Cryptanalytic Properties of Mealy Machines

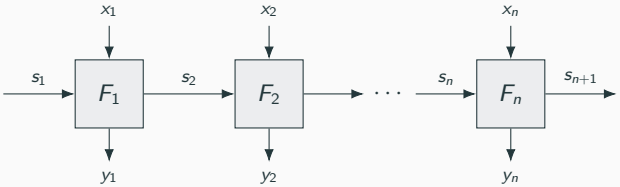
Zhongfeng Niu, Tim Beyne, Kai Hu, and Meiqin Wang
SDU, NTU, KU Leuven

Changsha, 2026

Outline

Mealy machines and S-functions

A **Mealy machine** produces output chunks $y_1, \ldots, y_n$ from input chunks $x_1, \ldots, x_n$, carried through a small **state** s_i of constant size:

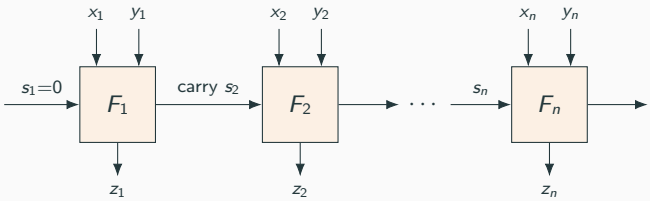


State-update function: $F_i \colon S_i \times X_i \to S_{i+1} \times Y_i$

S-function: $F = \text{S-FUNCTION}[F_1, \ldots, F_n]$

Example 1 – Modular addition $x \boxplus y \pmod{2^n}$

State. $s_i \in \mathbb{F}_2$ is the **carry** bit. (n positions; $s_1 = 0$.)



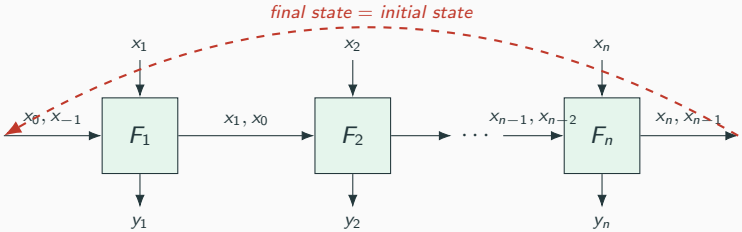
State-update function: $F_i \colon \mathbb{F}_2 \times \mathbb{F}_2^2 \to \mathbb{F}_2 \times \mathbb{F}_2$ (same for every i):

$$F_i\big(\underbrace{s_i}_{\text{carry in}};\ x_i, y_i\big) \;=\; \big(\underbrace{x_i y_i + (x_i + y_i)\, s_i}_{\text{carry out } s_{i+1}},\ \underbrace{x_i + y_i + s_i}_{\text{output bit } z_i}\big).$$

Example 2 – Daemen's χ-function on $\mathbb{F}_2^n$

Definition. $y_i = x_i + (x_{i-1} + 1)\, x_{i-2}$, with $x_0 = x_n$ and $x_{-1} = x_{n-1}$ (cyclic indexing).

State. Each bit y_i depends on x_i and the **previous two** input bits. So $s_i = (x_{i-1}, x_{i-2}) \in \mathbb{F}_2^2$ is the state.



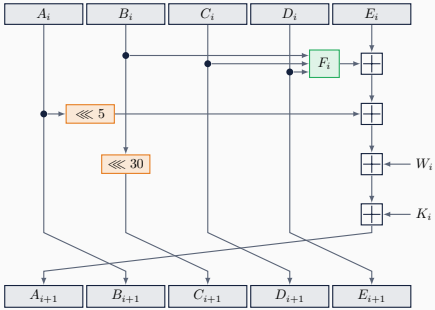
State-update function:

$$F_i\big(\underbrace{(x_{i-1}, x_{i-2})}_{s_i};\, x_i\big) \;=\; \big(\underbrace{(x_i, x_{i-1})}_{s_{i+1}},\, \underbrace{x_i + (x_{i-1} + 1)\, x_{i-2}}_{y_i}\big).$$

An S-function **with a cycle**; trace formula handles the boundary.

Example 3 – SHA-1 step function

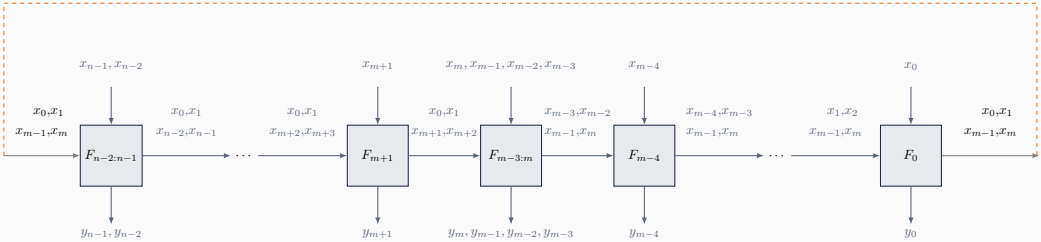
Step function adds 5 words of 32 bits with a Boolean mixing F_i, a message word W_i and constant K_i: $A_{i+1} = (A_i \lll 5) \boxplus F_i(B_i, C_i, D_i) \boxplus E_i \boxplus W_i \boxplus K_i$.



As an S-function (process bit by bit, $n = 32$): state $s_i = (c_0, c_1, c_2, c_3) \in \mathbb{F}_2^4$ collects the carries of the **four** chained additions $\boxplus$ on the right. Constant-size state $\Rightarrow$ S-function.

Example 4 – the $\chi\chi$-function (Belkheyar et al. '25)

Like χ, but the round function is **non-uniform**: different positions update different numbers of bits at once.



State. $s_i = (x_{i-1}, x_{i-2}) \in \mathbb{F}_2^2$ (two preceding input bits), like χ.

Cycle. Final state = initial state, so $s_{n+1} = s_1$ (dashed wrap).

Two decades of analysis – mostly modular addition

Year	Authors	Result
2001	Lipmaa, Moriai	Differential prob. of $\boxplus$
2002	Lipmaa–Wallén–Dumas	Additive diff. prob. of $\oplus$
2003	Wallén	Linear correlations of $\boxplus$
2005	Klimov–Shamir	T-functions / S-functions (general)
2010	Mouha–Velichkov–de Cannière–Preneel	General differential alg. for any S-function
2013	Schulte-Geers	CCZ-equivalent quadratic view of $\boxplus$
2020	Azimi et al.	Diff. of $x \boxplus c$ (constant addend)
2022	Beyne–Rijmen	Quasidifferential matrix of $\boxplus$
2023	Niu et al. / Wang et al.	DLCT, BCT of $\boxplus$

Each entry is a **custom** algorithm for a *specific* property of a *specific* S-function.

State of the art before this work

What we had

- For **modular addition**: linear, differential, quasidifferential, integral, DLCT, BCT – each from a different paper.
- For χ: linear and differential easy (it is quadratic); monomial prediction, BCT, ultrametric integral, DLCT, **unknown how to compute**.
- Mouha et al. gave a general algorithm **only** for differentials.

State of the art before this work

What we had

- For **modular addition**: linear, differential, quasidifferential, integral, DLCT, BCT – each from a different paper.
- For χ: linear and differential easy (it is quadratic); monomial prediction, BCT, ultrametric integral, DLCT, **unknown how to compute**.
- Mouha et al. gave a general algorithm **only** for differentials.

What we wanted

A **universal formula** that covers *any* cryptanalytic property of *any* S-function – linear, differential, integral, DL, boomerang, *d*-differential, ultrametric integral, ...

Replace a function $F : X \to Y$ by a **linear operator** on free vector spaces.

The free vector space $\mathbb{K}[X]$. Formal $\mathbb{K}$-linear combinations of elements of X, with basis $\{\delta_x : x \in X\}$. A generic element is

$$u = \sum_{x \in X} u[x]\, \delta_x, \qquad u[x] \in \mathbb{K}.$$

The dual space $\mathbb{K}^Y$. $\mathbb{K}$-valued functions on Y (= linear functions on $\mathbb{K}[Y]$). A generic element is $v : Y \to \mathbb{K}$, paired with $z \in \mathbb{K}[Y]$ by

$$v(z) = \sum_{y \in Y} z[y]\, v(y).$$

The pushforward T^F. Defined on the basis by $T^F \delta_x = \delta_{F(x)}$, extended linearly:

$$T^F\Big(\sum_x u[x]\,\delta_x\Big) \;=\; \sum_x u[x]\,\delta_{F(x)} \;\in\; \mathbb{K}[Y].$$

A cryptanalytic property is a pair $u \in \mathbb{K}[X],\ v \in \mathbb{K}^Y$. Its evaluation is the scalar

$$v\big(T^F\,u\big) = \sum_{x\in X, y\in Y} u[x]\,T^F_{y,x} v(y) \;=\; \sum_{x\in X} u[x]\,v\big(F(x)\big).$$

All cryptanalytic properties are instances of $v(T^F u) = \sum_x u[x]\, v(F(x))$ – only the pair (u, v) changes.

Take $X = Y = \mathbb{F}_2^n$, $\mathbb{K} = \mathbb{R}$, and pick masks $a, b \in \mathbb{F}_2^n$. The **character basis** for $\mathbb{R}^X$ and its dual for $\mathbb{R}[X]$:

$$\chi^a(x) \;=\; (-1)^{a^\top x} \;\in\; \mathbb{R}^X, \qquad \chi_a[x] \;=\; \frac{(-1)^{a^\top x}}{2^n} \;\in\; \mathbb{R}[X].$$

Setting $u = \chi_a$ and $v = \chi^b$:

$$v(T^F u) \;=\; \frac{1}{2^n} \sum_x (-1)^{a^\top x + b^\top F(x)} \;=\; \mathrm{Cor}\big(a \xrightarrow{F} b\big).$$

Tensor decomposition: $\chi_a = \chi_{a_1} \otimes \cdots \otimes \chi_{a_n}$, $\chi^b = \chi^{b_1} \otimes \cdots \otimes \chi^{b_n}$.

Differential cryptanalysis (quasidifferential basis)

Lift to pairs: $F^{\times 2} : X^2 \to Y^2$, $F^{\times 2}(x, y) = (F(x), F(y))$. Work over $\mathbb{K} = \mathbb{R}$.

The **quasidifferential basis** for $\mathbb{R}^{X^2}$ and its dual for $\mathbb{R}[X^2]$ (Beyne–Rijmen '22):

$$q^{(u,d)}(x, y) \;=\; (-1)^{u^\top x}\,\delta_d(x+y), \quad q_{(u,d)}[(x, y)] \;=\; \frac{(-1)^{u^\top x}\,\delta_d(x+y)}{2^n}.$$

Pick $u = q_{(0,a)}$ and $v = q^{(0,b)}$ (plain differential):

$$v\big(T^{F^{\times 2}}u\big) \;=\; \frac{1}{2^n}\,\#\{x : F(x+a) + F(x) = b\} \;=\; \mathrm{DP}(a \to b).$$

General $u \neq 0$: $v(T^{F^{\times 2}}u)$ is a **quasidifferential** transition entry – captures sign-information missed by plain DP.

Integral cryptanalysis

Take $X = Y = \mathbb{F}_2^n$ and pick exponents $a, b \in \mathbb{F}_2^n$. Work over $\mathbb{K} = \mathbb{Q}$ (**ultrametric**) or $\mathbb{K} = \mathbb{F}_2$ (**ordinary**). Write $x^a := \prod_{i \in \mathrm{supp}(a)} x_i$.

The **monomial / parity basis** for $\mathbb{K}^X$ and its dual for $\mathbb{K}[X]$:

$$\mu^a(x) \;=\; x^a \;\in\; \mathbb{K}^X, \qquad \mu_a[x] \;=\; (-1)^{\mathrm{wt}(x+a)}\, a^x \;\in\; \mathbb{K}[X].$$

(Reduce $\mu_a[x] \equiv a^x \pmod 2$ for ordinary integral cryptanalysis; keep the $\mathbb{Q}$-valued form for ultrametric integral.)

Setting $u = \mu_a$ and $v = \mu^b$:

$$v(T^F u) \;=\; \sum_x \mu_a[x]\, \mu^b(F(x)) \;=\; \text{integral transition matrix } [b,\, a].$$

Specializing to S-functions

Now apply the geometric framework to an S-function $F : X_1 \times \cdots \times X_n \to Y_1 \times \cdots \times Y_n$.
Restrict to **tensor-product** properties $u = u_1 \otimes \cdots \otimes u_n$, $v = v_1 \otimes \cdots \otimes v_n$:

$$v(T^F u) \;=\; \sum_{x_1,\ldots,x_n} \underbrace{u_1[x_1] \cdots u_n[x_n]}_{\text{factors over } i} \cdot \underbrace{v_1(y_1) \cdots v_n(y_n)}_{\textbf{not} \text{ obviously factoring}} .$$

The dream. If each $v_i(y_i)$ depended only on x_i, we could push the sum inside:

$$v(T^F u) \;\stackrel{?}{=}\; \prod_{i=1}^{n} \Big(\sum_{x_i,\, y_i} u_i[x_i]\, v_i(y_i) \Big).$$

n independent $\mathcal{O}(1)$ sums – trivial.

Specializing to S-functions – the obstruction

The problem. y_i depends on x_i *and on the state* s_i, which is a function of $x_1, \ldots, x_{i-1}$. Past and future inputs are **coupled** along the chain.

The key observation. The coupling between $(x_1, \ldots, x_{i-1})$ and $(x_i, \ldots, x_n)$ is carried by the single state s_i. **Fix s_i, and the chain decouples**.

Decoupling via the state – the matrix product appears

Trick. Insert a sum over all internal state sequences $s_1, \dots, s_{n+1}$, restricted to be consistent with F:

$$v(T^F u) \;=\; \sum_{s_1,\dots,s_{n+1}} \sum_{x_1,\dots,x_n} \prod_{i=1}^{n} u_i[x_i]\, v_i(y_i)\, \mathbb{1}[F_i(s_i, x_i) = (s_{i+1}, y_i)]\,.$$

Once the states $s_1, \dots, s_{n+1}$ are fixed, the product factors over i:

$$v(T^F u) \;=\; \sum_{s_1,\dots,s_{n+1}} \prod_{i=1}^{n} \underbrace{\sum_{x_i,\, y_i} u_i[x_i]\, v_i(y_i)\, \mathbb{1}[F_i(s_i, x_i) = (s_{i+1}, y_i)]}_{=:\; M_i[s_{i+1},\, s_i]}\,.$$

The outer sum over states is exactly a **matrix product**:

$$v(T^F u) \;=\; \sum_{s_1, s_{n+1}} b[s_{n+1}]\, \big(M_n M_{n-1} \cdots M_1\big)[s_{n+1}, s_1]\, a[s_1] \;=\; b\, M_n \cdots M_1\, a.$$

Each M_i has **constant size** $|S_{i+1}| \times |S_i|$ – our universal formula.

Partial functions with parameters (1/2)

Each round $F_i : S_i \times X_i \to S_{i+1} \times Y_i$ takes a state s and input x_i, producing s' and output y_i.

Fix the input and the output, study only the state. For each pair (x_i, y_i) define the *partial* function

$$F_i(\,\cdot\,;\, x_i, y_i) \;:\; S_i \;\dashrightarrow\; S_{i+1}, \qquad s \mapsto s' \iff F_i(s, x_i) = (s', y_i).$$

Only those states s consistent with the prescribed output are mapped.

Its pushforward is a tiny transition matrix:

$$\left[T^{F_i(\,\cdot\,;x_i,y_i)}\right]_{s',\,s} \;=\; \mathbb{1}[F_i(s, x_i) = (s', y_i)] \;\in\; \{0, 1\},$$

of size $|S_{i+1}| \times |S_i|$ – **constant**, independent of n.

Partial functions with parameters (2/2)

Bundle the input/output by a property pair (u_i, v_i).

$$M_i \;:=\; T^{F_i}(\,\cdot\,; u_i, v_i) \;:=\; \sum_{x_i,\, y_i} u_i[x_i]\, v_i(y_i)\; T^{F_i(\,\cdot\,; x_i, y_i)}.$$

Entry by entry,

$$M_i[s', s] \;=\; \sum_{(x_i, y_i)\,:\, F_i(s, x_i) = (s', y_i)} u_i[x_i]\, v_i(y_i).$$

Still constant size; the property only **weights** rows and columns.

Lemma 3 – composition along the chain

For $u = u_1 \otimes \cdots \otimes u_n$, $v = v_1 \otimes \cdots \otimes v_n$,

$$T^F(\,\cdot\,; u, v) \;=\; T^{F_n}(\,\cdot\,; u_n, v_n) \cdots T^{F_2}(\,\cdot\,; u_2, v_2)\, T^{F_1}(\,\cdot\,; u_1, v_1).$$

Theorem 2

For every S-function $F = \text{S-FUNCTION}[F_1, \ldots, F_n]$ and every tensor-product property $(a \otimes u_1 \otimes \cdots \otimes u_n,\ b \otimes v_1 \otimes \cdots \otimes v_n)$,

$$(b \otimes v)\big(T^F(a \otimes u)\big) \;=\; b\, M_n\, M_{n-1} \cdots M_2\, M_1\, a.$$

where each $M_i = T^{F_i}(\cdot; u_i, v_i)$ is the constant-size transition matrix of the parameterised partial function, and a, b encode the initial and final state.

Consequences

- One $\mathcal{O}(n)$ algorithm for all properties of all S-functions.
- Recovers Wallén / Lipmaa–Moriai / Schulte-Geers / Mouha et al. as special cases.
- First algorithms for ultrametric integral, BCT, DL on χ and $\chi\chi$.

Algorithm in three lines

$$\text{property}(F) \;=\; \underbrace{b}_{\text{row}} \quad \underbrace{M_n M_{n-1} \cdots M_2 M_1}_{n \text{ small matrices, all constant size}} \quad \underbrace{a}_{\text{column}} .$$

1. For each round i, build $M_i \in \mathbb{K}^{|S_{i+1}| \times |S_i|}$ from the truth table of F_i and the basis vectors u_i, v_i.
2. Evaluate left-to-right (or right-to-left).

Compactness via change-of-basis

If $\mathscr{B}_i M_i \mathscr{B}_{i+1}^{-1}$ are simultaneously **column-monomial**, the product collapses to a scalar product $\Rightarrow$ compact SAT/MILP constraints of size $\mathcal{O}(\mathrm{poly}(n))$.

Plain S-function (e.g. modular addition). Initial carry fixed to 0 $\Rightarrow a = \delta_0$. Final state not observed $\Rightarrow b = \chi^0 = \sum_s \delta^s$.

Cyclic S-function (e.g. χ on $\mathbb{F}_2^n$). Final state **equals** initial state. Use $a = \delta_s$ and $b = \delta^s$, and sum over s:

$$v(T^F u) \;=\; \sum_{s \in S} \delta^s(M_n\, M_{n-1} \cdots M_1\, \delta_s) \;=\; \mathrm{Tr}(M_n M_{n-1} \cdots M_1)\,.$$

Why this matters

Combined with Theorem 2, this is the **first** general recipe for boomerang, DL, ultrametric integral, d-differential… of χ, $\chi\chi$ and other cyclic S-functions.

Modular addition – new results in one table

$$\mathrm{Cor} \;=\; b\, M_n \cdots M_1\, a, \qquad M_i \in \mathbb{Q}^{2\times 2}, \;\; a, b \in \mathbb{Q}^2.$$

Property	Basis	Compactness
Linear	characters $\chi_{(u,v)}$, χ^w	row-monomializable
Differential / QDTM	quasidifferential $q_{(u,d)}$, $q^{(v,e)}$	row-monomializable
Integral (mod 2)	monomial μ_u, μ^v	column-monomializable
Ultrametric integral	ultrametric μ_u, μ^v	**new**
DLCT, BCT, d-differential	various 2-wise	uniform $\mathcal{O}(n)$

Recovers Wallén / Lipmaa-Moriai / Beyne-Rijmen / Niu et al., and gives **new** ultrametric integral matrices. Same machinery for $x \boxplus c$ – only the M_i change with c_i.

Recall (Example 2):

$$\chi_n : \mathbb{F}_2^n \to \mathbb{F}_2^n, \quad y_i = x_i + (x_{i-1} + 1)\, x_{i-2}, \quad i \bmod n,$$

with state $s_i = (x_{i-1}, x_{i-2}) \in \mathbb{F}_2^2$. An S-function **with a cycle**: $s_{n+1} = s_1$. Each $M_i \in \mathbb{Q}^{4\times 4}$.

Correlation under masks (u, v) (trace formula from the cyclic case):

$$\mathrm{Cor} \;=\; \sum_{s\in\mathbb{F}_2^2} \delta^s\Big(T^{F_n}(\,\cdot\,; \chi_{u_n}, \chi^{v_n}) \cdots T^{F_1}(\,\cdot\,; \chi_{u_1}, \chi^{v_1})\, \delta_s \Big) \;=\; \mathrm{Tr}\big(M_n M_{n-1} \cdots M_1\big).$$

The state-update functions F_i are **all equal**, so only four matrices appear (one per $(u_i, v_i) \in \mathbb{F}_2^2$).

Each $M_i = T^{F_i}(\cdot; \chi_{u_i}, \chi^{v_i})$ has size 4×4 (state space $\mathbb{F}_2^2$). Only four matrices appear (one per $(u_i, v_i) \in \mathbb{F}_2^2$):

$$T^{F_i}(\cdot; \chi_0, \chi^0) = \tfrac{1}{2}\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \quad T^{F_i}(\cdot; \chi_1, \chi^0) = \tfrac{1}{2}\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ -1 & -1 & 0 & 0 \\ 0 & 0 & -1 & -1 \end{bmatrix},$$

$$T^{F_i}(\cdot; \chi_0, \chi^1) = \tfrac{1}{2}\begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & -1 \end{bmatrix}, \quad T^{F_i}(\cdot; \chi_1, \chi^1) = \tfrac{1}{2}\begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

These are **not** simultaneously row/column-monomializable – no compact SAT/MILP constraint – but the $\mathcal{O}(n)$ algorithm still works.

New algorithms via Theorem 2 (a partial list)

- **Ultrametric integral** transition matrix of χ_n – 16 matrices of size 16×16, exact.
- **Boomerang** connectivity table of χ_n (via quasi-3-differentials) – size $2^n \times 2^n$, but $\mathcal{O}(n)$ per entry.
- **Differential-linear** correlations on χ_n.
- Same machinery on the $\chi\chi$**-function** (Belkheyar et al., Eurocrypt 2025) and the **SHA-1 step function** (4-bit state).

All of these were either unknown or only known through experimentation.

Subterranean 2.0 in one slide

- NIST LWC second-round candidate (Daemen et al., 2020).
- State size: 257 bits, sole nonlinear layer is χ_{257}.
- Round function $\rho_i = \pi \circ \theta \circ \iota_i \circ \chi_{257}$, 8 rounds in total.
- χ_{257}^{-1} has a complicated algebraic normal form $\Rightarrow$ computing $\mathrm{BCT}(\chi_{257})$ was open.

Goal of this application

Use the universal formula on χ_{257} to evaluate the BCT entry needed to glue an upper and a lower differential – yielding the first boomerang distinguisher for the **full** permutation.

Theorem 3 (Kidmose–Tiessen)

For F a bijection on $\mathbb{F}_2^n$, with $A = \{(\Delta, a, \Delta + a) : a \in \mathbb{F}_2^n\}$, $B = \{(b, \nabla, b + \nabla) : b \in \mathbb{F}_2^n\}$,

$$\mathrm{BCT}_F[\Delta, \nabla] \;=\; 2^n \,\Pr\big[A \xrightarrow{F} B\big].$$

The right-hand side is a **truncated 3-differential** – a sum of d-differentials with $d = 3$.
Quasi-3-differential basis (Wang et al.):

$$q^{(u,d,e,f)}(x, y, z, w) = (-1)^{u^\top x}\, \delta_d(x+y)\, \delta_e(x+z)\, \delta_f(x+w).$$

Apply Theorem 2 on the quartet-extended χ_{257}: state is $(2+3 \cdot 2) = 8$ bits $\Rightarrow$ matrices of size 256×256.

The 8-round boomerang distinguisher

- Upper trail $b_0 \to b_4$: probability 2^{-58} (Daemen et al.).
- Lower trail $b_4 \to a_1$ (3 rounds): probability 2^{-46}.
- Glue with $\mathrm{BCT}_{\chi_{257}}[b_4, a_1] = 2^{-5.8}$ (computed via Theorem 2 + quasi-3-differentials).

$$\Pr[\text{boomerang}] \approx 2^{-(2\cdot 58+2\cdot 46+5.8)} = 2^{-213.8}.$$

First boomerang distinguisher on the **full** 8-**round** permutation.

Remark: Beyond the security claim of all sponge constructions on top – still a proof-of-concept that BCT of a large χ is now computable.

Background: the inverse of χ_n

Daemen showed χ_n is invertible for odd n, with an algorithmic inverse – but **not** its algebraic normal form (ANF).

Liu, Sarkar, Meier, Isobe (JoC 2022) gave the first closed-form expression:

$$x_i \;=\; y_i \;+\; \sum_{j=1}^{(n-1)/2} y_{i-2j} \prod_{k=1}^{j} \left(y_{i-2k+1} + 1\right).$$

Their proof is technical and ad hoc.

Our approach

Compute the integral (parity) properties of χ_n^{-1} using Theorem 2, then read off the ANF as a path-counting problem in a finite state machine. Generalises to any S-function.

Step 1 – transition matrices of χ_n^{-1}

Theorem 4. For an injective partial function, $T^{F^{-1}} = (T^F)^\top$. Apply with $F = F_i$, choose the monomial / parity basis, reduce mod 2 – four 4×4 matrices for χ_n^{-1}:

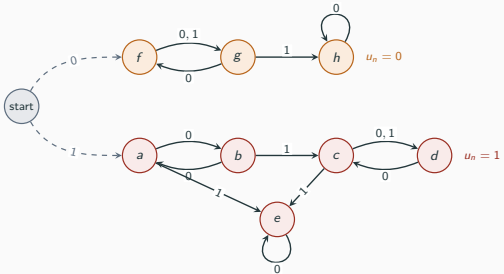
$$M_{0,0} = \begin{bmatrix} 1&0&0&0\\ 0&0&1&0\\ 0&1&0&0\\ 0&1&0&0 \end{bmatrix},\ M_{0,1} = \begin{bmatrix} 1&0&1&0\\ 1&0&1&0\\ 0&1&0&1\\ 0&1&0&1 \end{bmatrix},\ M_{1,0} = \begin{bmatrix} 0&0&0&0\\ 0&0&1&0\\ 0&0&0&0\\ 0&0&0&0 \end{bmatrix},\ M_{1,1} = \begin{bmatrix} 0&0&1&0\\ 0&0&1&0\\ 0&0&0&1\\ 0&0&0&1 \end{bmatrix}.$$

Here $M_{v_i,u_i} = T^{F_i^{-1}}(\cdot;\mu_{u_i},\mu^{v_i})$. The ANF coefficient of y^u in $(\chi_n^{-1})^v$ is

$$c_{v,u} \;=\; \mathrm{Tr}\big(M_{v_1,u_1}\,M_{v_2,u_2}\cdots M_{v_n,u_n}\big) \pmod 2.$$

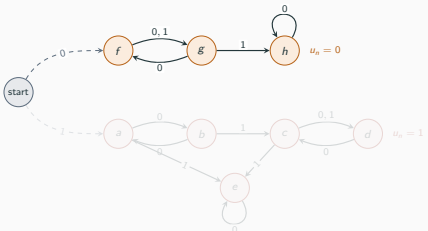
Step 2 – the finite state machine

Take $v = (0, \ldots, 0, 1)$. "Multiplying by M_{v_i, u_i}" becomes **walking** in a directed graph on **eight** reachable states a, b, c, d, e, f, g, h:



Edge label 0 means $u_i = 0$, label 1 means $u_i = 1$. A monomial y^u appears in x_n iff the path ends in $\{a, b, g, h\}$ – the trace-1 states.

Step 3a – valid paths when $u_n = 0$ (upper part)

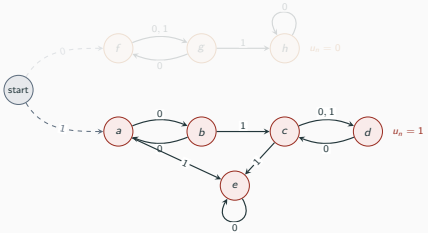


Observation. Once at h, stay (self-loop). Before h, the state alternates $f \leftrightarrow g$. Valid paths: j swaps $f \leftrightarrow g$, then jump to h and loop ($1 \le j \le \frac{n-1}{2}$).

Upper part contributes

$$\sum_{j=1}^{(n-1)/2} y_{n-2j} \prod_{k=1}^{j} \left(y_{n-2k+1} + 1\right).$$

Step 3b – valid paths when $u_n = 1$ (lower part)



Observation. Once at $\{c, d, e\}$, no return to $\{a, b\}$. But the path must end in a trace-1 state ($\{a, b\}$). Only valid path: $\text{start} \xrightarrow{1} a \xrightarrow{0} b \xrightarrow{0} a \xrightarrow{0} \cdots$, i.e. $u = (0, \ldots, 0, 1)$ – the monomial y_n.

Summing both cases and using shift-invariance of χ_n^{-1}:

$$x_i \;=\; y_i \;+\; \sum_{j=1}^{(n-1)/2} y_{i-2j} \prod_{k=1}^{j} \big(y_{i-2k+1} + 1\big).$$

Summary – one formula

One formula to rule them all

For any S-function F and any tensor-product property (u, v):

$$(b \otimes v)\big(T^F(a \otimes u)\big) \;=\; b\, M_n \cdots M_1\, a.$$

$\mathcal{O}(n)$ algorithm; matrices are constant-size; recovers all classical results.

Summary – applications

Applications

- Modular addition: ultrametric integral, plus a unified view of linear, differential, integral, DLCT, BCT.
- χ, $\chi\chi$, SHA-1 step: BCT, DL, ultrametric integral for the first time.
- Boomerang distinguisher on full Subterranean 2.0.
- New, geometric proof of the ANF of χ_n^{-1}.

Applications

- Modular addition: ultrametric integral, plus a unified view of linear, differential, integral, DLCT, BCT.
- χ, $\chi\chi$, SHA-1 step: BCT, DL, ultrametric integral for the first time.
- Boomerang distinguisher on full Subterranean 2.0.
- New, geometric proof of the ANF of χ_n^{-1}.

Thank you!

Questions?