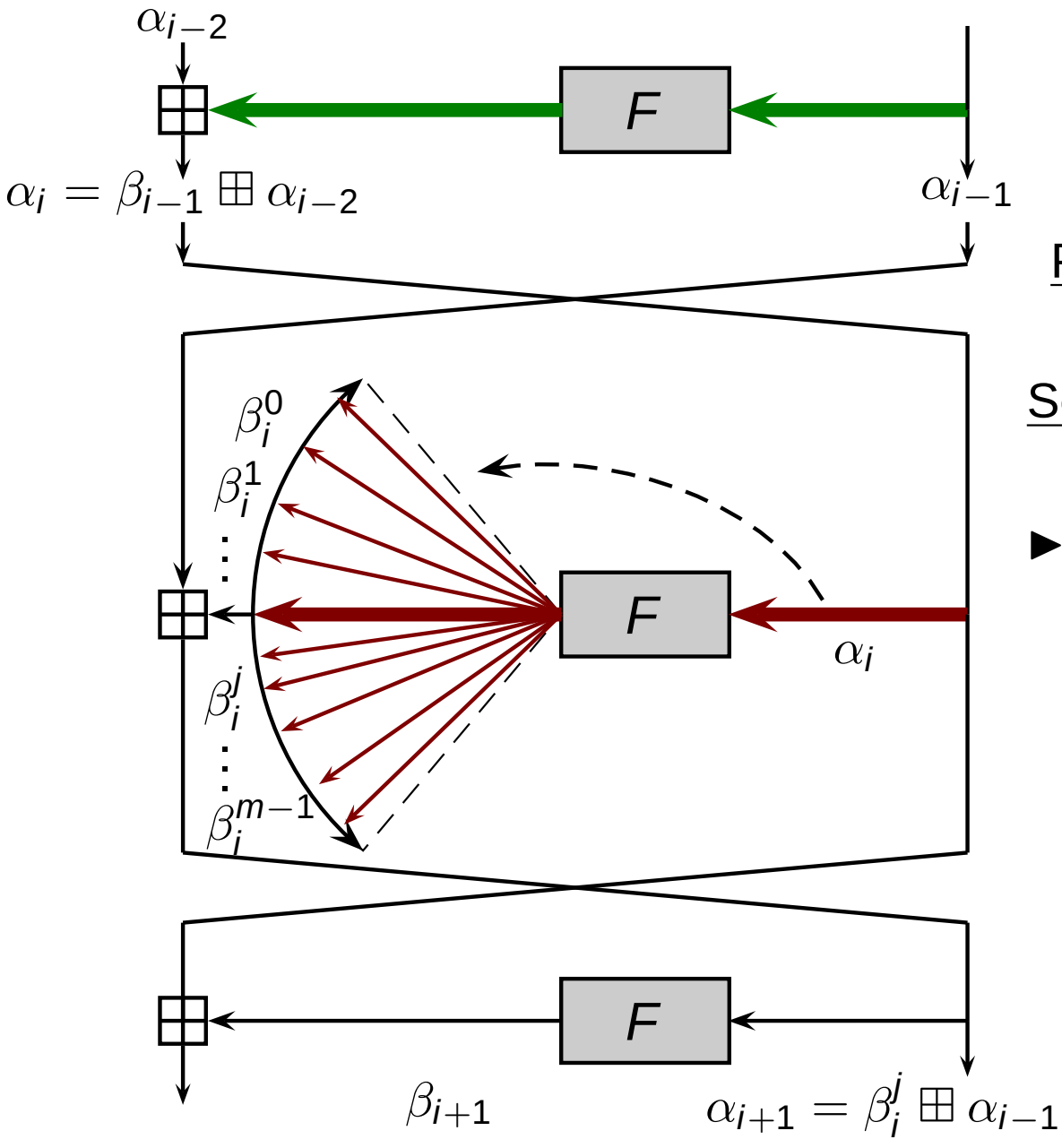
- Terminate and return $\perp$



Problem: $\nexists \beta : (\alpha_i, \beta) \in D$

Solution 2: Greedy choice:

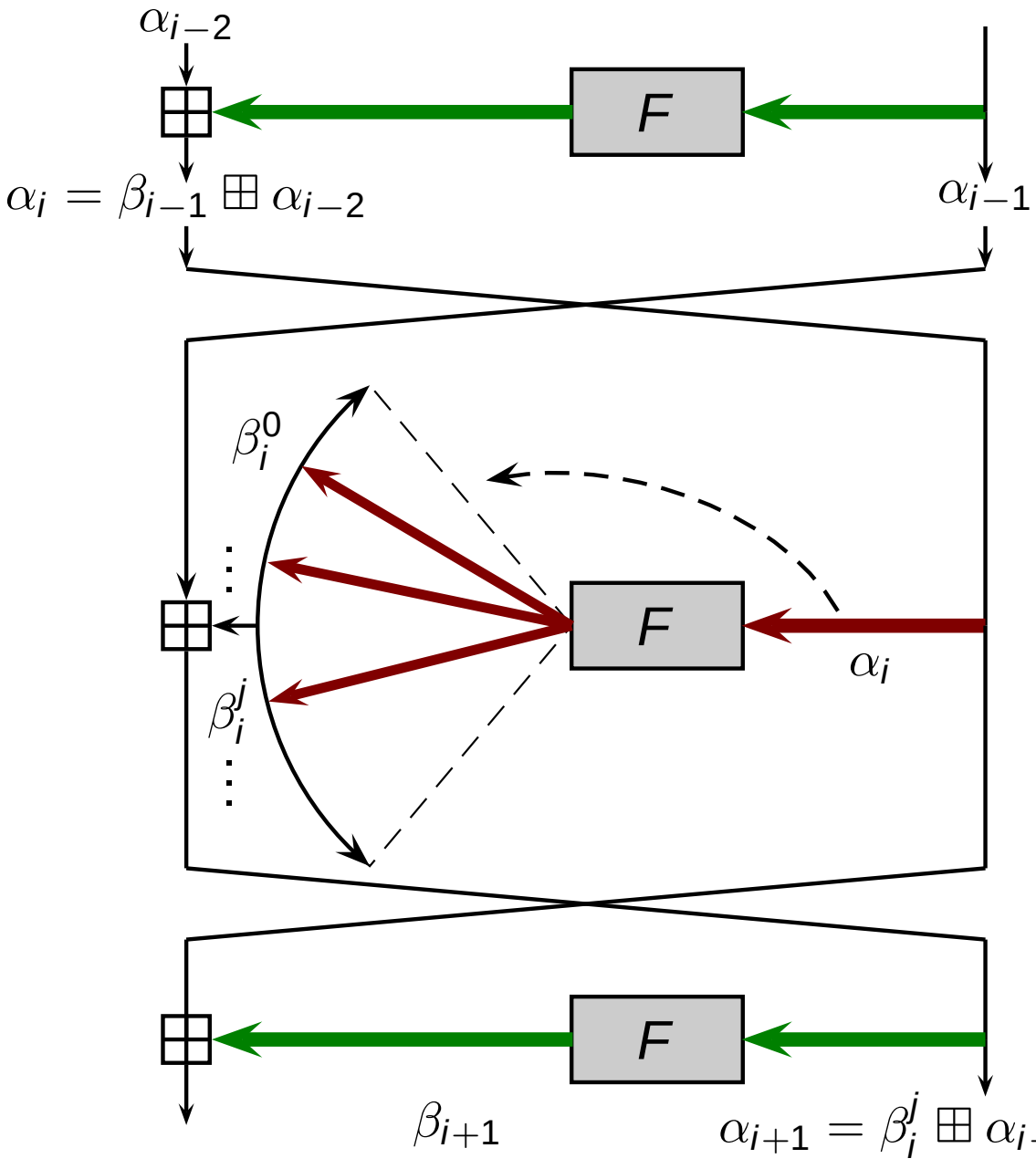
- $\beta_i^{\max} : p_i = \max_\beta p(\alpha_i \to \beta)$



Problem: $\nexists \beta : (\alpha_i, \beta) \in D$

Solution 3: Explore all β_i^j:

- $p(\alpha_i \to \beta_i^j) \geq \frac{\overline{B_n}}{p_1 p_2 \ldots p_{i-1} B_{n-i}}$



Problem: $\nexists \beta : (\alpha_i, \beta) \in D$

Solution 4: Explore some β_i^j:

- $p(\alpha_i \to \beta_i^j) \geq \frac{\overline{B_n}}{p_1 p_2 \ldots p_{i-1} B_{n-i}}$
- $\beta_i^j : (\alpha_{i+1}, \beta_{i+1}) \in D$

back-to-the-highway trick

Threshold Search

Application of Matsui's algorithm to ARX (threshold search):

1. Derive an expression for computing the DP of F.
2. Compute the pDDT of F (the highways table).
3. Execute the modified Matsui's algorithm with the pDDT as input.

Outline

UNIVERSITÉ DU LUXEMBOURG

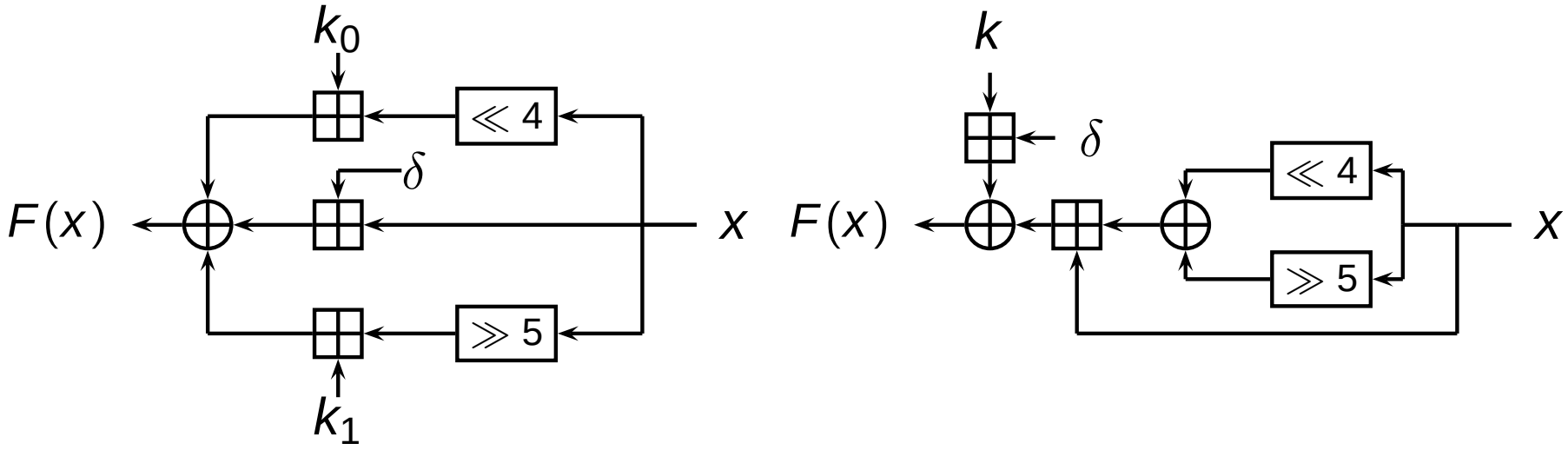


Figure: The F-functions of TEA (left) and XTEA (right).

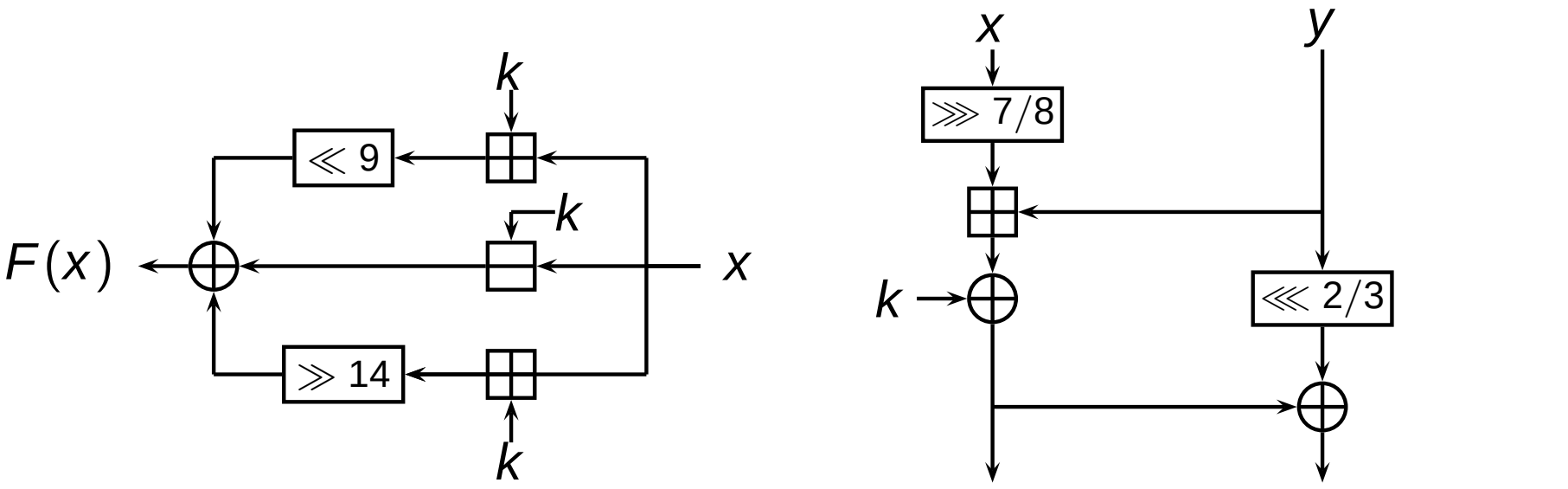


Figure: The F-functions of RAIDEN (left) and SPECK (right).

Results

Cipher	Type of Trail	#Rounds Covered	#Rounds Total	Ref.
TEA	Trunc.	5	64	[Moon02+]
	Trunc.	7		[Chen12+]
	Trunc.	8		[Hong03+, Bogdanov12+]
	Full	**18**		[**Sect. 8**]
XTEA	Trunc.	6	64	[Moon02+]
	Trunc.	7		[Chen12+]
	Trunc.	8		[Bogdanov12+]
	Full	**14**, 14		[**Sect. 8**], [Hong03+]
SPECK32	**Full**	**9**, 8*	22	[**Sect. 8**],[Abed13+]
SPECK48	**Full**	**10**, 10*	22/23	[**Sect. 8**], [Abed13+]
SPECK64	**Full**	**13**, 13*	26/27	[**Sect. 8**], [Abed13+]
RAIDEN	**Full**	**32**	32	**[Sect. 8]**

(*) differentials

Results: Update on SPECK (FSE 2014)

Cipher	Type of Trail	#Rounds Covered	#Rounds Total	Ref.
TEA	Trunc.	5	64	[Moon02+]
	Trunc.	7		[Chen12+]
	Trunc.	8		[Hong03+, Bogdanov12+]
	Full	**18**		[**Sect. 8**]
XTEA	Trunc.	6	64	[Moon02+]
	Trunc.	7		[Chen12+]
	Trunc.	8		[Bogdanov12+]
	Full	**14**, 14		[**Sect. 8**], [Hong03+]
SPECK32	**Full**	**9**, 8*	22	[**FSE '14**],[Abed13+]
SPECK48	**Full**	**11**, 10*	22/23	[**FSE '14**], [Abed13+]
SPECK64	**Full**	**14**, 13*	26/27	[**FSE '14**], [Abed13+]
RAIDEN	**Full**	**32**	32	**[Sect. 8]**

(*) differentials

Takeaway Message

Threshold search: first application of Matsui's algorithm to ARX.

The idea is very simple. It is the technique that's difficult.
– James Joyce on Ulysses

- Simple idea:
  - Matsui + pDDT + Highways and Country roads.
- Difficult technique:
  - **Choice of parameters**: p_{thres}, HW_{thres}, size of pDDT.
  - **pDDT:** complete vs. incomplete; pre-computed vs. dynamic update.
  - **Search strategy:** top-to-bottom vs. start-from-the-middle.
  - **Limit #CR:** back-to-highway (TEA) vs. limit-by-HW (SPECK).
- Depends on the cipher: **not a black-box tool to be applied as-is**.

uni.lu UNIVERSITÉ DU LUXEMBOURG

YAARX: Yet Another ARX Toolkit

General toolkit for analysis of ARX:

`https://github.com/vesselinux/yaarx`

Documentation:

`http://vesselinux.github.io/yaarx/index.html`

- Complements Gaëtan Leurent's **ARX Toolkit**.
- Extends **The S-function Toolkit** by Mouha et al.

Thank you for your attention!

uni.lu UNIVERSITÉ DU LUXEMBOURG

Questions


A
Questions?
B
Luxemburg City
Trier
Schweich
Newel
Echternach
Larochette
Junglinster
Lorentzweiler
Betzdorf
Konz
Pluwig
Wiltingen
Nittel
Ayl
Kell am See
Wadern
Freudenburg
Remich
Mettlach
Perl
Sierck-les-Bains
Waldrach
Heidenburg
Malborn

Backup Slides

Backup Slides

Monotonicity of the DP of XOR and ADD

Proposition

The differential probabilities (DP) of XOR and ADD (resp. xdp^{+} *and* $\mathrm{adp}^{\oplus}$*) are monotonously decreasing with the word size n of the differences* α, β, γ*:*

$$p_n \leq \ldots \leq p_{k+1} \leq p_k \leq p_{k-1} \leq \ldots \leq p_1 \ ,$$

where $p_k = \mathrm{DP}(\alpha_k, \beta_k \rightarrow \gamma_k) : \ n \geq k \geq 1$ *and* x_k *denotes the k LSB-s of the difference x.*

Corollary

For fixed p_{thres} *the pDDT of XOR (ADD) can be computed bitwise over the words of the differences from LSB to MSB.*

UNIVERSITÉ DU LUXEMBOURG

Bitwise Computation of pDDT for XOR and ADD

Algorithm 1 Compute pDDT for XOR (ADD).

Input: $n, p_{\text{thres}}, k, p_k, \alpha_k, \beta_k, \gamma_k$.

Output: Partial DDT D: $(\alpha, \beta, \gamma) \in D : \mathrm{DP}(\alpha, \beta \to \gamma) \geq p_{\text{thres}}$.

1: **if** $n = k$ **then**
2: Add $(\alpha, \beta, \gamma) \leftarrow (\alpha_k, \beta_k, \gamma_k)$ to D
3: **return**
4: **for** $x, y, x \in \{0, 1\}$ **do**
5: $\alpha_{k+1} \leftarrow x|\alpha_k, \quad \beta_{k+1} \leftarrow y|\beta_k, \quad \gamma_{k+1} \leftarrow z|\gamma_k \quad .$
6: $p_{k+1} = \mathrm{DP}(\alpha_{k+1}, \beta_{k+1} \to \gamma_{k+1})$
7: **if** $p_{k+1} \geq p_{\text{thres}}$ **then**
8: **Procedure 1**$(n, p_{\text{thres}}, k+1, p_{k+1}, \alpha_{k+1}, \beta_{k+1}, \gamma_{k+1})$

uni.lu UNIVERSITÉ DU LUXEMBOURG

Computation of Partial DDT: Timings, $n = 32$

	ADD		XOR	
$\mathbf{p}_{\text{thres}}$	**DDT size**	**Time**	**DDT size**	**Time**
0.1	$252,940$	36, *sec.*	$3,951,388$	2.29, *min.*
0.07	$361,420$	37, *sec.*	$3,951,388$	1.23, *min.*
0.05	$3,038,668$	5.35, *min.*	$167,065,948$	44.36, *min.*
0.01	$2,715,532,204$	17.46, *hours.*	–	–

TEA r	β		α	$\log_2 p$
1	F	$\leftarrow$	FFFFFFFF	-3.62
2	0	$\leftarrow$	0	-0.00
3	F	$\leftarrow$	FFFFFFFF	-2.87
4	0	$\leftarrow$	F	-7.90
5	FFFFFFF1	$\leftarrow$	FFFFFFFF	-3.60
6	0	$\leftarrow$	0	-0.00
7	FFFFFFF1	$\leftarrow$	FFFFFFFF	-2.78
8	2	$\leftarrow$	FFFFFFF1	-8.66
9	F	$\leftarrow$	1	-3.57
10	0	$\leftarrow$	0	-0.00
11	FFFFFFF1	$\leftarrow$	1	-2.87
12	FFFFFFFE	$\leftarrow$	FFFFFFF1	-7.90
13	F	$\leftarrow$	FFFFFFFF	-3.59
14	0	$\leftarrow$	0	-0.00
15	11	$\leftarrow$	FFFFFFFF	-2.79
16	0	$\leftarrow$	11	-8.83
17	FFFFFFEF	$\leftarrow$	FFFFFFFF	-3.61
18	0	$\leftarrow$	0	-0.00
$\sum_r \log_2 p_r$				-62.6
$\log_2 p_{\text{thres}}$				-4.32
#hways				68
Time:				21.36 min.

Copyright