

A Complete Classification of Whole-Output Linear Structures in FEILIAN-Type Components

Mounir IDRASSI
mounir@amcrypto.jp

27 September 2026

Abstract

We determine all input differences with constant whole-output XOR derivatives for a four-word ARX topology containing FEILIAN SubColumn. For every word width $w \geq 2$ and arbitrary XOR-linear internal maps, the complete linear-structure space contains a universal two-dimensional MSB family and has dimension at most five, with equality attained. We give a necessary-and-sufficient classifier involving at most 32 candidate masks and characterize all additional structures, including an exceptional coset possible only when the first map's image lies in the span of the least and most significant bits. For the specified 64-bit FEILIAN component, the complete space is exactly the universal family. With FEILIAN's actual ShiftRow, no nonzero member remains constant after both of the first two SubColumn layers. We also classify propagation at every boundary for all 20,160 invertible binary four-row mixers composed with ShiftRow. The five profiles stabilize by the third SubColumn layer, and exactly 4,032 mixers (20%) retain a nonzero family indefinitely. Algebraic proofs are supplemented by exhaustive small-word checks and a separate census audit. These results concern component structures and constancy at successive boundaries; they neither classify all endpoint structures of the iterated permutation nor establish an attack on the hash.

1 Scope and contribution

A probability-one differential identifies a symmetry of a component. A complete classification can determine which symmetries are forced by the operation topology, which depend on its internal maps, and which persist through subsequent layers. We study these questions for the four-word ARX topology underlying FEILIAN SubColumn, allowing arbitrary word widths and XOR-linear maps.

Our main result is a necessary-and-sufficient classification for this topology (Theorem 2). It includes singular maps and width two, and reduces the complete answer to at most 32 candidate masks after inspecting the first map's image. Corollary 3 gives the exact dimension and a sharp upper bound of five. Proposition 4 characterizes the exceptional coset through an image-containment condition and four map evaluations. The restricted-image branch forces singularity for $w \geq 3$, but can also occur for full-rank maps at width two.

Every allowed choice of maps admits the two-dimensional family $(x, y, x, x \oplus y)$ for $x, y \in \{0, 2^{w-1}\}$, with output difference $(x \oplus y, 0, y, 0)$. This separates a symmetry imposed by the topology from additional structures caused by particular maps. The FEILIAN specialization has exactly this family. Its actual ShiftRow eliminates nonzero constancy at both of the first two SubColumn boundaries (Corollary 5). Our second classification, Theorem 8, gives all five propagation profiles for the specified family of 20,160 four-row binary mixers composed with ShiftRow. The proof depends only on the component's complete structure space and difference map, so it applies to every parameter choice with the same pair. Together these

results settle the component symmetries and their boundary-compatible propagation within an explicit design family.

FEILIAN's specification gives probability-one SubColumn examples and a half-round example [7, Sections 4.5.2–4.5.3]. With $M = 2^{63}$ and the word order used below, both printed SubColumn examples are the same relation $(0, M, 0, M) \mapsto (M, 0, M, 0)$. Their input masks span only one dimension. The half-round example places that same mask in one column and applies ShiftRow; it supplies no second column generator. Corollary 5 lists the complete two-dimensional FEILIAN space, including the other two nonzero masks, and excludes every input mask outside it.

The proof treats the exact image of the first map and uses freely varying coordinates in the final addition, so neither internal map needs to be invertible. An explicit small example separates constancy at every component boundary from constancy only at the endpoint. Finite computations and comparisons with the identified scalar C check the implementations and examples.

Related work. Our whole-output condition is the binary vectorial condition on additive and linear translators studied by Lai [3, pp. 75–76], with zero adjoined to the translator set. Over $\mathbb{F}_2$, the additive and linear conditions coincide. The object is classical; our contribution is its complete characterization for the specific parameterized topology (1). Lipmaa and Moriai [5, Theorem 1, Algorithm 2, Lemma 3 in the ePrint version] characterize XOR differentials of two-input modular addition. The MSB-only probability-one fact used here is a specialization of that established theory. We give an elementary proof to keep the classification self-contained. Bao et al. [1, Section 3.1.5] distinguish Boolean linear structures from structures of vectorial functions defined using an output projection. We distinguish this projected terminology from the whole-output notion of Lai. Leander, Tezcan and Wiemer [4, Definitions 3–4, Lemmas 1–2, Section 2.4] relate derivatives, subspace trails and truncated differentials and give propagation procedures in a broader setting. Our calculation uses the special case where every selected input difference has one fixed output difference; it does not search all of their subspace trails. Dependency-aware analysis of consecutive modular additions is also established research [2]. Qin and Wu [6, Section 3.2], for example, explain how constraints on intermediate values can invalidate an independence assumption. Our universal coordinate changes are algebraic surjectivity arguments and do not assume independence of differential events.

2 Model and terminology

Fix a word width $w \geq 2$, and write

$$N = 2^w, \quad M = 2^{w-1}, \quad H = \{0, M\}, \quad W = M - 1.$$

Words are both elements of $\mathbb{Z}/N\mathbb{Z}$ for addition and vectors in $\mathbb{F}_2^w$ for XOR. All additions and subtractions of words are modulo N . Let $R_r = \text{ROR}_r$ and $R_s = \text{ROR}_s$ be right rotations, with counts reduced modulo w . Let $L_0, L_1 : \mathbb{F}_2^w \rightarrow \mathbb{F}_2^w$ be any linear maps. Define

$$\begin{aligned} p &= a + b, & u &= R_r(d \oplus p), \\ v &= c + d + L_0(u), & q &= R_s(b \oplus v), \\ F(a, b, c, d) &= (p + L_1(q), q, v, u). \end{aligned} \tag{1}$$

No rank assumption is needed for F to be a permutation. From its output (z, q, v, u) recover, in order,

$$\begin{aligned} p &= z - L_1(q), & b &= v \oplus R_s^{-1}(q), \\ d &= p \oplus R_r^{-1}(u), & a &= p - b, & c &= v - d - L_0(u). \end{aligned} \quad (2)$$

In particular, the pre-final coordinates (p, q, v, u) range freely.

For any vector-valued function G , set

$$D_\delta G(x) = G(x \oplus \delta) \oplus G(x), \quad \text{LD}(G) = \{\delta : D_\delta G \text{ is constant for all } x\}.$$

Write $\phi_G(\delta)$ for that constant, on $\text{LD}(G)$. The chain identity

$$D_{\delta \oplus \eta} G(x) = D_\delta G(x \oplus \eta) \oplus D_\eta G(x)$$

shows that $\text{LD}(G)$ is a vector space and ϕ_G is linear. If G is injective, ϕ_G is injective as well. By contrast, for a nonzero output mask λ , constancy of $\langle \lambda, D_\delta G(x) \rangle$ is only a projected structure. This paper's classification always concerns all output bits together. A projection can also retain several output bits. Projecting (1) onto its *complete fourth word* permits arbitrary masks on c, d and only requires the masks on a, b to lie in H . The dimension $2w + 2$ refers to this word-valued projection, rather than to an arbitrary scalar projection.

FEILIAN uses $w = 64$, $r = 8$, $s = 63$, and

$$L_0(z) = z \oplus \text{ROR}_5(z) \oplus \text{ROR}_{48}(z), \quad L_1(z) = z \oplus \text{ROR}_{11}(z) \oplus \text{ROR}_{40}(z). \quad (3)$$

Here FEILIAN means the Round 1 submission whose specification is dated 30 June 2026. These equations agree with its printed page 7 and the selected scalar C operations. The specification's word order is

$$(a, b, c, d) = (a_1, b_1, a_0, b_0), \quad (z, q, v, u) = (a'_1, b'_1, a'_0, b'_0).$$

In particular, $\text{ROR}_{63}(M) = 1$.

The surrounding AddConstant definitions differ: the specification's diagram puts the tweak in row 1 and constants in row 3, while the scalar C uses the opposite placement. In the fixed-message, fixed-tweak setting, both placements are common XOR translations and preserve constant differences at the selected boundaries. Our source correspondence concerns the component and row shift. Appendix A identifies the exact files and gives the detailed operation concordance, including this discrepancy in the full-hash descriptions.

3 Classification with arbitrary linear maps

The following lemma is the main step allowing a singular first map. Set $f_e(x) = x \oplus e$. The word 1 denotes the least significant bit. For the unrestricted summands, case (+) consists of additive translations $f_e(x) = x + e$ and case (−) of additive reflections $f_e(x) = e - x$ of the word group.

Lemma 1 (A restricted third summand). *Let $I \subseteq \mathbb{F}_2^w$ be a linear subspace and $K \in I$. For fixed masks C, D, V , the identity*

$$f_C(c) + f_D(d) + f_K(\ell) = f_V(c + d + \ell) \quad (c, d \in \mathbb{Z}/N\mathbb{Z}, \ell \in I) \quad (4)$$

holds if and only if one of the following disjoint cases holds:

$$\begin{aligned} (+) : & \quad C, D, K \in H, \quad V = C \oplus D \oplus K; \\ (-) : & \quad C, D \in W \oplus H, \quad K \in \{1, M + 1\}, \quad I \subseteq \text{span}_{\mathbb{F}_2}\{1, M\}, \quad V = C + D + K. \end{aligned}$$

Here $W \oplus H = \{M - 1, N - 1\}$, and V in the second case belongs to this same set.

Proof. Set $c = d = \ell = 0$ to get $V = C + D + K$, and define $g(x) = f_V(x) - V$ on the entire word group. With $d = \ell = 0$, the equation gives $f_C(c) - C = g(c)$; with $c = \ell = 0$, it gives $f_D(d) - D = g(d)$. Substitution back into the equation with $\ell = 0$ yields

$$f_C(x) - C = f_D(x) - D = f_V(x) - V = g(x), \quad g(c + d) = g(c) + g(d).$$

Every endomorphism of the cyclic additive word group has the form $g(x) = kx$. Evaluation at 1 gives $k = (1 \oplus V) - V \in \{1, -1\}$; this uses the domain of g and does not require $1 \in I$. If $k = 1$, each of C, D, V is in H : substitution of $x = e$ in $f_e(x) = x + e$ gives $2e = 0$. Conversely, XOR by an element of H is addition by that element. The zero identity then gives $K \in H$, proving case (+).

If $k = -1$, each of C, D, V belongs to $W \oplus H$. Indeed, substitution of $x = N - 1$ in $f_e(x) = e - x$ gives $2e = N - 2$, and both solutions do satisfy the identity for all x . Reducing $V = C + D + K$ modulo M forces $K \equiv 1 \pmod{M}$, hence $K \in \{1, M + 1\}$. The remaining condition on ℓ is

$$\ell \oplus K = K - \ell \iff 2(\ell \& \bar{K}) = 0 \pmod{N},$$

where $\bar{K} = (N - 1) \oplus K$ is the complement within w bits. Thus every bit of ℓ below the MSB must be supported on bit zero. This is exactly $I \subseteq \text{span}_{\mathbb{F}_2}\{1, M\}$. These conditions also make all terms in (4) agree, proving sufficiency. The two cases are disjoint because $w \geq 2$. $\square$

Taking $I = \{0\}$ and $K = 0$ gives the standard two-input addition fact: constant whole-output derivatives require both masks in H , with their XOR as output mask. Taking $I = \mathbb{F}_2^w$ recovers the analogous three-input fact for $w \geq 3$, since the negative case is then impossible. At $w = 2$ the negative case is possible; for example, masks $(1, 1, 1)$ on three summands give output mask 3. No width-two exception is suppressed in our theorem.

Theorem 2 (Complete classifier). *For the component (1), a mask (A, B, C, D) belongs to $\text{LD}(F)$ if and only if the following procedure accepts it. First require $A, B \in H$ and define*

$$P = A \oplus B, \quad U = R_r(D \oplus P), \quad K = L_0(U), \quad I = \text{im } L_0. \quad (5)$$

Require either case (+) or case (−) of Lemma 1, using these K, I and its prescribed value of V . Finally set

$$Q = R_s(B \oplus V), \quad J = L_1(Q), \quad \text{and require } J \in H. \quad (6)$$

The constant output difference is then

$$\phi_F(A, B, C, D) = (P \oplus J, Q, V, U). \quad (7)$$

Each branch has at most 16 candidate input masks, regardless of w .

Proof. Suppose the whole output derivative is constant. Its last word u forces the derivative of $p = a + b$ to be constant, since R_r is linear and invertible and the mask on d is fixed. The two-input fact gives $A, B \in H$ and the masks in (5).

The words c, d, u range independently: choose them arbitrarily, set $p = d \oplus R_r^{-1}(u)$, choose any b , and take $a = p - b$. Consequently $c, d, \ell = L_0(u)$ range over exactly $(\mathbb{Z}/N\mathbb{Z})^2 \times I$, even when L_0 is singular. The constant third output mask V is therefore equivalent to (4). Lemma 1 supplies precisely the two branches.

Now q has the fixed mask Q in (6). By (2), p, q vary freely. Restrict to $q = 0$ and vary p in the final addition. Its primed value is $(p \oplus P) + J = p + P + J$. Put $\Gamma = P + J$.

Evaluation at $p = 0$ fixes the constant difference as Γ , so $p + \Gamma = p \oplus \Gamma$ must hold for every p . Setting $p = \Gamma$ gives $2\Gamma = 0$, hence $\Gamma \in H$. Since $P \in H$, this forces $J \in H$, without any invertibility hypothesis on L_1 . Conversely, $J \in H$ makes XOR by J equal addition by J for every value of $L_1(q)$, giving the first mask $P \oplus J$. All the steps are reversible implications, so they prove both necessity and sufficiency. There are two choices for each of A, B, C, D in each branch. $\square$

Applying the classifier. Figure 1 takes the maps as their w basis images and returns every accepted input/output pair. The containment test inspects the basis images of L_0 before evaluating candidates. Each linear-map evaluation uses at most w XORs of w -bit words in this representation. Gaussian elimination on the accepted $4w$ -bit input masks selects a basis of $\text{LD}(F)$; retain the corresponding output masks to specify ϕ_F on that basis. Zero is returned as a structure but is omitted from the basis.

Classifier pseudocode. All word sums are modulo $N = 2^w$. Here $L[i]$ denotes evaluation from the supplied basis images.

```

M = N/2; H = {0,M}; W = M-1; accepted = empty dictionary
contained = every basis image of L0 belongs to {0,1,M,M+1}
for sign in {+, -}:
    if sign == - and not contained: continue
    choices = H if sign == + else {W,N-1}
    for (A,B,C,D) in H x H x choices x choices:
        P = A XOR B; U = R_r(D XOR P); K = L0[U]
        if sign == +:
            if K not in H: continue
            V = C XOR D XOR K
        else:
            if K not in {1,M+1}: continue
            V = C+D+K
        Q = R_s(B XOR V); J = L1[Q]
        if J in H: accepted[A,B,C,D] = (P XOR J,Q,V,U)
return accepted, a maximal independent set of its input masks
paired with their recorded output masks

```

Figure 1: At most 32 candidate masks, with map inspection performed separately. The returned input basis has at most five vectors by Corollary 3.

Corollary 3 (Dimension and exceptional coset). *For every choice of the rotations and linear maps,*

$$S := \{(x, y, x, x \oplus y) : x, y \in H\} \subseteq \text{LD}(F), \quad \phi_F(x, y, x, x \oplus y) = (x \oplus y, 0, y, 0). \quad (8)$$

Put

$$h_0 = L_0(R_r M), \quad h_1 = L_1(R_s M), \quad e_i = \begin{cases} 1 & h_i \in H, \\ 0 & h_i \notin H. \end{cases}$$

The positive branch S_+ is a subspace of dimension $2 + e_0 + e_1$. The negative branch is either empty or one coset of S_+ ; hence

$$\dim \text{LD}(F) = 2 + e_0 + e_1 + \mathbf{1}_{\{S_- \neq \emptyset\}} \leq 5.$$

For $w = 2$, the sharper lower bound $\dim \text{LD}(F) \geq 3$ holds. If $\text{im } L_0 \not\subseteq \text{span}_{\mathbb{F}_2}\{1, M\}$, in particular if $\text{rank } L_0 \geq 3$, the negative branch is empty. If also $h_0, h_1 \notin H$, then $\text{LD}(F) = S$.

Proof. Substitute $(A, B, C, D) = (x, y, x, x \oplus y)$ in the classifier. It gives $P = x \oplus y$, $U = K = 0$, $V = y$, and $Q = J = 0$, proving the universal inclusion and its difference map.

Write $A = aM, B = bM, C = cM, D = dM$ with bits a, b, c, d . Let $t = a \oplus b \oplus d$. The positive branch permits $t = 1$ exactly when $e_0 = 1$. Write $h_0 = k_0M$ when $e_0 = 1$, and set $k_0 = 0$ otherwise. The mask before R_s has coefficient $b \oplus c \oplus d \oplus k_0t$. It is constrained to vanish when $e_1 = 0$, and is unrestricted when $e_1 = 1$. These impose $2 - e_0 - e_1$ independent binary constraints: if both are imposed, they reduce to $d = a \oplus b$ and $c = a$; if only the second is imposed, its coefficient of c is one. This proves the dimension formula.

The whole set $\text{LD}(F)$ is a vector space. XOR of two negative-branch masks has every word in H and is therefore in S_+ , and XOR of a positive and a negative mask remains in the negative branch. A nonempty negative branch is consequently one coset of S_+ . The remaining claims follow from the classifier.

For the width-two bound, if $e_0 = 1$ or $e_1 = 1$, the positive branch already has dimension at least three. Otherwise both h_0, h_1 are odd. Image containment in $\text{span}_{\mathbb{F}_2}\{1, M\}$ is automatic, one of $g_0, g_0 \oplus h_0$ is odd, and one of $g_1, g_1 \oplus h_1$ is even, with g_0, g_1 as defined in Proposition 4. That proposition supplies the additional coset and hence the third dimension. $\square$

Proposition 4 (A direct exceptional-coset test). *In addition to h_0, h_1 above, put $g_0 = L_0(R_r W)$ and $g_1 = L_1(R_s W)$. The negative branch is nonempty if and only if all three conditions hold:*

$$\begin{aligned} \text{im } L_0 &\subseteq \text{span}_{\mathbb{F}_2}\{1, M\}, \\ \{g_0, g_0 \oplus h_0\} \cap \{1, M+1\} &\neq \emptyset, \\ \{g_1, g_1 \oplus h_1\} \cap H &\neq \emptyset. \end{aligned} \tag{9}$$

Together, the image-containment check and the four evaluations h_0, h_1, g_0, g_1 decide the indicator in the dimension formula.

Proof. Parameterize a negative-branch candidate by bits a, b, c, d as

$$A = aM, \quad B = bM, \quad C = W \oplus cM, \quad D = W \oplus dM.$$

Then $K = g_0 \oplus (a \oplus b \oplus d)h_0$. The first two conditions in (9) are exactly the image restriction and the existence of a choice with $K = 1 \oplus kM$, for a bit k . For any such choice, modular addition gives

$$V = C + D + K = W \oplus (1 \oplus c \oplus d \oplus k)M, \quad J = g_1 \oplus (b \oplus 1 \oplus c \oplus d \oplus k)h_1.$$

The free bit c selects either value of J independently of the earlier choice of K . The last condition is therefore necessary and sufficient for $J \in H$. This proves both directions, including $w = 2$. $\square$

Map inspection and candidate count. The bound of 32 concerns candidate masks. Checking image containment requires information about the whole map, such as its images on a basis of $\mathbb{F}_2^w$; it is not a constant-cost black-box test. Rank at most two is necessary for the exceptional branch, but the orientation of the image also matters. For example, take $w = 4$, $r = s = 0$, $L_1 = 0$, and first maps with basis images

$$(L_0(1), L_0(2), L_0(4), L_0(8)) = (1, 8, 8, 0), \quad (\tilde{L}_0(1), \tilde{L}_0(2), \tilde{L}_0(4), \tilde{L}_0(8)) = (1, 2, 2, 0).$$

Both have rank two and give $(h_0, h_1, g_0, g_1) = (0, 0, 1, 0)$. Their images are $\text{span}_{\mathbb{F}_2}\{1, 8\}$ and $\text{span}_{\mathbb{F}_2}\{1, 2\}$. Only the first satisfies containment in $\text{span}_{\mathbb{F}_2}\{1, 8\}$. Their complete derivative spaces have dimensions five and four, respectively. Thus the four evaluations alone do not decide existence.

Sharpness and hypotheses. For any $w \geq 2$, take $r = s = 0$, $L_0(z) = z \& 1$, and $L_1 = 0$. Then $e_0 = e_1 = 1$ and $(0, 0, W, W)$ is a negative-branch mask with output $(0, N-1, N-1, W)$. Thus dimension five is attained. As a worked instance, at $w = 3$ the two inputs $(1, 2, 3, 4)$ and $(1, 2, 0, 7)$ give $(p, u, L_0 u, v, q) = (3, 7, 1, 0, 2)$ and $(3, 4, 0, 7, 5)$. Their outputs are $(3, 2, 0, 7)$ and $(3, 5, 7, 4)$. The difference is $(0, 7, 7, 3)$ as predicted.

Appendix B gives further controls illustrating which rank assumptions can be weakened.

4 FEILIAN and composition through linear mixing

Let $\mathcal{A}$ denote a *SubColumn layer*: four independent copies of the component acting on a matrix in row-major order. Let R rotate row i left by i positions. The map $R \circ \mathcal{A}$ is a half-round core; a round has two such SubColumn layers.

Corollary 5 (Complete FEILIAN specialization). *For $w = 64$, $r = 8$, $s = 63$ and the maps (3), $\text{LD}(F) = S$ with dimension two. For $M = 2^{63}$ its three nonzero input/output differences are exactly*

Input difference	Constant output difference
$(M, 0, M, M)$	$(M, 0, 0, 0)$
$(0, M, 0, M)$	$(M, 0, M, 0)$
$(M, M, M, 0)$	$(0, 0, M, 0)$

Furthermore, $\text{LD}(\mathcal{A}) = S^4$ has dimension eight and

$$\text{LD}(\mathcal{A} \circ R \circ \mathcal{A}) \cap \text{LD}(\mathcal{A}) = \{0\}. \quad (10)$$

Proof. Both maps in (3) have rank 64. Represent right rotation by t in $\mathbb{F}_2[t]/(t^{64} + 1)$: the modulus is $(t + 1)^{64}$, and both defining polynomials evaluate to one at $t = 1$. Moreover,

$$h_0 = 2^{55} \oplus 2^{50} \oplus 2^7 \notin H, \quad h_1 = 1 \oplus 2^{53} \oplus 2^{24} \notin H.$$

Corollary 3 gives $\text{LD}(F) = S$ and its difference map. Independent column inputs give $\text{LD}(\mathcal{A}) = S^4$. Its image $T = \phi_{\mathcal{A}}(S^4)$ consists of MSB-only differences supported in rows 0 and 2. Hence $R(T) = T$ and $T \cap S^4 = \{0\}$. Proposition 6 below and injectivity of $\phi_{\mathcal{A}}$ give (10). The same argument applies to any permutation of columns separately within each row. $\square$

The middle row is the example printed twice in the specification. The first and middle input masks form a basis of the complete space. The intersection in (10) restricts the conclusion to constancy at both SubColumn boundaries.

Proposition 6 (Known structures through a second layer). *Let F be bijective, G any function, and L invertible and linear. Then*

$$\text{LD}(G \circ L \circ F) \cap \text{LD}(F) = \{\delta \in \text{LD}(F) : L\phi_F(\delta) \in \text{LD}(G)\}. \quad (11)$$

Proof. For $\delta \in \text{LD}(F)$, put $\epsilon = L\phi_F(\delta)$. Then $D_{\delta}(G \circ L \circ F)(x) = D_{\epsilon}G(LF(x))$. The argument $LF(x)$ ranges over the entire input space of G , proving the equivalence. $\square$

Surjectivity in this argument is essential to the reverse implication. Appending an invertible linear map or an XOR by a fixed constant does not affect constancy. The following example explains why the intersection cannot be removed.

An endpoint counterexample. Consider the width-two variant with $r = 0, s = 1, L_0 = 0$, and $L_1(z) = z \& 1$. For $\delta = (2, 2, 0, 0)$, write the first component's second output as $q = 2h + t$, where $h, t \in \{0, 1\}$. Use primes for the translated input. The intermediate words p, u, v are unchanged, and

$$q' = 2h + 1 - t, \quad z = p + t, \quad z' = p + 1 - t,$$

with additions modulo four. The first-word derivative is therefore $p \oplus (p + 1)$, which takes both values one and three, so $\delta \notin \text{LD}(F)$.

The second component takes input (z, q, v, u) . Its intermediate sum satisfies

$$p_2 = z + q = p + 2h + 2t, \quad p'_2 = z' + q' = p_2 + 2.$$

Its third output $v_2 = v + u$ is unchanged. Its fourth output $u_2 = u \oplus p_2$ changes by the MSB mask two. Since $q' \oplus q = 1$, its second output $q_2 = R_1(q \oplus v_2)$ also changes by two. This leaves the low bit selected by L_1 unchanged, so the first output $p_2 + L_1(q_2)$ changes by two as well. Hence, for every base input,

$$D_\delta(F \circ F) = (2, 2, 0, 2).$$

Complete truth-table classification, recorded in Section 5, gives the full spaces

$$\begin{aligned} \text{LD}(F^2) &= \text{span}_{\mathbb{F}_2} \{(2, 2, 0, 0), (2, 0, 2, 0), (0, 0, 0, 2)\}, \\ \text{LD}(F^2) \cap \text{LD}(F) &= \text{span}_{\mathbb{F}_2} \{(0, 2, 2, 0), (2, 2, 0, 2)\}. \end{aligned}$$

Their dimensions are three and two, respectively. The displayed counterexample lies in the first space and outside the second.

4.1 An exact rank procedure

Proposition 7 (Constancy at every component boundary). *Let $F : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ be a permutation, $L : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ invertible and linear, and $S = \text{LD}(F)$ of dimension d . Let $E : \mathbb{F}_2^d \rightarrow \mathbb{F}_2^n$ be a basis embedding of S . Define the subspaces $K_k \subseteq \mathbb{F}_2^d$ by*

$$EK_k = \bigcap_{j=1}^k \text{LD}((L \circ F)^j) \quad (k \geq 1). \quad (12)$$

The space EK_k consists precisely of the input masks constant after each of the first k component layers; an invertible final L preserves constancy. Choose linear maps

$$C_S : \mathbb{F}_2^d \rightarrow \mathbb{F}_2^n, \quad Q_S : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^{n-d}, \quad J_S : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^d,$$

where $C_S = \phi_F E$, $\ker Q_S = S$, and $J_S E = I_d$. Set $D_S = Q_S L C_S : \mathbb{F}_2^d \rightarrow \mathbb{F}_2^{n-d}$ and $T_S = J_S L C_S : \mathbb{F}_2^d \rightarrow \mathbb{F}_2^d$. Then

$$K_1 = \mathbb{F}_2^d, \quad K_k = \bigcap_{j=0}^{k-2} \ker(D_S T_S^j) \quad (k \geq 2) \quad (13)$$

In particular, for $k \geq 2$,

$$\dim K_k = d - \text{rank} \begin{pmatrix} D_S \\ D_S T_S \\ \vdots \\ D_S T_S^{k-2} \end{pmatrix}, \quad (14)$$

where the stacked matrix has $(k-1)(n-d)$ rows and d columns. The spaces decrease and stabilize at the first equality, with at most d strict dimension drops. Their values do not depend on the extension of J_S outside S .

Proof. Since L is invertible, $\text{LD}(L \circ F) = S$, so (12) defines K_k uniquely. The first boundary requires a mask in S . For its parameter v , the next input difference is $LC_S v$. Membership in S is equivalent to $D_S v = 0$, and then $LC_S v = ET_S v$. Every prefix is bijective, so Proposition 6 applies at each step; induction gives (13), and rank-nullity gives (14). Only vectors in S are decoded by J_S along an admissible trajectory, proving independence of its extension. Equivalently $K_{k+1} = K_2 \cap T_S^{-1}(K_k)$. This recurrence preserves the first equality. A strict inclusion reduces dimension by at least one. $\square$

In particular, the two-layer space is $E \ker(Q_S LC_S)$, computable by Gaussian elimination. This recurrence concerns constancy at *every* boundary, a stronger condition than constancy only at the endpoint of $(L \circ F)^k$.

4.2 A complete census of row mixing variants

Take any parameter choice with $\text{LD}(F) = S$ and the difference map in (8); the actual 64-bit component is one such choice. Keep the same four-column layout and row shift R . For each $B \in \text{GL}(4, 2)$ let B act by XOR on the four row words independently in every column, and replace the interlayer map by $L_B = B \circ R$. This is a precisely specified class of experimental variants, with 20,160 members. It includes $B = I$, which recovers actual ShiftRow when the component is the specified FEILIAN instance. The state has $16w$ bits; the census concerns this four-row mixing family.

Encode an MSB-only matrix by 16 bits $z_{r,c}$. A parity check for S^4 is sufficient in this subspace: C_S produces only MSB differences, and L_B preserves bit positions within words, so every relevant trajectory stays in that 16-dimensional subspace. The eight parity-check equations are, for each c ,

$$z_{0,c} \oplus z_{2,c} = 0, \quad z_{3,c} \oplus z_{0,c} \oplus z_{1,c} = 0.$$

For parameters $(x_c, y_c) \in \mathbb{F}_2^2$ the maps E, C_S have columns $(x_c, y_c, x_c, x_c \oplus y_c)$ and $(x_c \oplus y_c, 0, y_c, 0)$. Thus (13) is an eight-variable calculation, independent of base states. The following proof explains Table 1; enumeration provides a separate check.

$\dim K_1$	$\dim K_2$	$\dim K_3$	$\dim K_\infty$	Number of B
8	0	0	0	9,216
8	4	0	0	6,912
8	4	2	2	1,152
8	4	4	4	2,304
8	8	8	8	576

Table 1: Exact census for $L_B = B \circ R$. K_k requires a constant whole-output derivative at each of k SubColumn-layer boundaries. Every profile has stabilized by K_3 , as proved in Theorem 8.

Theorem 8 (Algebraic row-mixing census). *For $L_B = B \circ R$ with $B \in \text{GL}(4, 2)$, the five profiles and counts in Table 1 are complete whenever $\text{LD}(F) = S$ with difference map (8). Every K_3 is stable.*

Proof. Let $e_0, \dots, e_3$ be the coordinate vectors of $\mathbb{F}_2^4$. Put $E_0(x, y) = (x, y, x, x \oplus y)$, $S_0 = \text{im } E_0$, and

$$T_0 = \text{span}_{\mathbb{F}_2}\{e_0, e_2\}.$$

Since $R(T) = T$, $\dim K_2 = 4 \dim(BT_0 \cap S_0)$. There are 35 two-dimensional subspaces of $\mathbb{F}_2^4$: 16 meet S_0 trivially, 18 in a line, and one equals S_0 . For the first count, graphs of the 16 linear maps from a fixed complement to S_0 give all disjoint planes. For the second, each of the three lines in S_0 lies in six planes other than S_0 . Each plane is the image of T_0 under $20,160/35 = 576$ invertible maps. This proves the totals 9,216, 10,368, and 576 at dimensions zero, four, and eight. The zero case stays zero. If $BT_0 = S_0$, every family mask returns to the family, so all eight dimensions persist.

For the remaining line-intersection case, unique nonzero pairs (α, β) and (γ, η) in $\mathbb{F}_2^2$ satisfy

$$B(\alpha e_0 \oplus \beta e_2) = E_0(\gamma, \eta).$$

Write $x, y \in \mathbb{F}_2^4$ for row vectors of family parameters, and let σ be their cyclic shift by two positions. Then $\sigma^2 = I$ and $\dim \ker(I \oplus \sigma) = 2$. Surviving the first transition is equivalent to

$$x \oplus y = \alpha z, \quad \sigma y = \beta z \quad (z \in \mathbb{F}_2^4),$$

and the next parameters are $x' = \gamma z$, $y' = \eta z$. For nonzero (α, β) , a pair (u, v) has the form $(\alpha t, \beta t)$ exactly when $\beta u \oplus \alpha v = 0$. Hence the next transition survives exactly when

$$[\beta(\gamma \oplus \eta)I \oplus \alpha \eta \sigma]z = 0. \tag{15}$$

The nine possibilities have the following kernel dimensions:

$(\alpha, \beta) \setminus (\gamma, \eta)$	(1, 0)	(0, 1)	(1, 1)
(1, 0)	4	0	0
(0, 1)	0	0	4
(1, 1)	0	2	0

Write ab for the bit pair (a, b) . The two dimension-four cases stay in the family: for labels (10, 10) the parameters are $(z, 0)$ and are fixed; for (01, 11) they are $(\sigma z, \sigma z)$ and map to (z, z) , exchanging the two forms. The unique dimension-two case has labels (11, 01) and $z = \sigma z$; its parameters $(0, z)$ are fixed. This proves stability at K_3 in every case.

Each fixed pair of nonzero labels occurs for $12 \cdot 12 \cdot 8 = 1152$ matrices. Fix a domain basis beginning with the distinguished vector in T_0 and a complementary vector of T_0 . The first image is prescribed; the second must be outside S_0 , giving 12 choices. The final two images must lie outside the spans of the preceding images, giving 12 and 8 choices. Conversely these choices give exactly the prescribed intersection and labels. Six label pairs have kernel dimension zero, one has dimension two, and two have dimension four. Their counts are therefore 6912, 1152, and 2304, completing the table. $\square$

The proof uses only $(S, \phi_F|_S)$ and the row shift. In particular, it also covers the width-three modular-rotation reduction and the width-four example with $r = 0$, $s = 1$, $L_1 = \text{id}$ and basis images $(L_0(1), L_0(2), L_0(4), L_0(8)) = (1, 2, 0, 1)$. Both have exactly this pair. Of the 20,160 mixers, $1152 + 2304 + 576 = 4032$, exactly 20%, retain a nonzero family indefinitely. The other 80% eliminate it by the third SubColumn layer. These percentages describe this structural property and are not a security ranking of the mixers.

A partially surviving family. Take $B(a, b, c, d) = (b, a, a \oplus c, a \oplus d)$, an invertible map with binary row masks (2, 1, 5, 9). Membership after one transition requires $x = y \oplus \sigma y$, leaving four dimensions. The next parameters are $(0, \sigma y)$. A second transition requires $y = \sigma y$, leaving the two-dimensional stable family with $x = 0$. This explains one representative of the $8 \rightarrow 4 \rightarrow 2$ profile without reference to a search log.

Appendix C gives a positive control preserving all eight dimensions and its feedforward consequence.

5 Exact finite experiments and model checks

The artifact contains an equation-based Python implementation, a separately written small-word C evaluator, and a C wrapper around the hash-identified original scalar FEILIAN1024 implementation. Table 2 distinguishes the roles of the proofs and computations.

Evidence	What it establishes	Main record
Proof	Classification for all allowed maps and widths; boundary propagation; algebraic census	Lemma 1, Theorems 2 and 8, Propositions 4 and 7
Exact finite checks	Every accepted mask checked on all bases; each rejected mask has a witness; finite census implementation	Truth tables, exclusion witnesses, and direct parameter-orbit audit
Reference C	Agreement of the equations with selected operations of the identified implementation	Source manifest, concrete inputs, and recorded C/Python comparisons

Table 2: Complementary evidence. The finite checks do not prove the universal theorem; program agreement does not by itself establish specification fidelity.

Small-word classifications. For each parameter setting we generate every input/output pair. For every mask δ we compare $F(x \oplus \delta) \oplus F(x)$ with its value at zero. A member is accepted only after all base inputs are exhausted; a nonmember has a saved base x producing a different derivative. Table 3 breaks down the 2,123 parameter-setting records. Repeated settings across groups remain repeated checks. The eight four-bit controls consist of four identity-map alignment choices and one each for reflection, zero maps, a rank-two first map, and a rank-one second map.

Width	Parameter group	Records
2	All maps and rotation counts	1,024
3	All 512 first maps; $L_1 = 0$ or id; $r = s = 0$	1,024
3	Seeded choices of both maps and rotations	64
2, 3, 4	Direct modular-rotation reductions	3
4	Alignment, reflection, zero-map and rank controls	8
Total, retaining repeated settings		2,123

Table 3: Finite classification coverage. The complete three-bit first-map slice has $512 \times 2 = 1,024$ settings.

In total, 5,312,768 truth-table entries agree between C and Python, and 75,956,572 mask/base derivative comparisons classify all 5,312,768 candidate masks across the listed settings. Every classifier prediction agrees. Every rejected mask has a stored certificate checked against the Python truth table. The dimension counts across these settings are 32 of dimension two, 1,284 of dimension three, 726 of dimension four, and 81 of dimension five. This finite sweep covers its stated parameter sets only. Separately, the two/three-summand addition lemma is enumerated at widths 2, 3 and 4, including the width-two exception.

A separate Python audit rechecks every finite membership and exclusion witness. Its parameter-orbit calculation follows all 256 trajectories for each mixing matrix until departure or repetition. It is independent of the rank recurrence but shares the component, difference-map and encoding routines with the main computation. Thus it checks the census by a different algorithm, while shared model or ordering errors could remain undetected. The

algebraic nine-label calculation agrees with every enumerated matrix. The code supplement also checks the exceptional-coset formula and worked examples (Appendix A).

The direct reduction of the FEILIAN equations takes every rotation count modulo w and uses exactly the three XOR terms in each map, allowing cancellation when rotations coincide. Table 4 records the resulting ranks; in particular, the second map is singular at width three. These are experimental reductions and have no asserted security relation to the 64-bit construction.

w	r	s	rank L_0	rank L_1	dim LD(F)
2	0	1	2	2	3
3	2	0	3	1	2
4	0	3	4	4	2

Table 4: Exact ranks and complete derivative dimensions of the modular-rotation reductions. The theorem allows the singular width-three map.

Complete truth tables of the width-two endpoint example classify all 256 input masks and confirm the endpoint and both-boundary bases displayed after Proposition 6.

Actual 64-bit checks. The checks include the predicted column and matrix differences, the inverse, and two distinct derivatives for each of the 255 nonzero members of S^4 through $\mathcal{A} \circ R \circ \mathcal{A}$. The scalar C agrees on 261 ordinary columns, 69 half-round states, 69 second-layer states, and 129 evaluations of the two Sigma maps and the rotations. It also agrees with both recorded derivative values in each of the 255 negative certificates.

Resources and reproduction. The exact classifier of a truth table on $4w$ bits uses $O(2^{4w})$ memory and at most 2^{8w} comparisons. Widths never exceed four in this search, so each C table has at most 65,536 32-bit entries. The reproduction code fixes the parameter sets and random seeds, and generates the result records, binary exclusion witnesses and numerical tables. Appendix A describes the code supplement.

6 Relation to the hash and remaining limits

All classification quantifiers range over arbitrary internal component inputs. Neither coordinate surjectivity nor a chosen chaining value proves reachability from the fixed hash IV. For a fixed common message and tweak, XOR injections and round constants are translations. They preserve constant differences through each fixed component boundary. Actual changed-message computations compare different sequences of translations, and reuse the same message halves at later scheduled injection points. They are not modeled by an independently chosen state correction at one boundary.

The actual message injection changes only rows 0 and 2. Its full word-valued difference space meets S^4 trivially: zero rows 1 and 3 in $(x, y, x, x \oplus y)$ force $y = x = 0$. Consequently the first injection from equal states cannot directly create a nonzero member of this family. This does not exclude other message-dependent relations later in the computation. The reference source includes version, final-block flag and counter injection; we make no reduced-round or compression-level experimental claim requiring a new model of that schedule.

There is a limited general feedforward statement. For a fixed-message, fixed-tweak permutation E , define $C(h) = E(h) \oplus h$ on arbitrary chaining states. Then

$$D_\delta C(h) = D_\delta E(h) \oplus \delta, \quad \text{LD}(C) = \text{LD}(E).$$

This identity supplies no classification of $\text{LD}(E)$. A final linear output projection can create additional projected structures, so completeness for the untruncated component cannot be transferred to a truncated digest. Appendix C gives a worked feedforward consequence for a deliberately altered mixer. For actual FEILIAN, the full-round endpoint space outside S^4 , projected structures and quantitative differential probabilities remain open here. The component results establish no hash attack or bound on the hash’s security margin.

The classification identifies every whole-output linear structure of topology (1), including its restricted-image exceptions, for arbitrary XOR-linear maps and widths. It separates the universal MSB symmetry from additional map-dependent structures. For components whose complete space is that universal family, the proved row-mixing census then gives every boundary-compatible propagation profile and its stabilization. Together these results provide exact criteria for the existence and persistence of these component symmetries.

7 AI assistance

I used OpenAI Codex, GPT-6 Astra and Moonshot kimi-k3 as aids for selected parts of the analysis, code development and manuscript preparation. These tools supported my own research and development. I am responsible for the paper’s arguments, results and final text.

A Computational reproducibility

A small code supplement, version 2026-09-27-r08, contains the Python and C implementations, deterministic parameter generators, expected numerical summaries and a short README. Its original code is provided under the MIT License, included as LICENSE. The supplement is available as release v1.0.0:

<https://github.com/amcrypto-jp/feilian-linear-structures/releases/tag/v1.0.0>

The release provides `feilian-structures-artifact-20260927-rev08.zip`, accompanied by a SHA-256 checksum. Its exact SHA-256 identifier is

5afc9572a724a3c8cd46555bb96f7baa2ea06bd79ddfae5311f9a364aabc593

The computational code is also recorded at repository commit

fea65479a4a6e1cdea28ef096bfeb4705021e5b2.

It reproduces the finite classifications, mixing census, endpoint example and modified-mixer controls, and generates the numerical tables in CSV and LaTeX form. Python 3.10 or later and a C99 compiler suffice; no TeX installation is needed to run the computations. From the supplement directory, run

```
python3 run.py --full --out results
```

The command computes its own result files and checks them against the expected summaries. Without `--submission-root`, only the non-reference summaries are compared, and the completion message explicitly reports that the original-C comparisons were not executed. Omitting `--full` runs a small preliminary subset. Seeds, encodings, output files and optional reference-C comparisons are described in the README. All mathematical statements and proofs are in the paper.

Analyzed implementation. The specification is dated 30 June 2026 and the scalar source comes from the same official Round 1 archive [7]. The exact files, relative to its extracted FEILIAN/ directory, and their SHA-256 identifiers are:

- Algorithm specifications/Specification.pdf
8e3a9a109f3188ab57cbd0156c243f80a6453051cf98006dfe11d66bdeb5fa8
- Implementations/Reference_Implementation/FEILIAN1024/CryptHash_AlgorithmInstance.c
421ee5370e0c5ed33618f371b4ea5a3b14a55f7bc71f0237bf40274e4229f02e

The code’s `sources.json` records the archive locator and required source hashes. To repeat the reference comparisons, obtain the submission separately and extend the command as follows:

```
python3 run.py --full --out results-reference \
  --submission-root /path/to/FEILIAN
```

The C file and its header are hash-checked before compilation. The adapter includes the original C unchanged and adds exported comparison entry points and array copies. These comparisons are optional; the mathematical experiments and numerical tables run without the original submission.

A release check on 27 September 2026 used a fresh archive extraction, an isolated Python virtual environment and a new output directory on Ubuntu 26.04.1 LTS (x86_64), with Python 3.14.4 and GCC 15.2.0. The complete reference-enabled command above finished in approximately 38 seconds. All expected summaries, including the original-C comparisons, matched. The C evaluators were compiled afresh for this run.

Source-to-model concordance. Using the word order in Section 2, the following table matches the sequential operations on printed page 7 of the pinned specification to (1). C line numbers refer to the identified `CryptHash_AlgorithmInstance.c`.

Model value	Specification assignment	Scalar C location
$p = a + b$	Updated a_1 , also named s_0	<code>sbox_column</code> , line 180
$u = R_8(d \oplus p)$	Updated b'_0 , using that s_0	Line 182
$v = c + d + L_0(u)$	Updated a'_0 , also named s_1	Lines 181, 183
$q = R_{63}(b \oplus v)$	Updated b'_1 , using that s_1	Line 184
$z = p + L_1(q)$	Final a'_1	Line 185

The Sigma definitions on printed page 6 give (3) and match `Sigma0_sbox` and `Sigma1_sbox` (C lines 46–54). The C `rotr64` function (lines 36–40) fixes the right-rotation convention. In a state matrix, (a, b, c, d) occupies rows 0–3 of one column; `sbox_matrix` (lines 188–192) uses exactly this order. The adapter flattens entry (i, j) at index $4i + j$. `ShiftRow` on printed page 8 and `shift_rows` (lines 198–211) both implement $R(X)_{i,j} = X_{i,(j+i) \bmod 4}$.

For the surrounding translations, printed pages 9–11 describe `AddConstant`, message injection into rows 0 and 2, and the two `SubColumn/ShiftRow` pairs per round; printed page 12 gives the phase rotation of the constants. The corresponding C locations are lines 143–161, 219–254 and 285–293. There is a discrepancy outside the component equations: the printed `AddConstant` diagram puts the tweak in row 1 and constants in row 3, whereas the scalar C puts constants in row 1 and the tweak in row 3. Both are common XOR translations in the fixed-message, fixed-tweak setting used here. Our correspondence claim concerns the component and row shift; it does not assert identical full-hash definitions across those descriptions.

B Additional parameter controls

Invertibility of L_1 is unnecessary even for $\text{LD}(F) = S$. For example, with the 64-bit FEILIAN L_0, r, s but $L_1(z) = z \& 1$, we still have $h_1 = 1 \notin H$ and obtain the same S . Even invertibility of L_0 can be weakened: at $w = 4, r = 0, s = 1$, the map with basis images $(1, 2, 0, 1)$ has rank two, image outside $\text{span}_{\mathbb{F}_2}\{1, 8\}$, and $h_0 = 1$; L_1 equal to the identity gives $h_1 = 4$. Again the complete space is S . Conversely, at $w = 2$ with both maps the identity and $r = 0, s = 1$, the positive branch has dimension three and the negative branch adds one. The mask $(2, 0, 1, 1)$ has derivative $(0, 2, 1, 3)$. These are mathematical controls for the classification.

C A deliberately modified feedforward control

This worked control illustrates the consequence of preserving component structures through an altered mixer. It uses freely chosen chaining states.

Preserved family and parameter map. Take

$$B(a, b, c, d) = (a \oplus c, c, a \oplus b \oplus c, a \oplus d). \quad (16)$$

It is invertible: recover c from the second output, then a, b , and d . It maps T_0 onto S_0 , so all eight dimensions survive every boundary. Writing the family parameters as four-bit row vectors x, y and denoting the two-position cyclic shift by σ , one complete component/mixing step follows the equations

$$(x, y, x, x \oplus y) \xrightarrow{A} (x \oplus y, 0, y, 0) \xrightarrow{R} (x \oplus y, 0, \sigma y, 0) \xrightarrow{B} (x', y', x', x' \oplus y'),$$

where $x' = x \oplus y \oplus \sigma y$ and $y' = \sigma y$. Applying this parameter map twice returns (x, y) . Direct evaluations check this prediction for all 256 parameter choices through four layers.

Corollary 9 (Feedforward in the deliberately modified control). *Use the positive-control B in (16), the specified 64-bit component layer A , and $L_B = B \circ R$. For $m \geq 1$ define*

$$E_{2m} = (L_B \circ A)^{2m}, \quad C_{2m}(h) = E_{2m}(h) \oplus h.$$

For every $\delta \in S^4$ and every 1024-bit state h ,

$$D_\delta E_{2m}(h) = \delta, \quad D_\delta C_{2m}(h) = 0.$$

Consequently all 256 distinct states of each coset $h \oplus S^4$ have the same feedforward output.

Proof. The parameter map $\tau(x, y) = (x \oplus y \oplus \sigma y, \sigma y)$ satisfies $\tau^2 = I$, since $\sigma^2 = I$. Every family mask stays in S^4 through each modified layer, so its output difference after $2m$ layers is its initial value. XOR with the input cancels that difference. The family has dimension eight, giving 256 distinct states per coset. $\square$

This is a collision family for an explicitly altered permutation followed by feedforward on freely chosen chaining states. The cosets need not be complete fibers. The construction uses the changed mixer and is not the specified FEILIAN hash or a fixed-IV message-collision attack.

References

- [1] Zhenzhen Bao, Jian Guo, San Ling, and Yu Sasaki. SoK: Peigen—a platform for evaluation, implementation, and generation of S-boxes. Cryptology ePrint Archive, Paper 2019/209, 2019. Published in IACR Transactions on Symmetric Cryptology, 2019(1); Section 3.1.5, Definitions 34–35. URL: <https://eprint.iacr.org/2019/209>.
- [2] Muhammad ElSheikh, Ahmed Abdelkhalek, and Amr M. Youssef. On MILP-based automatic search for differential trails through modular additions with application to Bel-T. Cryptology ePrint Archive, Paper 2019/476, 2019. URL: <https://eprint.iacr.org/2019/476>.
- [3] Xuejia Lai. Additive and linear structures of cryptographic functions. In Bart Preneel, editor, *Fast Software Encryption: Second International Workshop, Leuven, Belgium, December 14–16, 1994, Proceedings*, volume 1008 of *Lecture Notes in Computer Science*, pages 75–85. Springer, 1995. doi:10.1007/3-540-60590-8_6.
- [4] Gregor Leander, Cihangir Tezcan, and Friedrich Wiemer. Searching for subspace trails and truncated differentials. *IACR Transactions on Symmetric Cryptology*, 2018(1):74–100, 2018. doi:10.13154/tosc.v2018.i1.74-100.
- [5] Helger Lipmaa and Shiho Moriai. Efficient algorithms for computing differential properties of addition. Cryptology ePrint Archive, Paper 2001/001, 2001. The theorem and algorithm numbers cited in the text refer to this ePrint version, PDF pp. 7–8. URL: <https://eprint.iacr.org/2001/001>.
- [6] Haiwen Qin and Baofeng Wu. Towards non-independence of modular additions in searching differential trails of ARX ciphers: new automatic methods with application to SPECK and Chaskey. arXiv:2203.09741v1, 2022. Version of 18 March 2022; Section 3.2. URL: <https://arxiv.org/abs/2203.09741v1>.
- [7] Lei Wang, Ling Song, Yaobin Shen, Kaixuan Wang, Zicheng Shi, and Xin Yi. A hash function proposal FEILIAN, 2026. Specification dated 30 June 2026, in the official Round 1 archive. Sections 2.2.2, 4.5.2–4.5.3. Exact analyzed specification and scalar-source identifiers are given in Appendix A. URL: <https://www.niccs.org.cn/niccs/Proposal/Cryptographic%20Hash%20Algorithms/Round%201%20candidates/FEILIAN.zip>.