Backstages of Anemoi:
A new approach to ZK-friendliness.

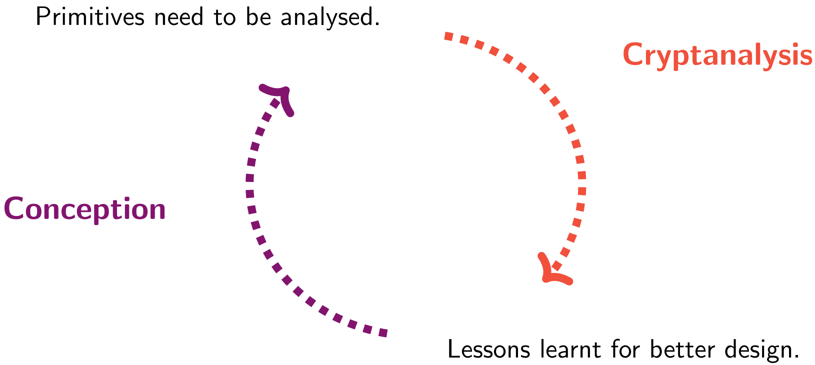
Clémence Bouvier [1,2]

joint work with Pierre Briaud[1,2], Pyrros Chaidos[3], Léo Perrin[2] and Vesselin Velichkov[4,5]

[1]Sorbonne Université, [2]Inria Paris,
[3]National & Kapodistrian University of Athens, [4]University of Edinburgh, [5]Clearmatics, London

August 29th, 2022

Motivation


Primitives need to be analysed.
Cryptanalysis
Conception
Lessons learnt for better design.

Motivation

Primitives need to be analysed.

Cryptanalysis

- ☞ Degree of MiMC [BCP22]
- ☞ Algebraic attacks [BBLP22]

Conception

Lessons learnt for better design.

Motivation

Primitives need to be analysed.

Cryptanalysis

- ☞ Degree of MiMC [BCP22]
- ☞ Algebraic attacks [BBLP22]

Conception

- ☞ Anemoi [BBC+22]

Lessons learnt for better designs.

A fast moving domain

Many primitives have already been proposed

- ⋆ MiMC / Feistel–MiMC [AGR+16]
- ⋆ *Rescue* / Rescue–Prime [AAB+20, SAD20]
- ⋆ Poseidon [GKR+21]
- ⋆ Reinforced Concrete [GKL+21]
- ⋆ Neptune [GOP+21]
- ⋆ Griffin [GHR+22]

Degree of MiMC

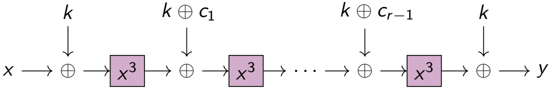
☞ On the Algebraic Degree of Iterated Power Functions, *Bouvier*, *Canteaut*, *Perrin*, submitted to DCC22

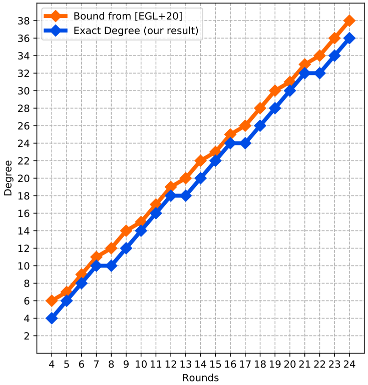
Definition

Algebraic degree of $F : \mathbb{F}_{2^n} \rightarrow \mathbb{F}_{2^n}$:

$$\deg_a(F) = \max\{wt(i),\ 0 \leq i < 2^n,\ \text{and}\ \ b_i \neq 0\}$$

MiMC$_3$ [AGR+16]:





Degree of MiMC

☞ On the Algebraic Degree of Iterated Power Functions, *Bouvier*, *Canteaut*, *Perrin*, submitted to DCC22

$$F : \mathbb{F}_{2^{11}} \rightarrow \mathbb{F}_{2^{11}}, x \mapsto x^3$$

$$F : \mathbb{F}_2^{11} \rightarrow \mathbb{F}_2^{11}, (x_0, \ldots, x_{10}) \mapsto$$

$$(x_0x_{10} + x_0 + x_1x_5 + x_1x_9 + x_2x_7 + x_2x_9 + x_2x_{10} + x_3x_4 + x_3x_5 + x_4x_8 + x_4x_9 + x_5x_{10} + x_6x_7 + x_6x_{10} + x_7x_8 + x_9x_{10},$$
$$x_0x_1 + x_0x_6 + x_2x_5 + x_2x_8 + x_3x_6 + x_3x_9 + x_3x_{10} + x_4 + x_5x_8 + x_5x_9 + x_6x_9 + x_7x_8 + x_7x_9 + x_7 + x_{10},$$
$$x_0x_1 + x_0x_2 + x_0x_{10} + x_1x_5 + x_1x_6 + x_1x_9 + x_2x_7 + x_3x_4 + x_3x_7 + x_4x_5 + x_4x_8 + x_4x_{10} + x_5x_{10} + x_6x_7 + x_6x_8 + x_6x_9 + x_7x_{10} + x_8 + x_9x_{10},$$
$$x_0x_3 + x_0x_6 + x_0x_7 + x_1 + x_2x_5 + x_2x_6 + x_2x_8 + x_2x_{10} + x_3x_6 + x_3x_8 + x_3x_9 + x_4x_5 + x_4x_6 + x_4 + x_5x_8 + x_5x_{10} + x_6x_9 + x_7x_9 + x_7 + x_8x_9 + x_{10},$$
$$x_0x_2 + x_0x_4 + x_1x_2 + x_1x_6 + x_1x_7 + x_2x_9 + x_2x_{10} + x_3x_5 + x_3x_6 + x_3x_7 + x_3x_9 + x_4x_5 + x_4x_7 + x_4x_9 + x_5 + x_6x_8 + x_7x_8 + x_8x_9 + x_8x_{10},$$
$$x_0x_5 + x_0x_7 + x_0x_8 + x_1x_2 + x_1x_3 + x_2x_6 + x_2x_7 + x_2x_{10} + x_3x_8 + x_4x_5 + x_4x_8 + x_5x_6 + x_5x_9 + x_7x_8 + x_7x_9 + x_7x_{10} + x_9,$$
$$x_0x_3 + x_0x_6 + x_1x_4 + x_1x_7 + x_1x_8 + x_2 + x_3x_6 + x_3x_7 + x_3x_9 + x_4x_7 + x_4x_9 + x_4x_{10} + x_5x_6 + x_5x_7 + x_5 + x_6x_9 + x_7x_{10} + x_8x_{10} + x_8 + x_9x_{10},$$
$$x_0x_7 + x_0x_8 + x_0x_9 + x_1x_3 + x_1x_5 + x_2x_3 + x_2x_7 + x_2x_8 + x_3x_{10} + x_4x_6 + x_4x_7 + x_4x_8 + x_4x_{10} + x_5x_6 + x_5x_8 + x_5x_{10} + x_6 + x_7x_9 + x_8x_9 + x_9x_{10},$$
$$x_0x_4 + x_0x_8 + x_1x_6 + x_1x_8 + x_1x_9 + x_2x_3 + x_2x_4 + x_3x_7 + x_3x_8 + x_4x_9 + x_5x_6 + x_5x_9 + x_6x_7 + x_6x_{10} + x_8x_9 + x_8x_{10} + x_{10},$$
$$x_0x_{10} + x_1x_4 + x_1x_7 + x_2x_5 + x_2x_8 + x_2x_9 + x_3 + x_4x_7 + x_4x_8 + x_4x_{10} + x_5x_8 + x_5x_{10} + x_6x_7 + x_6x_8 + x_6 + x_7x_{10} + x_9,$$
$$x_0x_5 + x_0x_{10} + x_1x_8 + x_1x_9 + x_1x_{10} + x_2x_4 + x_2x_6 + x_3x_4 + x_3x_8 + x_3x_9 + x_5x_7 + x_5x_8 + x_5x_9 + x_6x_7 + x_6x_9 + x_7 + x_8x_{10} + x_9x_{10}) .$$

Degree of MiMC

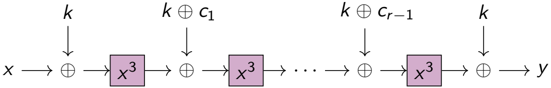
☞ On the Algebraic Degree of Iterated Power Functions, *Bouvier*, *Canteaut*, *Perrin*, submitted to DCC22

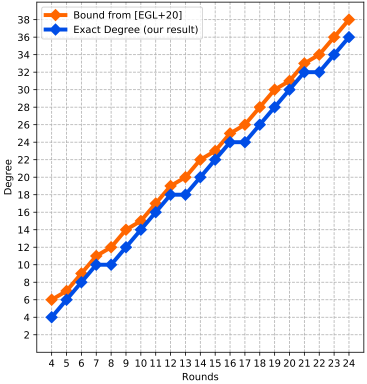
Definition

Algebraic degree of $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$:

$$\deg_a(F) = \max\{wt(i),\ 0 \leq i < 2^n,\ \text{and}\ \ b_i \neq 0\}$$

MiMC$_3$ [AGR+16]:





Degree of MiMC

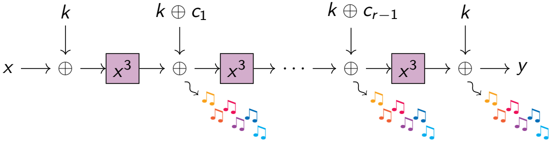
☞ On the Algebraic Degree of Iterated Power Functions, *Bouvier*, *Canteaut*, *Perrin*, submitted to DCC22

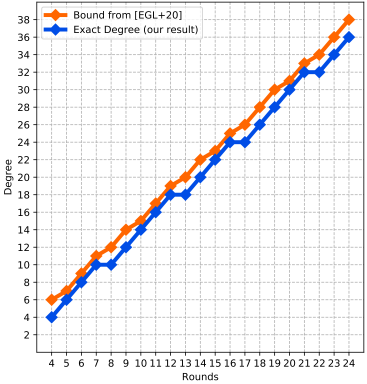
Definition

Algebraic degree of $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$:

$$\deg_a(F) = \max\{wt(i),\ 0 \leq i < 2^n,\ \text{and}\ \ b_i \neq 0\}$$

MiMC$_3$ [AGR+16]:





Degree of MiMC

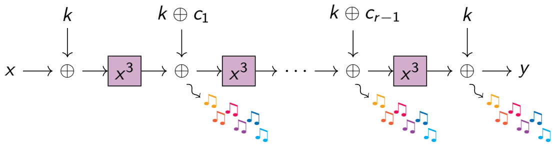
☞ On the Algebraic Degree of Iterated Power Functions, *Bouvier*, *Canteaut*, *Perrin*, submitted to DCC22

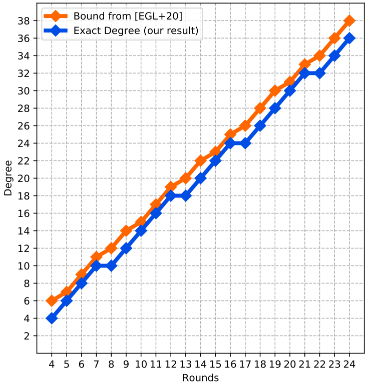
Definition

Algebraic degree of $F : \mathbb{F}_{2^n} \rightarrow \mathbb{F}_{2^n}$:

$$\deg_a(F) = \max\{wt(i),\ 0 \leq i < 2^n,\ \text{and}\ \ b_i \neq 0\}$$

MiMC$_3$ [AGR+16]:





Take Away

Concepts that are apparently quite simple have actually complex behaviours...

Algebraic attacks

☞ Algebraic Attacks against some Arithmetization-oriented Primitives,
Bariant, *Bouvier*, *Leurent*, *Perrin*, ToSC22(3) - to appear

Cryptanalysis Challenge for ZK-friendly Hash Functions!
In November 2021, by the Ethereum Foundation.

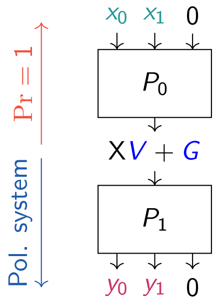
Definition

Constrained Input Constrained Output (CICO) problem:
Find $X, Y \in \mathbb{F}_q^{t-u}$ s.t. $P(X, 0^u) = (Y, 0^u)$.

Results on Feistel-MiMC, Poseidon and Rescue–Prime

⋆ build univariate systems

⋆ a trick for SPN



Algebraic attacks

☞ Algebraic Attacks against some Arithmetization-oriented Primitives, *Bariant*, *Bouvier*, *Leurent*, *Perrin*, ToSC22(3) - to appear

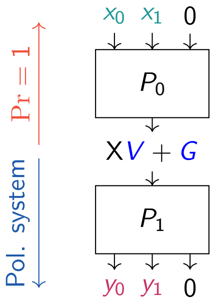
Cryptanalysis Challenge for ZK-friendly Hash Functions!
In November 2021, by the Ethereum Foundation.

Definition

Constrained Input Constrained Output (CICO) problem:
Find $X, Y \in \mathbb{F}_q^{t-u}$ s.t. $P(X, 0^u) = (Y, 0^u)$.

Results on Feistel-MiMC, Poseidon and Rescue–Prime

- ⋆ build univariate systems
- ⋆ a trick for SPN



Take Away

It might be better to avoid low degree functions...

Content

Backstages of Anemoi:
A new approach to ZK-friendliness.

1. Preliminaries
   - Emerging uses in symmetric cryptography
   - CCZ-equivalence

2. Anemoi
   - New S-box: Flystel
   - New Mode: Jive
   - Comparison to previous work

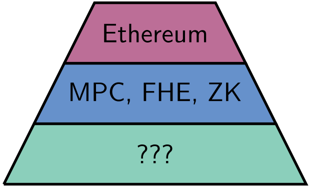
3. Conclusions and Future work

A need of new primitives

Problem: Designing new symmetric primitives

Protocols requiring new primitives:

- ⋆ Multiparty Computation (MPC)
- ⋆ Homomorphic Encryption (FHE)
- ⋆ Systems of Zero-Knowledge (ZK) proofs
  Example: SNARKs, STARKs, Bulletproofs

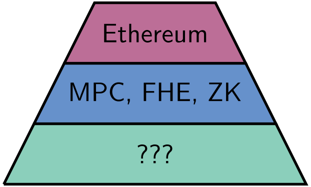


A need of new primitives

Problem: Designing new symmetric primitives

Protocols requiring new primitives:

- ⋆ Multiparty Computation (MPC)
- ⋆ Homomorphic Encryption (FHE)
- ⋆ Systems of Zero-Knowledge (ZK) proofs
  Example: SNARKs, STARKs, Bulletproofs



⇒ What differs from the "usual" case?

Comparison with "usual" case

A new environment

"Usual" case

- Field size:
  $\mathbb{F}_{2^n}$, with $n \simeq 4, 8$ (AES: $n = 8$).
- Operations:
  logical gates/CPU instructions

Arithmetization-friendly

- Field size:
  $\mathbb{F}_q$, with $q \in \{2^n, p\}, p \simeq 2^n$, $n \geq 64$.
- Operations:
  large finite-field arithmetic

Comparison with "usual" case

A new environment

"Usual" case

- Field size:
  $\mathbb{F}_{2^n}$, with $n \simeq 4, 8$ (AES: $n = 8$).
- Operations:
  logical gates/CPU instructions

Arithmetization-friendly

- Field size:
  $\mathbb{F}_q$, with $q \in \{2^n, p\}, p \simeq 2^n,\ n \geq 64$.
- Operations:
  large finite-field arithmetic

New properties

"Usual" case

- Operations:
  $$y \leftarrow E(x)$$
- Efficiency:
  implementation in software/hardware

Arithmetization-friendly

- Operations:
  $$y == E(x)$$
- Efficiency:
  integration within advanced protocols

Comparison with "usual" case

A new environment

"Usual" case

- Field size:
  $\mathbb{F}_{2^n}$, with $n \simeq 4, 8$ (AES: $n = 8$).
- Operations:
  logical gates/CPU instructions

Arithmetization-friendly

- Field size:
  $\mathbb{F}_q$, with $q \in \{2^n, p\}, p \simeq 2^n$, $n \geq 64$.
- Operations:
  large finite-field arithmetic

New properties

"Usual" case

- Operations:
  $$y \leftarrow E(x)$$
- Efficiency:
  implementation in software/hardware

Arithmetization-friendly

- Operations:
  $$y == E(x)$$
- Efficiency:
  integration within advanced protocols

Our approach

Need: verification using few multiplications.

Our approach

Need: verification using few multiplications.

First approach: evaluation also using few multiplications.

$\Rightarrow$ vulnerability to some attacks...

Our approach

Need: verification using few multiplications.

First approach: evaluation also using few multiplications.

$\Rightarrow$ vulnerability to some attacks...

New approach:

CCZ-equivalence

Our vision

A function is arithmetization-oriented if it is **CCZ-equivalent** to a function that can be verified efficiently.

Affine-equivalence

Definition

$F : \mathbb{F}_q \to \mathbb{F}_q$ and $G : \mathbb{F}_q \to \mathbb{F}_q$ are **affine equivalent** if

$$F(x) = (B \circ G \circ A)(x) ,$$

where A, B are affine permutations.

Definition

$F : \mathbb{F}_q \to \mathbb{F}_q$ and $G : \mathbb{F}_q \to \mathbb{F}_q$ are **extended affine equivalent** if

$$F(x) = (B \circ G \circ A)(x) + C(x) ,$$

where A, B, C are affine functions with A, B permutations s.t.

$$\Gamma_F = \{ (x, F(x)) \mid x \in \mathbb{F}_q \} = \begin{pmatrix} A^{-1} & 0 \\ CA^{-1} & B \end{pmatrix} \{ (x, G(x)) \mid x \in \mathbb{F}_q \} ,$$

CCZ-equivalence

Definition

$F : \mathbb{F}_q \to \mathbb{F}_q$ and $G : \mathbb{F}_q \to \mathbb{F}_q$ are **extended affine equivalent** if

$$\Gamma_F = \left\{ (x, F(x)) \mid x \in \mathbb{F}_q \right\} = \begin{pmatrix} A^{-1} & 0 \\ CA^{-1} & B \end{pmatrix} \left\{ (x, G(x)) \mid x \in \mathbb{F}_q \right\} ,$$

CCZ-equivalence

Definition

$F : \mathbb{F}_q \to \mathbb{F}_q$ and $G : \mathbb{F}_q \to \mathbb{F}_q$ are **extended affine equivalent** if

$$\Gamma_F = \left\{ (x, F(x)) \mid x \in \mathbb{F}_q \right\} = \begin{pmatrix} A^{-1} & 0 \\ CA^{-1} & B \end{pmatrix} \left\{ (x, G(x)) \mid x \in \mathbb{F}_q \right\} ,$$

Definition [Carlet, Charpin, Zinoviev, DCC98]

$F : \mathbb{F}_q \to \mathbb{F}_q$ and $G : \mathbb{F}_q \to \mathbb{F}_q$ are **CCZ-equivalent** if

$$\Gamma_F = \left\{ (x, F(x)) \mid x \in \mathbb{F}_q \right\} = \mathcal{A}(\Gamma_G) = \left\{ \mathcal{A}(x, G(x)) \mid x \in \mathbb{F}_q \right\} ,$$

where $\mathcal{A}$ is an affine permutation, $\mathcal{A}(x) = \mathcal{L}(x) + c$.

CCZ-equivalence

Definition

$F : \mathbb{F}_q \to \mathbb{F}_q$ and $G : \mathbb{F}_q \to \mathbb{F}_q$ are **extended affine equivalent** if

$$\Gamma_F = \{(x, F(x)) \mid x \in \mathbb{F}_q\} = \begin{pmatrix} A^{-1} & 0 \\ CA^{-1} & B \end{pmatrix} \{(x, G(x)) \mid x \in \mathbb{F}_q\},$$

Definition [Carlet, Charpin, Zinoviev, DCC98]

$F : \mathbb{F}_q \to \mathbb{F}_q$ and $G : \mathbb{F}_q \to \mathbb{F}_q$ are **CCZ-equivalent** if

$$\Gamma_F = \{(x, F(x)) \mid x \in \mathbb{F}_q\} = \mathcal{A}(\Gamma_G) = \{\mathcal{A}(x, G(x)) \mid x \in \mathbb{F}_q\},$$

where $\mathcal{A}$ is an affine permutation, $\mathcal{A}(x) = \mathcal{L}(x) + c$.

- ⋆ EA-equivalence and CCZ-equivalence preserve differential and linear properties,

$$\delta_G(a, b) = \delta_F(\mathcal{L}^{-1}(a, b)) \quad \text{and} \quad \mathcal{W}_G(\alpha, \beta) = (-1)^{c\cdot(\alpha,\beta)}\mathcal{W}_F(\mathcal{L}^T(\alpha, \beta))$$

- ⋆ EA-equivalence preserves the degree BUT CCZ-equivalence does not!

CCZ-equivalence

Definition

$F : \mathbb{F}_q \to \mathbb{F}_q$ and $G : \mathbb{F}_q \to \mathbb{F}_q$ are **extended affine equivalent** if

$$\Gamma_F = \{ (x, F(x)) \mid x \in \mathbb{F}_q \} = \begin{pmatrix} A^{-1} & 0 \\ CA^{-1} & B \end{pmatrix} \{ (x, G(x)) \mid x \in \mathbb{F}_q \},$$

Definition [Carlet, Charpin, Zinoviev, DCC98]

$F : \mathbb{F}_q \to \mathbb{F}_q$ and $G : \mathbb{F}_q \to \mathbb{F}_q$ are **CCZ-equivalent** if

$$\Gamma_F = \{ (x, F(x)) \mid x \in \mathbb{F}_q \} = \mathcal{A}(\Gamma_G) = \{ \mathcal{A}(x, G(x)) \mid x \in \mathbb{F}_q \},$$

where $\mathcal{A}$ is an affine permutation, $\mathcal{A}(x) = \mathcal{L}(x) + c$.

- ⋆ EA-equivalence and CCZ-equivalence preserve differential and linear properties,

$$\delta_G(a, b) = \delta_F(\mathcal{L}^{-1}(a, b)) \quad \text{and} \quad \mathcal{W}_G(\alpha, \beta) = (-1)^{c \cdot (\alpha, \beta)} \mathcal{W}_F(\mathcal{L}^T(\alpha, \beta))$$

- ⋆ EA-equivalence preserves the degree BUT CCZ-equivalence does not!

$\Rightarrow$ **Can we get CCZ-equivalence from EA-equivalence?**

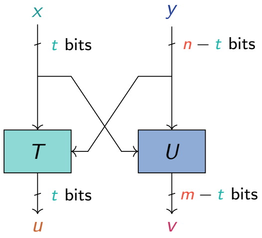
Twist

Using isomorphisms $\mathbb{F}_2^n \simeq \mathbb{F}_2^t \times \mathbb{F}_2^{n-t}$ and $\mathbb{F}_2^m \simeq \mathbb{F}_2^t \times \mathbb{F}_2^{m-t}$:

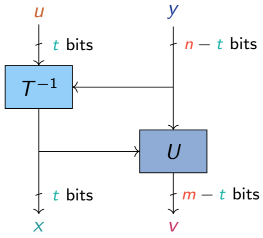
Definition

$F : \mathbb{F}_2^t \times \mathbb{F}_2^{n-t} \to \mathbb{F}_2^t \times \mathbb{F}_2^{m-t}$ and $G : \mathbb{F}_2^t \times \mathbb{F}_2^{n-t} \to \mathbb{F}_2^t \times \mathbb{F}_2^{m-t}$ are **t-twist-equivalent** if T_y is a permutation for all y and

$$G(u, y) = (T_y^{-1}(u), U_{T_y^{-1}(u)}(y)) .$$



t-twist $\Longleftrightarrow$



$\Gamma_F = \{(x, F(x)) \mid x \in \mathbb{F}_2^n\}$ swap matrix M_t $\Longleftrightarrow$ $\Gamma_G = \{(x, G(x)) \mid x \in \mathbb{F}_2^n\}$

CCZ = EA + twist

Theorem [Canteaut, Perrin, FFA19]

Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ and $G : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be two CCZ-equivalent functions. We can obtain G from F or F from G by composing:

$$\text{EA transformation} + t\text{-twist} + \text{EA transformation} .$$

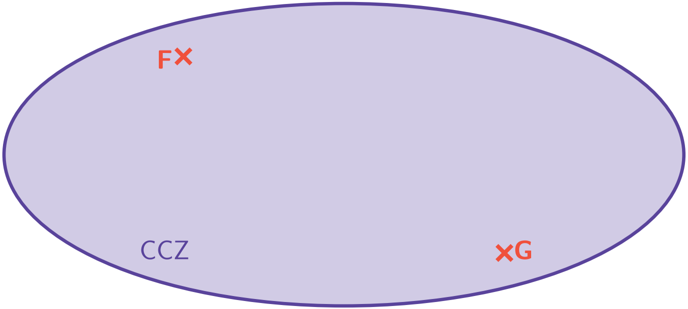
$$\Gamma_F = \mathcal{A}(\Gamma_G) ,$$

with $\mathcal{A}$ affine permutation.

$$\Downarrow$$

$$\Gamma_F = (A \cdot M_t \cdot B)(\Gamma_G) ,$$

with M_t swap matrix and A, B EA-mappings.



CCZ = EA + twist

Theorem [Canteaut, Perrin, FFA19]

Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ and $G : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be two CCZ-equivalent functions. We can obtain G from F or F from G by composing:

$$\text{EA transformation} + t\text{-twist} + \text{EA transformation}\ .$$

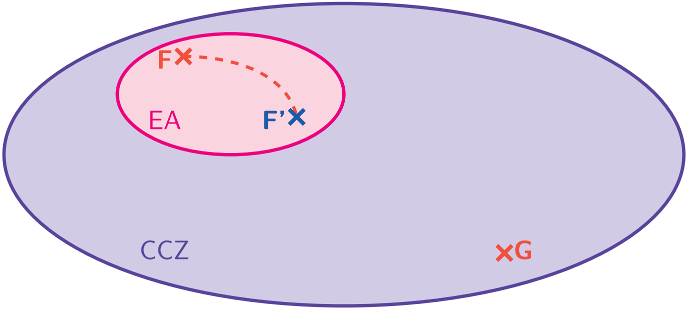
$$\Gamma_F = \mathcal{A}(\Gamma_G)\ ,$$

with $\mathcal{A}$ affine permutation.

$$\Downarrow$$

$$\Gamma_F = (A \cdot M_t \cdot B)(\Gamma_G)\ ,$$

with M_t swap matrix
and A, B EA-mappings.



CCZ = EA + twist

Theorem [Canteaut, Perrin, FFA19]

Let $F : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ and $G : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ be two CCZ-equivalent functions. We can obtain G from F or F from G by composing:

$$\text{EA transformation} + t\text{-twist} + \text{EA transformation} .$$

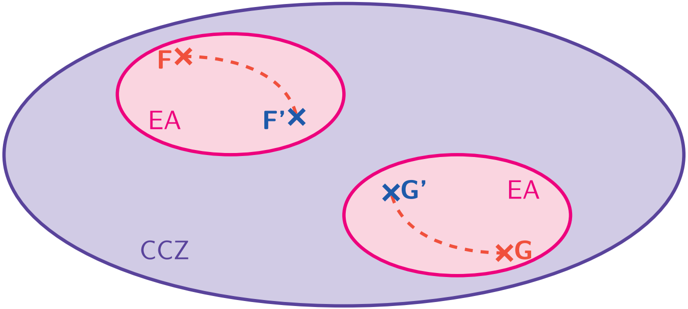
$$\Gamma_F = \mathcal{A}(\Gamma_G) ,$$

with $\mathcal{A}$ affine permutation.

$$\Downarrow$$

$$\Gamma_F = (A \cdot M_t \cdot B)(\Gamma_G) ,$$

with M_t swap matrix
and A, B EA-mappings.



CCZ = EA + twist

Theorem [Canteaut, Perrin, FFA19]

Let $F : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ and $G : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ be two CCZ-equivalent functions. We can obtain G from F or F from G by composing:

$$\text{EA transformation} + t\text{-twist} + \text{EA transformation} .$$

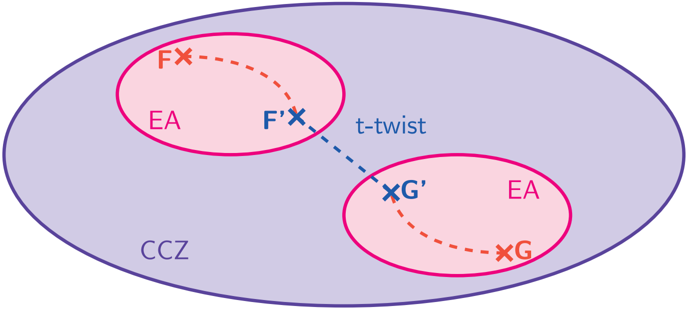
$$\Gamma_F = \mathcal{A}(\Gamma_G) ,$$

with $\mathcal{A}$ affine permutation.

$$\Downarrow$$

$$\Gamma_F = (A \cdot M_t \cdot B)(\Gamma_G) ,$$

with M_t swap matrix and A, B EA-mappings.

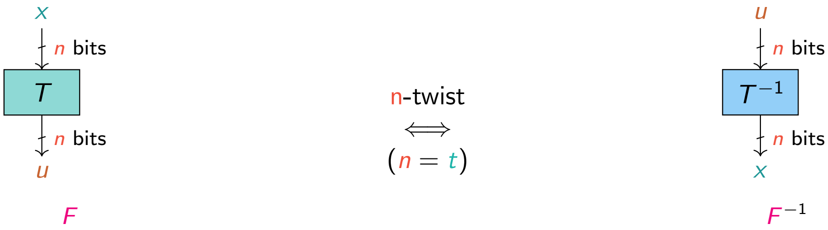


Example: Inverse

Let $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$,

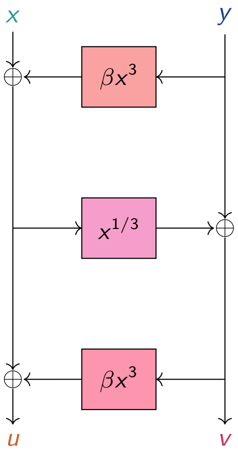
$$\Gamma_F = \left\{ (x, F(x)) \mid x \in \mathbb{F}_{2^n} \right\} \quad \text{and} \quad \Gamma_{F^{-1}} = \left\{ \left(y, F^{-1}(y)\right) \mid y \in \mathbb{F}_{2^n} \right\} = \left\{ (F(x), x) \mid x \in \mathbb{F}_{2^n} \right\}.$$

$$\begin{pmatrix} x \\ F(x) \end{pmatrix} = \begin{pmatrix} 0 & I_n \\ I_n & 0 \end{pmatrix} \begin{pmatrix} F(x) \\ x \end{pmatrix} \quad \Rightarrow \quad \text{swap matrix } M_n = \begin{pmatrix} 0 & I_n \\ I_n & 0 \end{pmatrix}.$$



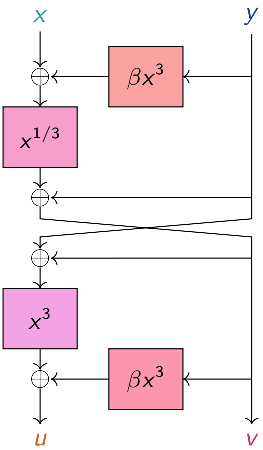
$\Rightarrow$ F and F^{-1} are CCZ-equivalent and the degree is indeed not preserved.

Example: Butterfly [PUB16]

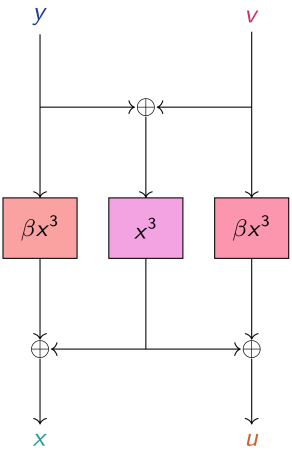


F

$\rightarrow$

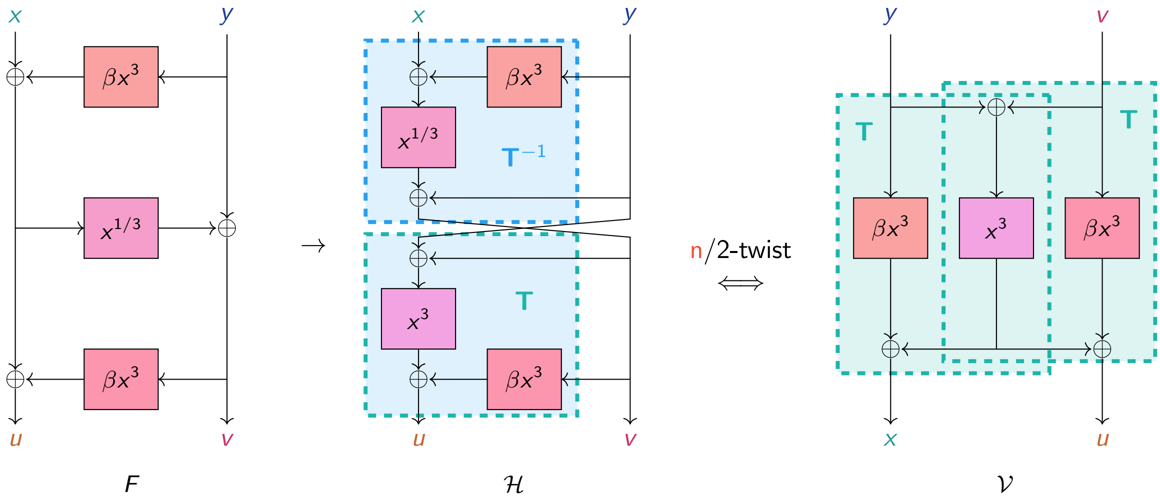


$\mathcal{H}$



$\mathcal{V}$

Example: Butterfly [PUB16]



Sum up on CCZ-equivalence

Important things to remember!

Let $F : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ and $G : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ s.t. $\Gamma_G = \mathcal{A}(\Gamma_F)$, with $\mathcal{A}(x) = \mathcal{L}(x) + c$.

- ⋆ F and G have the same differential properties

$$\delta_G(a, b) \;=\; \delta_F(\mathcal{L}^{-1}(a, b)) \;.$$

Sum up on CCZ-equivalence

Important things to remember!

Let $F : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ and $G : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ s.t. $\Gamma_G = \mathcal{A}(\Gamma_F)$, with $\mathcal{A}(x) = \mathcal{L}(x) + c$.

- ⋆ F and G have the same differential properties

$$\delta_G(a, b) \;=\; \delta_F(\mathcal{L}^{-1}(a, b)) \;.$$

- ⋆ F and G have the same linear properties

$$\mathcal{W}_G(\alpha, \beta) \;=\; (-1)^{c\cdot(\alpha,\beta)}\mathcal{W}_F(\mathcal{L}^T(\alpha, \beta)) \;.$$

Sum up on CCZ-equivalence

Important things to remember!

Let $F : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ and $G : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ s.t. $\Gamma_G = \mathcal{A}(\Gamma_F)$, with $\mathcal{A}(x) = \mathcal{L}(x) + c$.

- ⋆ F and G have the same differential properties

$$\delta_G(a, b) \ = \ \delta_F(\mathcal{L}^{-1}(a, b)) \ .$$

- ⋆ F and G have the same linear properties

$$\mathcal{W}_G(\alpha, \beta) \ = \ (-1)^{c \cdot (\alpha, \beta)} \mathcal{W}_F(\mathcal{L}^T(\alpha, \beta)) \ .$$

- ⋆ Verification is the same: if $y \leftarrow F(x)$, $v \leftarrow G(u)$

$$y == F(x)? \quad \Longleftrightarrow \quad v == G(u)?$$

Sum up on CCZ-equivalence

Important things to remember!

Let $F : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ and $G : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ s.t. $\Gamma_G = \mathcal{A}(\Gamma_F)$, with $\mathcal{A}(x) = \mathcal{L}(x) + c$.

- ⋆ F and G have the same differential properties

$$\delta_G(a, b) \;=\; \delta_F(\mathcal{L}^{-1}(a, b)) \;.$$

- ⋆ F and G have the same linear properties

$$\mathcal{W}_G(\alpha, \beta) \;=\; (-1)^{c\cdot(\alpha,\beta)}\mathcal{W}_F(\mathcal{L}^T(\alpha, \beta)) \;.$$

- ⋆ Verification is the same: if $y \leftarrow F(x)$, $v \leftarrow G(u)$

$$y == F(x)? \quad \iff \quad v == G(u)?$$

- ⋆ The degree is not preserved.

Sum up on CCZ-equivalence

Important things to remember!

Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ and $G : \mathbb{F}_2^n \to \mathbb{F}_2^m$ s.t. $\Gamma_G = \mathcal{A}(\Gamma_F)$, with $\mathcal{A}(x) = \mathcal{L}(x) + c$.

- ⋆ F and G have the same differential properties

$$\delta_G(a, b) = \delta_F(\mathcal{L}^{-1}(a, b)) .$$

- ⋆ F and G have the same linear properties

$$\mathcal{W}_G(\alpha, \beta) = (-1)^{c \cdot (\alpha, \beta)} \mathcal{W}_F(\mathcal{L}^T(\alpha, \beta)) .$$

- ⋆ Verification is the same: if $y \leftarrow F(x)$, $v \leftarrow G(u)$

$$y == F(x)? \iff v == G(u)?$$

- ⋆ The degree is not preserved.

1. Preliminaries
   - Emerging uses in symmetric cryptography
   - CCZ-equivalence

2. Anemoi
   - New S-box: Flystel
   - New Mode: Jive
   - Comparison to previous work

3. Conclusions and Future work

Goals and Principles

- Design goals:
  - Compatibility with Various Proof Systems.
  - Limited Reliance on Randomness.
  - Different Algorithms for Different Purposes.
  - Design Consistency.

Goals and Principles

- Design goals:
  - Compatibility with Various Proof Systems. $\rightarrow$ R1CS, Plonk, AIR, ...
  - Limited Reliance on Randomness.
  - Different Algorithms for Different Purposes.
  - Design Consistency.

Goals and Principles

- Design goals:
  - Compatibility with Various Proof Systems. → R1CS, Plonk, AIR, ...
  - Limited Reliance on Randomness. → fixed MDS matrices
  - Different Algorithms for Different Purposes.
  - Design Consistency.

Goals and Principles

- Design goals:
  - Compatibility with Various Proof Systems. → R1CS, Plonk, AIR, ...
  - Limited Reliance on Randomness. → fixed MDS matrices
  - Different Algorithms for Different Purposes. → hash function $\neq$ compression function
  - Design Consistency.

Goals and Principles

- Design goals:
  - Compatibility with Various Proof Systems. → R1CS, Plonk, AIR, ...
  - Limited Reliance on Randomness. → fixed MDS matrices
  - Different Algorithms for Different Purposes. → hash function $\neq$ compression function
  - Design Consistency. → same structure for all uses

Goals and Principles

- Design goals:
  - Compatibility with Various Proof Systems. → R1CS, Plonk, AIR, ...
  - Limited Reliance on Randomness. → fixed MDS matrices
  - Different Algorithms for Different Purposes. → hash function $\neq$ compression function
  - Design Consistency. → same structure for all uses
- Our contributions:
  - Link between AO and CCZ-equivalence
  - Flystel: a new S-box
  - Jive: a new mode

Why Anemoi?

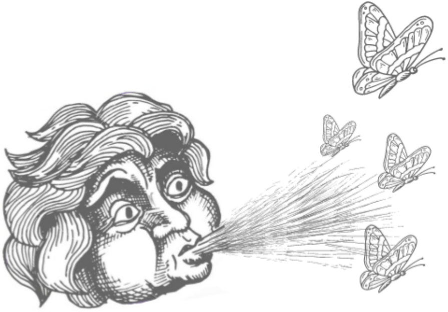
- **Auld**
  Alliance between France and Scotland

Why Anemoi?

- ~~Auld~~
  ~~Alliance between France and Scotland~~
- Athena
  Greek goddess, protector of Athens

Why Anemoi?

- ~~Auld~~
  ~~Alliance between France and Scotland~~
- ~~Athena~~
  ~~Greek goddess, protector of Athens~~
- Anemoi
  Greek gods of winds

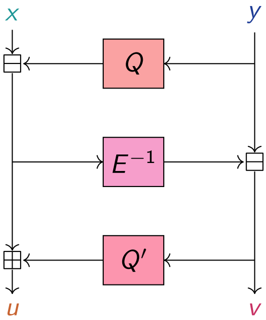


The Flystel

Butterfly + Feistel ⇒ Flystel

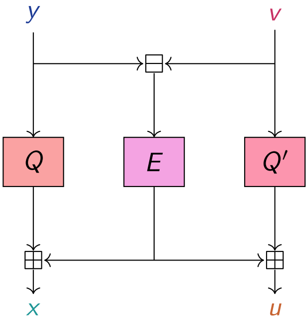
A 3-round Feistel-network with
$Q : \mathbb{F}_q \rightarrow \mathbb{F}_q$ and $Q' : \mathbb{F}_q \rightarrow \mathbb{F}_q$ two quadratic functions, and $E : \mathbb{F}_q \rightarrow \mathbb{F}_q$ a permutation

High-degree permutation



Open Flystel $\mathcal{H}$.

Low-degree function

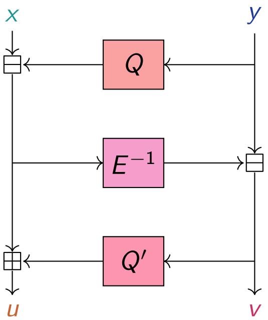


Closed Flystel $\mathcal{V}$.

The Flystel

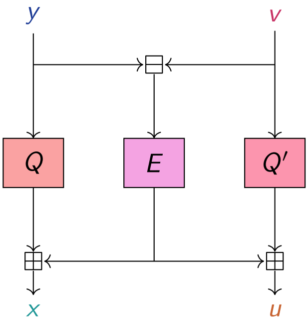
$$\begin{aligned}\Gamma_{\mathcal{H}} &= \{(\ (x,y),\ \mathcal{H}((x,y))\ )\mid (x,y)\in\mathbb{F}_q^2\}\\ &= \mathcal{A}\left(\{(\ (v,y),\ \mathcal{V}((v,y))\ )\mid (v,y)\in\mathbb{F}_q^2\}\right)\\ &= \mathcal{A}(\Gamma_{\mathcal{V}})\end{aligned}$$

High-degree permutation



Open `Flystel` $\mathcal{H}$.

Low-degree function

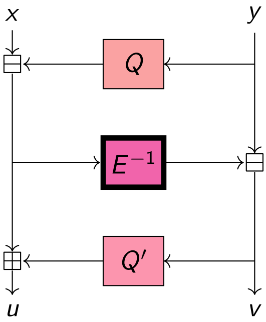


Closed `Flystel` $\mathcal{V}$.

Advantage of CCZ-equivalence

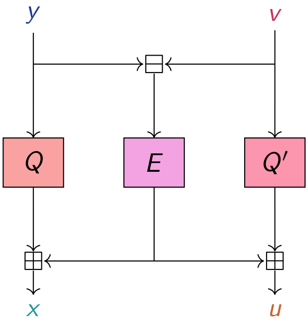
- ⋆ High Degree Evaluation.

High-degree permutation



Open Flystel $\mathcal{H}$.

Low-degree function



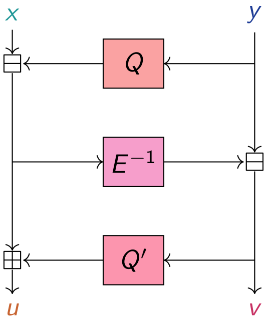
Closed Flystel $\mathcal{V}$.

Advantage of CCZ-equivalence

- ⋆ High Degree Evaluation.
- ⋆ Low Cost Verification.

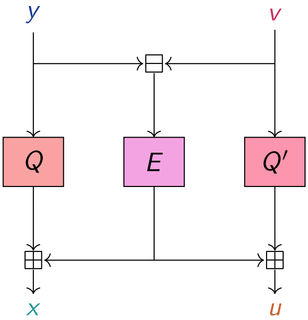
$$(u, v) == \mathcal{H}(x, y) \Leftrightarrow (x, u) == \mathcal{V}(y, v)$$

High-degree permutation



Open `Flystel` $\mathcal{H}$.

Low-degree function

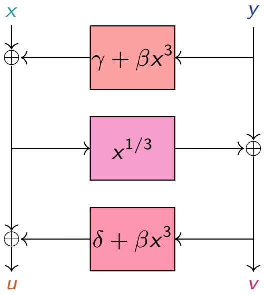


Closed `Flystel` $\mathcal{V}$.

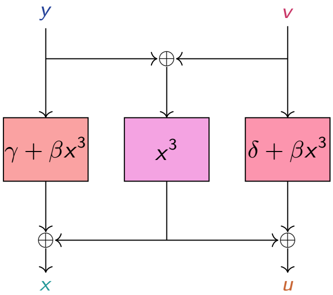
Flystel in $\mathbb{F}_{2^n}$

$$\mathcal{H} : \begin{cases} \mathbb{F}_{2^n} \times \mathbb{F}_{2^n} & \to \mathbb{F}_{2^n} \times \mathbb{F}_{2^n} \\ (x, y) \mapsto & \left(x + \beta y^3 + \gamma + \beta\left(y + (x + \beta y^3 + \gamma)^{1/3}\right)^3 + \delta \right., \\ & \left. y + (x + \beta y^3 - \gamma)^{1/3}\right) . \end{cases}$$

$$\mathcal{V} : \begin{cases} \mathbb{F}_{2^n} \times \mathbb{F}_{2^n} & \to \mathbb{F}_{2^n} \times \mathbb{F}_{2^n} \\ (x, y) & \mapsto \left((y + v)^3 + \beta y^3 + \gamma\right., \\ & \left.(y + v)^3 + \beta v^3 + \delta\right) , \end{cases}$$

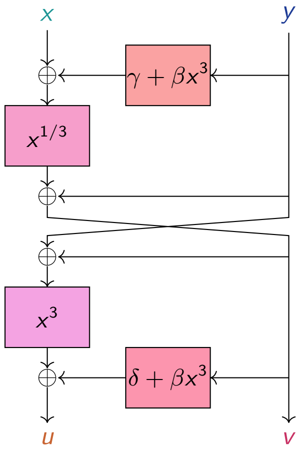


Open Flystel$_2$.



Closed Flystel$_2$.

Properties of Flystel in $\mathbb{F}_{2^n}$



Degenerated Butterfly.

First introduced by [Perrin et al. 2016].

Well-studied butterfly.

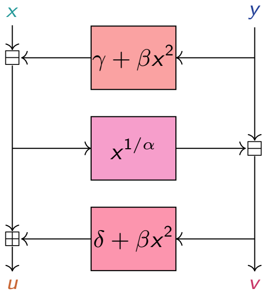
Theorems in [Li et al. 2018] state that if $\beta \neq 0$:

- Differential properties
  - Flystel$_2$: $\delta_{\mathcal{H}} = \delta_{\mathcal{V}} = 4$
- Linear properties
  - Flystel$_2$: $\mathcal{W}_{\mathcal{H}} = \mathcal{W}_{\mathcal{V}} = 2^{2n-1} - 2^n$
- Algebraic degree
  - Open Flystel$_2$: $\deg_{\mathcal{H}} = n$
  - Closed Flystel$_2$: $\deg_{\mathcal{V}} = 2$

Flystel in $\mathbb{F}_p$

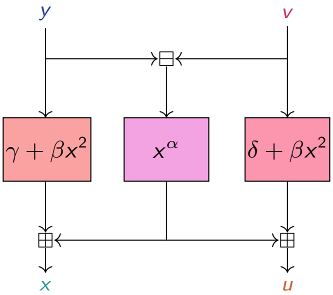
$$\mathcal{H} : \begin{cases} \mathbb{F}_p \times \mathbb{F}_p & \to \mathbb{F}_p \times \mathbb{F}_p \\ (x, y) & \mapsto \Big(x - \beta y^2 - \gamma + \beta \left(y - (x - \beta y^2 - \gamma)^{1/\alpha}\right)^2 + \delta \ , \\ & \quad y - (x - \beta y^2 - \gamma)^{1/\alpha}\Big) \ . \end{cases}$$

$$\mathcal{V} : \begin{cases} \mathbb{F}_p \times \mathbb{F}_p & \to \mathbb{F}_p \times \mathbb{F}_p \\ (y, v) & \mapsto \left((y - v)^\alpha + \beta y^2 + \gamma \ , \right. \\ & \quad \left. (v - y)^\alpha + \beta v^2 + \delta\right) \ . \end{cases}$$



Open $\texttt{Flystel}_p$.

usually
$\alpha = 3$ or 5.



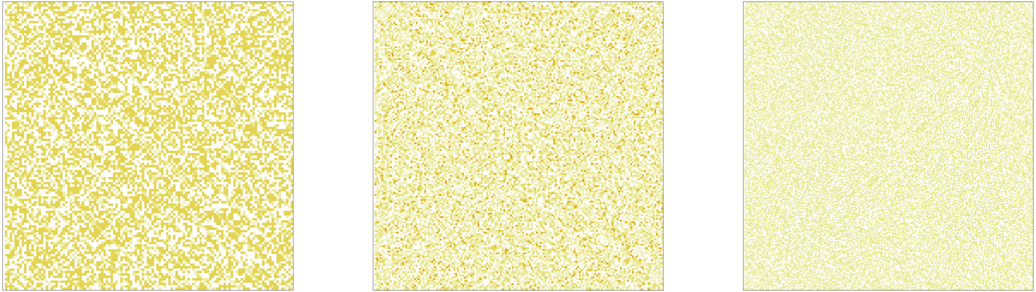
Closed $\texttt{Flystel}_p$.

Properties of Flystel in $\mathbb{F}_p$

- Differential properties
  $\texttt{Flystel}_\texttt{p}$ has a differential uniformity equals to $\alpha - 1$.

Properties of Flystel in $\mathbb{F}_p$

- ⋆ Differential properties
  $\texttt{Flystel}_p$ has a differential uniformity equals to $\alpha - 1$.



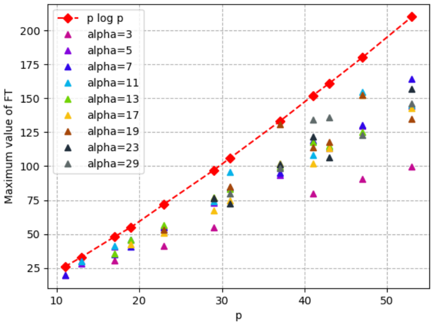
(a) *when* $p = 11$ *and* $\alpha = 3$. **(b)** *when* $p = 13$ *and* $\alpha = 5$. **(c)** *when* $p = 17$ *and* $\alpha = 3$.

DDT of $\texttt{Flystel}_p$.

Properties of Flystel in $\mathbb{F}_p$

- Linear properties

$$\mathcal{W} \leq p \log p \ ?$$

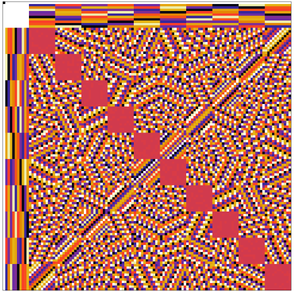


Conjecture for the linearity.

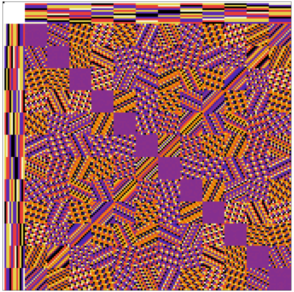
Properties of Flystel in $\mathbb{F}_p$

- Linear properties

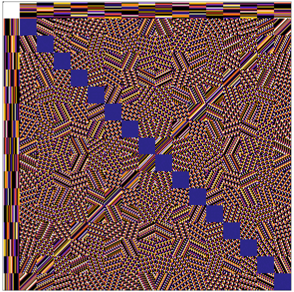
$$\mathcal{W} \leq p \log p \; ?$$



(a) *when* $p = 11$ *and* $\alpha = 3$.



(b) *when* $p = 13$ *and* $\alpha = 5$.

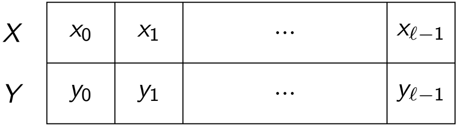


(c) *when* $p = 17$ *and* $\alpha = 3$.

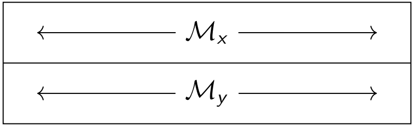
LAT of $\texttt{Flystel}_p$.

The SPN Structure

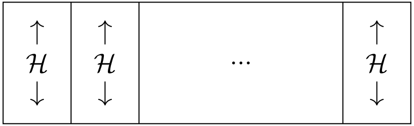
The internal state of Anemoi and its basic operations.



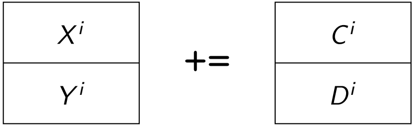
(a) *Internal state*



(b) *The diffusion layer $\mathcal{M}$.*

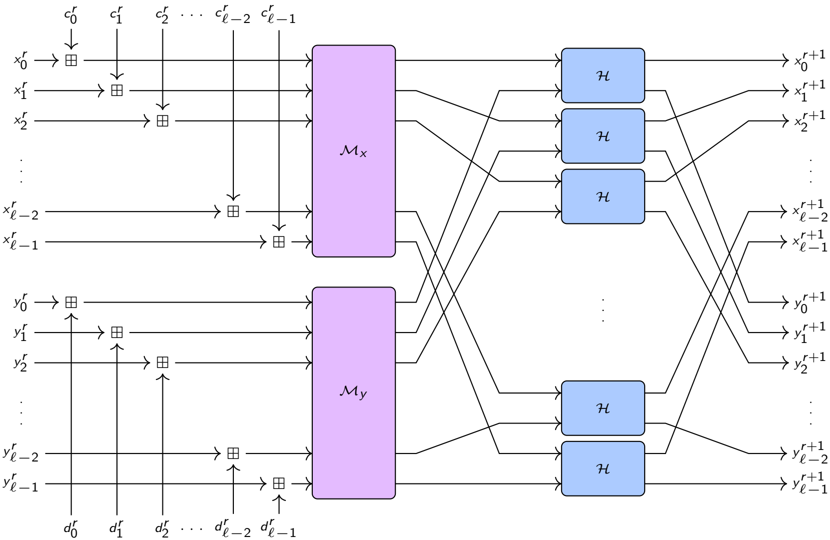


(c) *The S-box layer $\mathcal{S}$.*



(d) *The constant addition $\mathcal{A}$.*

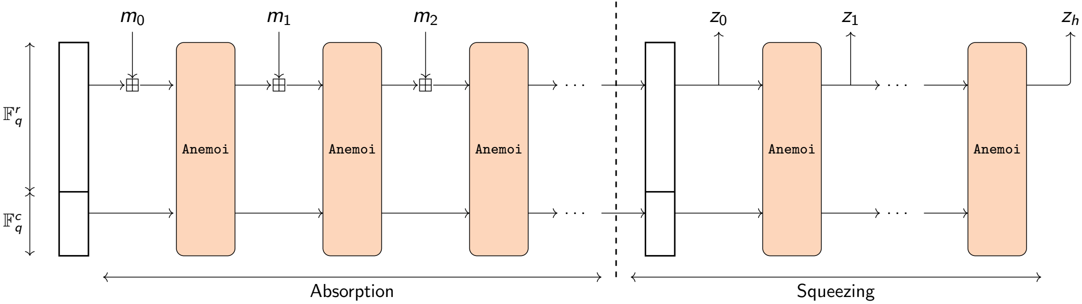
The SPN Structure



Overview of `Anemoi`*.*

New Mode

- Hash function:
  - input: arbitrary length
  - ouput: fixed length

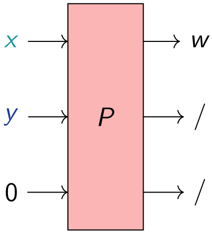


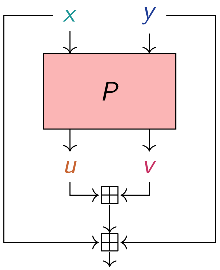
New Mode

- Hash function:
  - input: arbitrary length
  - ouput: fixed length
- Compression function:
  - input: fixed length
  - output: length 1

Dedicated mode ⇒ 2 words in 1

$$(x, y) \mapsto x + y + u + v \ .$$





Comparison for R1CS

SNARK performances using R1CS representation:

$\sim$ number of multiplications

m	*Rescue′*	Poseidon	Griffin	Anemoi
4	224	232	112	**96**
6	216	264	-	**120**
8	256	296	176	**160**

(a) *when* $\alpha = 3$.

m	*Rescue′*	Poseidon	Griffin	Anemoi
4	264	264	**110**	120
6	288	315	-	**150**
8	384	363	**162**	200

(b) *when* $\alpha = 5$.

R1CS constraints for Rescue–Prime, Poseidon, Griffin *and Anemoi*, $s = 128$, *and prime field of* 256 *bits.*

Comparison for Plonk

SNARK performances using Plonk representation:

$\sim$ multiplications gates + addition gates

m	*Rescue′*	POSEIDON	GRIFFIN	Anemoi
4	560	1336	334	**216**
6	756	3024	-	**330**
8	1152	5448	969	**520**

(a) *when* $\alpha = 3$.

m	*Rescue′*	POSEIDON	GRIFFIN	Anemoi
4	528	1032	287	**240**
6	768	2265	-	**360**
8	1280	4003	821	**560**

(b) *when* $\alpha = 5$.

Plonk constraints for Rescue–Prime, POSEIDON, GRIFFIN *and Anemoi,*
$s = 128$*, and prime field of* 256 *bits.*

Comparison for Plonk (with optimizations)

	m	Constraints
Poseidon	2	88
	3	110
Reinforced Concrete	2	236
	3	378
AnemoiJive	2	**79**

(a) *With 3 wires.*

	m	Constraints
Poseidon	2	82
	3	98
Reinforced Concrete	2	174
	3	267
AnemoiJive	2	**60**

(b) *With 4 wires.*

Constraints comparison with $\alpha = 5$, $s = 128$, *and prime field sizes of* $256, 384$.

Comparison for AIR

STARK performances using AIR representation:

$$w \cdot T \cdot d_{\max}$$

Here $w = m$, $d_{\max} = \alpha$, and $T = R$ (or $RF + \lceil RP/m \rceil$).

m	*Rescue'*	Poseidon	Griffin	Anemoi
4	168	348	168	**144**
6	**162**	396	-	180
8	**192**	480	264	240

(a) *with* $\alpha = 3$.

m	*Rescue'*	Poseidon	Griffin	Anemoi
4	**220**	440	**220**	240
6	**240**	540	-	300
8	**320**	640	360	400

(b) *with* $\alpha = 5$.

AIR constraints for Rescue–Prime, Poseidon, Griffin and Anemoi, $s = 128$, and prime field of 256 bits.

Conclusions

- A new family of ZK-friendly hash functions:
  ⇒ Anemoi efficient accross proof system
- New observations of fundamental interest:
  - Standalone components:
    - New S-box: Flystel
    - New mode: Jive
  - Identify a link between AO and CCZ-equivalence

Conclusions

- A new family of ZK-friendly hash functions:

  ⇒ Anemoi efficient accross proof system

- New observations of fundamental interest:
  - Standalone components:
    - New S-box: Flystel
    - New mode: Jive
  - Identify a link between AO and CCZ-equivalence

Cryptanalysis and designing of arithmetization-oriented primitives remain to be explored!

Future work

- On Anemoi:
  - pushing further the cryptanalysis.
  - explaining linear properties of the Flystel.
  - constructing a Flystel with more branches?
    $\Rightarrow$ see [BCLP22]

- Extending the study of the algebraic degree of MiMC to
  - other permutations x^d for any d.
  - SPN constructions.
    $\Rightarrow$ see [LAW+22]: can we extend the coefficient grouping strategy to other primitives than Chaghri?

Future work

- On Anemoi:
  - pushing further the cryptanalysis.
  - explaining linear properties of the Flystel.
  - constructing a Flystel with more branches?
    ⇒ see [BCLP22]
- Extending the study of the algebraic degree of MiMC to
  - other permutations x^d for any d.
  - SPN constructions.
    ⇒ see [LAW+22]: can we extend the coefficient grouping strategy to other primitives than Chaghri?

Thanks for your attention!