Exact linear correlations and the cost of Walsh-transform key recovery, with application to SPEEDY

Guoqiang Liu [1*] and Bing Sun[1]

[1*]College of Science, National University of Defense Technology, Fuyuan Road, Changsha, 410072, Hunan, China.

*Corresponding author(s). E-mail(s): liuguoqiang87@hotmail.com;

Abstract

When two S-box layers of a round are separated by no key addition, the round correlation is a signed sum over all compatible intermediate masks, not a product of layer correlations, so the product rule can fail in either direction. Our central finding is that evaluating this intra-round sum exactly changes the assessment of the published linear cryptanalysis of SPEEDY, whose two S-box layers are separated only by ShiftColumns. We first develop the linear cryptanalysis of this setting: an exact one-round algorithm with a decidable exactness condition for the product rule, a dependency-graph decomposition, a covering-number bound strengthening linear-trail weight bounds, and a Walsh-support criterion in which the affine dimension of that support, limited by the endpoint key masks, fixes the key-recovery transform cost. Potentials use the independent-round-key model; complexities are in equivalent encryptions. Applied to SPEEDY, these tools revise published results: a reported five-round mask sequence has exact correlation $2^{-90.0962}$, not $2^{-93.0147}$; the new bound raises the unrestricted five-round weight bound from 53.7714 to 62.2616 bits; and the full-round attack on SPEEDY-7-192 reported at time $2^{158.06}$ needs at least $2^{199.97}$ encryptions in the pruning class considered. For SPEEDY-6-192 we give a six-round known-plaintext attack (data $2^{169.84}$, time $2^{170.20}$, memory 2^{156}) and show that the attack class defined here contains no attack with data and time both at most 2^{128}, its time being at least $2^{136.302}$. The same exact evaluation also revises a four-round differential-linear correlation.

Keywords: Linear cryptanalysis, Exact intra-round correlation, Covering number, Fast Walsh transform, Key-recovery attack, SPEEDY

MSC Classification: 94A60 , 11T71 , 06E30

1 Introduction

In a key-alternating block cipher the correlation of one round factors into the correlations of its layers, because the round key separates consecutive nonlinear layers and randomises the intermediate mask between them. Low-latency designs break this pattern: to reduce the combinational depth of a fully unrolled hardware implementation they place two S-box layers inside one round with no key addition in between, and the correlation of that round becomes a signed sum over all compatible intermediate masks. SPEEDY [15] follows this design objective with a dedicated 6-bit S-box and two S-box layers per round. The family contains three instances, SPEEDY-r-192 for $r \in \{5, 6, 7\}$, each with a 192-bit block and a 192-bit key. SPEEDY-5-192 claims security up to 2^{64} data and 2^{128} time, whereas SPEEDY-6-192 claims security up to 2^{128} data and time. SPEEDY-7-192 claims security up to 2^{192} data and time. The two S-box layers are separated by ShiftColumns but not by a round-key addition. For fixed input and output masks, several intermediate masks may therefore contribute to the same one-round linear approximation. Their signed contributions must be summed; multiplying the correlations of two selected layer transitions is not valid in general.

This dependence has already affected differential cryptanalysis of SPEEDY. Beyne and Neyt [1] showed that the dominant characteristic used in the EUROCRYPT 2023 attack of Boura *et al.* [4] has key-averaged probability zero. They subsequently recomputed the reported characteristics using quasidifferential trails and invalidated several attacks whose probability estimates treated consecutive nonlinear layers independently [2]. Zhang [22] also gave a *related-key* differential attack on full-round SPEEDY-7-192 ($2^{186.2}$ chosen plaintexts, time $2^{160.13}$); [2] found that two one-round characteristics contained in the four-, five- and six-round characteristics of that paper (reported probabilities 2^{-42} and $2^{-44.2}$) both have probability zero, and gave alternatives with probabilities 2^{-49} and $2^{-47.83}$, respectively. Valid full-round differential attacks were later obtained in [2, 5, 6]. Wang *et al.* [20] proposed a linear key-recovery attack on full-round SPEEDY-7-192, with $2^{188.50}$ known plaintexts, time $2^{189.41}$, and memory $2^{185.04}$. Boura *et al.* [5] later showed that its key-recovery complexity was underestimated because the key-schedule dependency $l_{0,7}$ had been applied to the dimension of the Walsh transform. Nevertheless, the analysis of [5] retained the reported correlation $2^{-93.0147}$ of the underlying 5-round linear distinguisher.

The same within-round dependence also applies to linear correlations, but its effect is different because trail contributions are signed. Our central finding is that evaluating the intra-round correlation exactly changes the assessment of the published linear cryptanalysis of SPEEDY: it moves the correlation of the reported distinguisher, the weight bound that limits how good such a distinguisher can be, and the cost of the key recovery built on it. This work therefore first develops the exact treatment of such rounds and the constraints it places on the feasibility of key recovery, and then applies both to SPEEDY, re-evaluating the published distinguisher with exact one-round correlations and examining the consequences for the associated key-recovery procedure.

Our contributions.

1. We give and independently validate an exact algorithm for one-round correlations (Section 3). We formulate the corresponding sum for nonlinear segments without intervening key additions, give a decidable sufficient exactness criterion and a dependency-graph decomposition (Section 3.5), and derive structural bounds used in the subsequent trail searches.
2. Applying the algorithm to the published 5-round distinguisher changes the correlation of its reported mask sequence from $2^{-93.0147}$ to $2^{-90.0962}$ and gives independent-round-key bounds ELP $\geq 2^{-168.9034}$ for the reported external masks and ELP $\geq 2^{-166.3865}$ when the input mask is varied (Section 5). We exclude improvements obtained by changing any single interface mask position and reassess the associated full-round SPEEDY-7-192 key recovery: the affine dimension of its Walsh-spectrum support is 324, and the normalized costs of affine-subspace pruning restricted to 6-bit-aligned blocks are at least $2^{281.33}$ encryptions for one subspace and $2^{199.97}$ for a union (Section 6). These bounds have the scope stated in Theorem 7.
3. We generalise the partition quantity that [22] uses for differential counting to linear masks, no longer require the internal interface masks to be single-column, and place the resulting cyclic covering number in the trail-search state, raising the unrestricted 5-round trail lower bound from 53.7714 to 62.2616 bits (Section 4.4). For SPEEDY-6-192 we construct a 6-round known-plaintext attack with data, time, and memory $2^{169.84}$, $2^{170.20}$, and 2^{156} and, under the dominant-trail estimate and the stated FGNP/Walsh model, exclude the claimed 2^{128} data–time region for the defined attack class (Section 7). We also apply the exact correlation algorithm to revise the estimate of a 4-round differential-linear correlation from $2^{-23.0950}$ to $2^{-22.6909}$ (Section 8).

Comparison with existing results. Table 1 compares the key-recovery attacks on SPEEDY-6-192, whose security claim is $(2^{128}, 2^{128})$ in data and time. Table 2 lists the SPEEDY-5-192 results relevant to the differential-linear re-evaluation in Section 8.

Table 1 Known key-recovery attacks on SPEEDY-6-192.

Rounds	Method	Data	Time	Memory	Reference
3	Integral	$2^{17.6}$ CP	$2^{52.5}$	$2^{25.5}$	[17]
5	Diff.	$2^{97.93}$ CC	$2^{97.93}$	2^{96}	[5]
5.5	Diff.	$2^{126.75}$ CC	$2^{132.90}$	2^{42}	[2, 4]
6	Diff.	$2^{126.75}$ CC	$2^{156.77}$	2^{42}	[2, 4]
6	Linear	$2^{169.84}$ KP	$2^{170.20}$	2^{156}	This work[a]

Note. KP, CP and CC denote known plaintext, chosen plaintext and chosen ciphertext, respectively; time is measured in encryptions and memory in blocks. [a] The known-plaintext result and the chosen-ciphertext attacks use different data models and are not directly comparable.

No previous linear key-recovery result is available for full-round SPEEDY-6-192. The attacks in Table 1 therefore use different cryptanalytic methods and provide only

external reference points for data, time, and memory complexity. In particular, the relevant full-round comparison is with the differential meet-in-the-middle attack of [4], whose characteristic probability was recomputed in [2]. The 6-round differential attack of [21] is not listed in Table 1: it targets SPEEDY-7-192 reduced to six rounds (reported as $2^{158.04}$ chosen ciphertexts, time $2^{158.06}$, memory $2^{158.04}$, the first two figures being corrected to $2^{154.01}$ and $2^{154.03}$ in [2]), a version whose sixth round still contains MC, whereas our attack relies precisely on the absence of MC in the last round of SPEEDY-6-192; the two are therefore not interchangeable (Section 7). However, the attacks use different data models. The differential attack requires chosen ciphertexts, whereas our linear attack uses known plaintexts. Their complexities are therefore reported without claiming a direct ordering between the two attacks.

Section 7 establishes a valid linear key-recovery result using exact one-round correlations, and also a lower bound for the same modelled attack class: under the dominant-trail estimate, its normalized time is at least $2^{136.302}$, so the distance between the $2^{156.77}$ of the differential attack and our $2^{170.20}$ cannot be closed by tuning parameters inside the class. Under the time and memory limits of the revised differential attack, the trail bound exceeds the admissible weight, and this exclusion now holds for arbitrary internal interface masks. Consequently, neither full-round attack in Table 1 violates the 2^{128} time claim of SPEEDY-6-192. For linear cryptanalysis, Section 7.4 establishes a stronger result: the claim holds for class $\mathcal{A}$ under the dominant-trail estimate and the stated FGNP/Walsh cost model.

Table 2 Published attacks on SPEEDY-5-192 relevant to this work.

Rounds	Method	Data	Time	Memory	Reference
3	Integral	$2^{17.6}$ CP	$2^{52.5}$	$2^{25.5}$	[17]
4	Diff.	$2^{59.42}$ CC	$2^{118.11}$	2^{83}	[2, 20]
4	Diff.-lin.	2^{61} CP	2^{105}	2^{105}	[20][a]
4	Diff.	$2^{63.59}$ CC	$2^{66.18}$	$2^{63.59}$	[5]
5	Diff.	$2^{106.75}$ CC[b]	$2^{112.90}$	2^{42}	[2, 4]

Note. Notation and units are as in Table 1. [a] Section 8 re-evaluates only the distinguisher; the published key-recovery complexity is retained. [b] The data complexity exceeds the 2^{64} security claim of SPEEDY-5-192.

The 5-round result of Section 5 concerns the distinguisher independently of a key-recovery procedure. Section 6 then explains why it does not yield a feasible attack using the unpruned fast Walsh transform on full-round SPEEDY-7-192. By contrast, the 6-round attack of Section 7 uses the same three internal interface masks a_2, a_3, a_4 and uses the last round without MC.

The remainder of the paper is organised as follows. Section 2 introduces SPEEDY, mask propagation, and the FWT-based key-recovery framework. Section 3 presents exact one-round correlations, structural bounds, and the generalisation to a class of ciphers with consecutive nonlinear layers (Section 3.5). Sections 4 and 5 derive trail bounds and an improved 5-round distinguisher. Section 6 reassesses the key-recovery

attack of [20]. Section 7 describes the analysis of SPEEDY-6-192, and Section 8 revisits the 4-round differential-linear distinguisher. Section 9 concludes and states the limitations of the present results. A data-availability statement describing the programs submitted with the manuscript follows the conclusion.

2 Preliminaries

This section fixes the notation and introduces the three tools used repeatedly later: the specification of SPEEDY (Section 2.2), the cost function and compatibility conditions for one-round mask propagation (Section 2.3), and the linear key-recovery framework based on the fast Walsh transform (Section 2.4). A reader familiar with this material may proceed directly to Section 3, retaining only the cost function $\mathrm{cost}(\cdot)$ of Section 2.3 and the data threshold of 94.7573 bits derived at the end of Section 2.4.

2.1 Notation

For a Boolean function $f : \mathbb{F}_2^n \to \mathbb{F}_2$ the correlation is $c(f) = 2^{-n} \sum_x (-1)^{f(x)}$; for a vectorial Boolean function F and a mask pair (a, b),

$$c_F(a, b) = 2^{-n} \sum_x (-1)^{\langle b, F(x)\rangle \oplus \langle a, x\rangle}.$$

The linear potential is $\mathrm{LP} = c^2$. Throughout the linear-cryptanalysis estimates, we use the standard independent-round-key model, under which averaging over the round keys gives the expected linear potential ELP. For key recovery, the data complexities below use the dominant-trail estimate of the relevant linear approximation. The correlation weight of the selected trail is $w = -\log_2 |c|$, so its linear-potential contribution is 2^{-2w}; the dominant-trail estimate uses this value as the ELP estimate. We write $\mathrm{wt}(\cdot)$ for the Hamming weight, $\mathrm{supp}(\cdot)$ for the support, and $\mathrm{rotl}(v, s)$, $\mathrm{rotr}(v, s)$ for 32-bit cyclic shifts. The linearity of an S-box S is $\mathrm{Lin}(S) = \max_{a,b\neq 0} \left| \sum_x (-1)^{\langle b, S(x)\rangle \oplus \langle a, x\rangle} \right|$.

2.2 Description of SPEEDY-r-6ℓ

The state of SPEEDY-r-6ℓ is an $\ell \times 6$ bit matrix; we are concerned only with $\ell = 32$, i.e. a 192-bit block and a 192-bit key, the state being 32 rows by 6 columns. The round function of the first $r - 1$ rounds is

$$R_t = \mathsf{Ac}_t \circ \mathsf{MC} \circ \mathsf{SC} \circ \mathsf{SB} \circ \mathsf{SC} \circ \mathsf{SB} \circ \mathsf{Ak}_t, \qquad 0 \le t \le r - 2,$$

while the last round omits the second SC, the MC layer and the constant addition, and adds one extra round key instead:

$$R_{r-1} = \mathsf{Ak}_r \circ \mathsf{SB} \circ \mathsf{SC} \circ \mathsf{SB} \circ \mathsf{Ak}_{r-1}.$$

An r-round version therefore uses $r + 1$ round keys $k_0, \ldots, k_r$. The components are as follows.

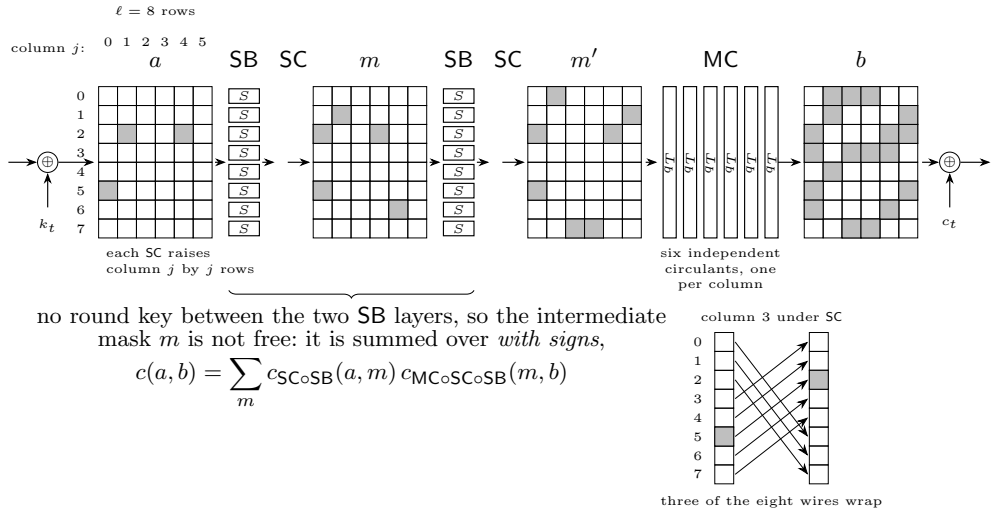


Fig. 1 Linear-mask propagation through one SPEEDY round, drawn for $\ell = 8$ so that it fits on the page; SPEEDY itself has $\ell = 32$, that is 32 rows and 6 columns.

- Ak (round-key addition): XOR the round key k_i at the beginning of round i. The round key is added once per round, and there is no key between the two SB layers. This is the starting point of everything technical in this paper.
- SB (S-box layer): apply the same 6-bit S-box S to each row of the state, i.e. 32 S-boxes in parallel. Its linearity is $\mathrm{Lin}(S) = 24$, so a single active S-box has minimum weight $\lambda = -\log_2(24/64) = 1.4150$; its differential uniformity is 8.
- SC (ShiftColumns): rotate column j upwards by j bits, $j = 0, \ldots, 5$.
- MC (MixColumns): apply the same 32×32 circulant matrix to every column, i.e. multiply by

$$q(x) = 1 + x + x^5 + x^9 + x^{15} + x^{21} + x^{26}, \qquad \mathrm{TAPS} = \{0, 1, 5, 9, 15, 21, 26\}$$

  in $\mathbb{F}_2[x]/(x^{32} + 1)$. Its transpose corresponds to q_T, and all mask propagation in this paper uses q_T. One has $\mathrm{wt}(q_T^{-1}) = 19$, a number that reappears as a hard lower bound in Section 4.3. The branch number of MC is 8, the theoretical maximum for a circulant matrix with seven nonzero coefficients.
- Ac (constant addition): the round constants are taken from the binary expansion of $\pi - 3$ and are irrelevant to linear cryptanalysis.

Figure 1 shows the propagation of a linear mask through one round for $\ell = 8$; SPEEDY uses $\ell = 32$. Shading denotes illustrative active bits. The S-box layer preserves rows, ShiftColumns rotates each column, and MixColumns acts independently on the columns. Because no key is added between the two S-box layers, the intermediate masks must be summed with their signs. The last round omits MC and the second SC; Section 7 uses this fact to avoid guessing key bits on the ciphertext side.

The key schedule is linear: k_0 is the master key and $k_{t+1} = P_B(k_t)$, where P_B is the affine permutation of bit positions $(6i' + j') \equiv \beta(6i + j) + \gamma \pmod{6\ell}$ with $\beta = 7$

and $\gamma = 1$. For SPEEDY-7-192 the round keys are $k_0, \ldots, k_7$; k_0 and k_7 are XORed onto the plaintext and the ciphertext respectively and are called outer keys; k_6 sits on the end of the distinguisher and only affects the sign of the correlation, so the number of inner keys in the framework of Section 2.4 is $L_{\text{int}} = 1$.

2.3 One-round mask propagation and the cost function

The mask a_t between two consecutive rounds is called an *interface mask*: a five-round approximation has six of them, $a_1, \ldots, a_6$, of which a_1 and a_6 are the endpoints and $a_2, \ldots, a_5$ are called the internal interface masks. A mask between the two SB layers of one round is always called an intermediate mask instead; the two are different. Searching for linear trails requires a quantity that estimates quickly how much a given interface mask costs. Writing the input mask a of a round as six 32-bit words $z_0, \ldots, z_5$, one per column, we define

$$\mathrm{Ra}(a) = \bigcup_j \mathrm{supp}(z_j), \quad \mathrm{Rw}(a) = \bigcup_j \mathrm{supp}\big(\mathrm{rotl}(q_T z_j,\, j)\big), \quad \mathrm{cost}(a) = |\mathrm{Ra}(a)| + |\mathrm{Rw}(a)|,$$

so that Ra collects the rows touched by a (the rows of the active S-boxes in the first SB layer) and Rw the rows touched after MC^T and SC^T. Since every active S-box contributes at least λ bits of weight, $\mathrm{cost}(a)$ is a lower bound on the weight associated with that interface mask.

Two dilation operators capture the range of the SC shifts:

$$\mathrm{dil}_{\downarrow}(v) = \bigvee_{s=0}^{5} \mathrm{rotr}(v, s), \qquad \mathrm{dil}_{\uparrow}(v) = \bigvee_{s=0}^{5} \mathrm{rotl}(v, s).$$

For three consecutive interface masks a_{t-1}, a_t, a_{t+1}, writing $o = \mathrm{Rw}(a_{t+1})$ and $p = \mathrm{Ra}(a_{t-1})$, compatibility is expressed by the four conditions

$$\begin{aligned} &\text{(A) } o \subseteq \mathrm{dil}_{\downarrow}(\mathrm{Ra}(a_t)), && \text{(B) } \mathrm{Ra}(a_t) \subseteq \mathrm{dil}_{\uparrow}(o), \\ &\text{(C) } \mathrm{Rw}(a_t) \subseteq \mathrm{dil}_{\downarrow}(p), && \text{(D) } p \subseteq \mathrm{dil}_{\uparrow}(\mathrm{Rw}(a_t)). \end{aligned}$$

Conditions (B) and (C) are linear constraints over GF(2), so the set of masks a_t satisfying both forms a linear space. This observation is the technical basis of the exhaustive enumeration in Section 5: we can enumerate that linear space instead of searching heuristically among 2^{192} masks. The search also uses per-column upper bounds on the one-round correlation, given by two tables indexed by the SC shift amount:

$$\begin{aligned} \mathrm{LAM}_1 &= (1.4150,\ 1.4150,\ 1.4150,\ 1.4150,\ 1.6781,\ 1.6781), \\ \mathrm{LAM}_2 &= (1.4150,\ 2,\ 2,\ 2,\ 1.4150,\ 1.4150). \end{aligned}$$

2.4 Key recovery with the fast Walsh transform

We use the key-recovery framework of Flórez-Gutiérrez and Naya-Plasencia [10] (the FGNP framework below), in the formulation of [11, Definition 1], which is the most

convenient one to instantiate term by term. Writing the empirical correlation of the distinguisher as

$$\nu(p, c, k_{\text{ext}}, k_{\text{int}}) = f_0(p, c) \cdot f(x \oplus k_{\text{ext}}, k_{\text{int}}),$$

with $x \in \mathbb{F}_2^l$ the concatenation of the plaintext and ciphertext parts and $l = |k_0| + |k_7|$, and denoting by L the size of the candidate key set and by L_{int} the size of the inner key set, the costs of the algorithms in this family and of its pruned and punctured variants are those of Table 13, where "each fast Walsh transform costs $l \cdot 2^l$ additions" is the phrasing of [11]. All of Section 6 consists in pinning down the three quantities l, d and L_{int} exactly for SPEEDY.

Data complexities are estimated with Selçuk's formula, and we take the definition of *advantage* from its source [18, Theorem 2]: an attack that ranks 2^m candidate values of the guessed key material has *a-bit advantage* when success is declared as soon as the right value is ranked within the best 2^{m-a} of them, i.e. when the retained fraction of the candidate space is 2^{-a}. With that convention, the right key is retained with probability P_S as soon as

$$N = \left(\Phi^{-1}(P_S) + \Phi^{-1}\left(1 - 2^{-(a+1)}\right)\right)^2 / \text{ELP},$$

where Φ is the distribution function of the standard normal law and $\text{ELP} = c^2$ is the expected squared correlation, so that $N = \left(\Phi^{-1}(P_S) + \Phi^{-1}(1 - 2^{-(a+1)})\right)^2 / (4\varepsilon^2)$ in the bias notation $\varepsilon = |p - 1/2| = |c|/2$ used in [18]. The exponent $-(a+1)$, rather than $-a$, is what the ranking statistic being an *absolute* correlation produces: retaining a fraction 2^{-a} of the candidates by absolute value cuts $2^{-(a+1)}$ from each tail. This is the form proved in [18, Theorem 2] and the one used throughout this paper; we note it explicitly because the variant with 2^{-a} also circulates, and the two differ by up to about one bit of data. For $a \geq 60$ the argument $1 - 2^{-(a+1)}$ rounds to 1 in double precision, so Φ^{-1} is evaluated by bisecting the identity $\operatorname{erfc}(z/\sqrt{2}) = 2^{-a}$ rather than by a rational approximation.

Wherever we want a negative statement to be as strong as possible (Section 6.4) we use the constant 5.6, which is the most favourable to the attacker; wherever we present a positive attack (Section 7) we use the full constant matching the target advantage. It follows that a full-round attack based on a 5-round distinguisher stays below the full codebook only if the total weight of its 5-round trail is below

$$\text{threshold} = (192 - \log_2 5.6)/2 = 94.7573 \text{ bits}.$$

This threshold is the reference point for every negative statement in Sections 4 and 6.4.

3 Exact one-round correlations for SPEEDY

This section is the technical foundation of the paper. We first explain why the layer-wise multiplication rule fails on SPEEDY and then give an exact algorithm (Section 3.1). We apply it to recompute the published 5-round distinguisher (Section 3.2), and verify the algorithm by four independent methods (Section 3.3). We also prove two counting lemmas and a structural theorem (Section 3.4), independently

verified by exact evaluation and used repeatedly as a pruning criterion in the searches that follow. Finally, Section 3.5 formulates the same correlation sum for nonlinear segments without intervening key additions, gives a two-sided error characterisation, a decidable sufficient exactness criterion, and a dependency-graph decomposition. Among the listed instances, full-round SPEEDY is the only one whose dependency graph remains too wide for direct tabulation; this observation does not exclude other exact algorithms.

3.1 Exact one-round correlation computation

Consider the part $\mathsf{SC} \circ \mathsf{SB} \circ \mathsf{SC} \circ \mathsf{SB}$ inside one round. Given an input mask u and an output mask v for that part, split it as $(\mathsf{SC} \circ \mathsf{SB}) \circ (\mathsf{SC} \circ \mathsf{SB})$; the composition rule for Walsh transforms then gives

$$c(u, v) = \sum_{m} c_{\mathsf{SC}\circ\mathsf{SB}}(u, m) \cdot c_{\mathsf{SC}\circ\mathsf{SB}}(m, v),$$

the sum ranging over all intermediate masks m, that is, over all masks in the position immediately after the first SC. A linear layer Λ acts on masks through its transpose, so that $c_{\Lambda\circ\mathsf{SB}}(x, y) = c_{\mathsf{SB}}(x, \Lambda^T y)$, and for the column shift $\mathsf{SC}^T = \mathsf{SC}^{-1}$. In a full round the outer SC is replaced by $\mathsf{MC} \circ \mathsf{SC}$ (Figure 1); in the last round the outer SC and MC are both absent, which is the case $\Lambda_2 = \mathrm{id}$ of Section 3.5. Had a round key been added between the two layers, averaging over that key would cancel the cross terms and leave only the squares of the individual terms, which is precisely why the layer-wise multiplication rule holds in that case. SPEEDY adds no such key, so the terms above are summed *with signs*, and $c(u, v)$ is a sum, not a single term.

In evaluating its 5-round distinguisher, [20] retained a single term of this sum per round (that is, one internal trail), obtaining $2^{-93.0147}$. This is exactly correct when the intermediate mask is unique (Corollary 1 in Section 3.5 turns this observation into a criterion), and yields a biased estimate otherwise. The direction of the bias cannot be assumed a priori: when the omitted terms share the sign of the main term the true correlation is larger, and when they do not it is smaller, while Theorem 3 shows that the underestimate is bounded by a few bits whereas the overestimate admits no bound at all: for SPEEDY there are mask pairs whose true correlation is *exactly zero* while the estimate is positive. On the particular masks of [20] the bias happens to be an underestimate, but that is a property of those masks, not of the rule. This behaviour is consistent with the differential analysis of [1], in which omitted quasidifferential trails can drive the probability down to zero.

The failure of layer-wise multiplication means that all compatible intermediate masks must be retained. Our algorithm therefore evaluates the above sum directly, but organises it column by column to keep the size under control. Given an input mask a and an output mask b of one round:

1. propagate b backwards to the output mask of the second SB layer and read off, row by row, the possible input masks and their correlations from the linear approximation table (the 64×64 table is precomputed once);

Table 3 Exact per-round correlations of the published 5-round distinguisher.

Round	Internal trails	Exact correlation (rational)	Gain (bits)
1	1	$+589\,824/64^5$	0
2	3042	$-N_2/64^{19}$	+2.751
3	1	$-786\,432/64^5$	0
4	15	$-N_4/64^{16}$	+0.168
5	1	$+9\,216/64^3$	0
Total ($-\log_2$): exact 90.0962, layer-wise 93.0147			+2.9184

Note. Gain is the improvement, in bits, of the exact correlation over the layer-wise estimate. The total gain 2.9184 is obtained from the exact rationals; subtracting the two totals as printed to four decimals gives 2.9185.

2. propagate a forwards to the input mask of the first SB layer and read off, row by row, the possible output masks;
3. the intermediate SC is a bit permutation, and aligns the row masks of both sides onto the same intermediate state mask;
4. sum over all compatible intermediate masks, giving the exact $c(a, b)$.

Exact arithmetic is essential: the summation must be carried out over the rationals rather than in floating point. Each entry of the linear approximation table is an integer in $[-32, 32]$, and a term with n active S-boxes has the form $(\prod \mathrm{LAT})/64^n$. We bring all terms to the common denominator 64^n, sum them exactly as integers, and only then take logarithms. The result is free of rounding error; this matters especially in round 2, where 3042 terms are added and a great deal of cancellation occurs.

3.2 Re-evaluation of Wang et al.'s five-round distinguisher

Applying the algorithm round by round to the mask trail reported by Wang et al. [20] gives Table 3.

The two large numerators are

$$N_2 = 357\,571\,687\,124\,785\,948\,524\,544, \qquad N_4 = 245\,854\,318\,407\,922\,483\,200,$$

whence

$$\text{published: } c = 2^{-93.0147},\ \mathrm{ELP} = 2^{-186.0294}; \quad \text{exact: } c = 2^{-90.0962},\ \mathrm{ELP} = 2^{-180.1925}.$$

The layer-wise multiplication rule underestimates the correlation by 2.92 bits. In rounds 1, 3 and 5 the space of candidate intermediate masks has size exactly one before filtering, so the rule happens to give the right answer there; the entire error is concentrated in rounds 2 and 4, the two rounds in which the internal trail is not unique.

These 2.92 bits deserve to be decomposed, because they contain two effects of a quite different nature. Write p_i for the contribution of the individual internal trails inside one round; they share the denominator 64^n and can therefore be compared as integers. For the very same pair of interface masks, the best single internal trail of round 2 is $2^{-36.2204}$, whereas the one drawn in Figure 6 of [20] is $2^{-38.5081}$: merely taking the best single trail already accounts for 2.288 bits, an amount that has nothing to do with the two S-box layers being coupled and only reflects that the published search was not optimal inside the round. Summation then adds a further 0.463 bits in round 2 and 0.168 bits in round 4 (whose drawn trail happens to be the best one). Hence

$$2.9184 = \underbrace{2.2877}_{\text{best single trail}} + \underbrace{0.6307}_{\text{intra-round summation}} .$$

Two directly measurable reasons explain why the summation term stays far below the ceiling that the number of terms would allow. The first is that the signs are almost perfectly balanced: of the 3042 terms of round 2, 1522 are positive and 1520 negative, with $\sum_i |p_i| = 2^{-30.0977}$ against $|\sum_i p_i| = 2^{-35.7574}$, so cancellation destroys 5.66 bits. The second is that the magnitudes decay quickly: the largest term carries only 1.44% of $\sum_i |p_i|$, the largest hundred carry 45.25% together, and the effective number of terms $\sum_i |p_i| / \max_i |p_i|$ is merely 69.7 rather than 3042. Had all 3042 terms attained the maximum magnitude with a common sign, the ceiling on the gain would have been 13.86 bits; the measured value is 2.751. Round 4 shows the opposite profile and the same conclusion: the largest of its 15 terms already carries 72.3% of $\sum_i |p_i|$, cancellation costs only 0.30 bits, and its gain of 0.168 bits is likewise far below the corresponding ceiling of 3.91 bits.

These two sets of figures show that linear and differential propagation behave differently. In the differential analysis of [1], quasidifferential trails drive the probability of the main characteristic to zero, i.e. complete cancellation; here the nearly balanced distribution of signs keeps the sum within the same order of magnitude as its largest term; the error is in the attacker's favour, but it cannot be large. Intra-round coupling therefore has to be treated exactly, since otherwise even the sign of these 2.92 bits is mispredicted, yet it does not overturn conclusions by an order of magnitude the way it does for differentials. This is also why intra-round summation and the cross-round linear hull are treated separately here: the former is summed exactly at the level of the algorithm, whereas the contribution of the latter is measured on its own and amounts to 0.0022 bits in the region of interest (Section 5).

3.3 Verification of the correlation algorithm

All subsequent results depend on the one-round correlations. We therefore validate the exact algorithm by four complementary methods: an independent exact computation, a complete Walsh transform on a reduced instance, direct enumeration without pruning, and sampling of the full-size round function.

- (V1) An independent computation over the rationals gives the five correlations of Table 3 exactly.

- (V2) On the reduced variant with $\ell = 4$ (a 24-bit block), the complete Walsh transform can be computed over 2^{24}. For both $\mathsf{SB} \circ \mathsf{SC} \circ \mathsf{SB} \circ \mathsf{SC}$ (without MC) and one full round (with MC) we evaluated 576 mask pairs, of which 378 have nonzero correlation; the exact algorithm agrees with the exhaustive computation in every case. This validates the correlation formula independently of the optimized enumeration.
- (V3) Direct enumeration of all intermediate-mask combinations without pruning agrees with the exact algorithm on the tested low-weight masks.
- (V4) Sampling on the real $\ell = 32$ round function. For the three rounds of Table 3 that can be sampled, we estimated the correlation on the real 192-bit keyless round function with 2^{26} random plaintexts; the deviations are 1.40σ, 0.63σ and 0.00σ. Twelve independent samples at scale 2^{20} give deviations between -1.24σ and $+0.65\sigma$.

3.4 Structural bounds for low-weight masks

Alongside the exact algorithm there is a purely structural statement, proved by counting rather than by evaluation. It depends on no search, yet it constrains the possible values of the interface mask weights throughout the paper; the exact algorithm then serves to verify it independently.

Everything in this subsection follows from one combinatorial property of the linear layer, which we isolate first. Write $\alpha(x) = \sum_{a \in \mathrm{TAPS}} x^a$ for the polynomial of the circulant MC. Since $\alpha(1) = |\mathrm{TAPS}| = 7$ is odd, α is invertible in $\mathbb{F}_2[x]/(x^{32}+1)$; MC^T acts on each column as multiplication by α, and the forward map on masks as multiplication by α^{-1}, whose weight is 19. Recall also that SC displaces column j by exactly j rows with $j \in \{0, \ldots, 5\}$, and that one round contains *two* SC layers. The second of them is a rotation inside each column, so it translates the objects counted below without changing their number or weight; it must nevertheless appear in the derivations.

Lemma 1 *Let b be a round output mask of weight 1. Every one-round linear trail ending in b activates all seven S-boxes of the second layer and at least four of the first. Its input mask therefore has weight at least 4, and the round activates at least eleven S-boxes.*

Proof Let the active bit of b lie in row i of column j_0. As MC^T acts column by column, $\mathsf{MC}^T b$ is supported in column j_0 alone, in the seven rows $i + \mathrm{TAPS}$, which are distinct modulo 32; the second SC layer translates them to $i + j_0 + \mathrm{TAPS}$, still seven distinct rows of a single column. Each of those rows is an S-box of the second layer with the nonzero output mask e_{j_0}, hence with a nonzero input mask y_r. Pulling y_r back through the first SC layer, a bit of y_r in column j originates in row $r + j$ with $0 \le j \le 5$, so the set A of active rows of the first layer meets $\{r, \ldots, r+5\}$ for each of the seven rows r. Equivalently, arcs of six consecutive residues, one ending at each element of A, must cover a translate of $\mathrm{TAPS} = \{0, 1, 5, 9, 15, 21, 26\}$ in $\mathbb{Z}_{32}$. No arc of length six holds four of these residues; exactly one triple, $\{0, 1, 5\}$, lies in a common arc; and among $9, 15, 21, 26$ only the pair $21, 26$ is at distance at most 5, the gaps $9 \to 15$ and $15 \to 21$ both being 6. So one arc covers at most $\{0, 1, 5\}$, the residues $9, 15, 21, 26$

need three more, and $|A| \geq 4$. Every active row of the first layer has a nonzero input mask, whence $\mathrm{wt}(a) \geq 4$ and the round activates at least $7 + 4 = 11$ S-boxes. □

Theorem 1 *Let R be one full round of SPEEDY (including MC). Then $c_R(a, b) = 0$ whenever $\mathrm{wt}(b) = 1$ and $\mathrm{wt}(a) \leq 3$. In particular, one full round has no $1 \to 1$ approximation.*

Proof By Section 3.1, $c_R(a, b)$ is the signed sum over all intermediate masks of the products of the two layers' correlations. Lemma 1 shows that for $\mathrm{wt}(b) = 1$ every intermediate mask with a nonzero product forces at least four active rows in the first layer; if $\mathrm{wt}(a) \leq 3$ there is no such mask and the sum is empty. □

The bound of the lemma is attained: the minimum number of active input rows is four for $j_0 \in \{0, 1, 3, 5\}$ and five for $j_0 \in \{2, 4\}$. We do not rely on the argument above but also evaluate all $192 \times 192 = 36\,864$ weight-1 to weight-1 cases exactly; every value vanishes. For comparison, the same sweep applied to $\mathsf{SC} \circ \mathsf{SB} \circ \mathsf{SC} \circ \mathsf{SB}$ without MC produces 4480 nonzero values, the best being $2^{-3.0931}$. Hence the vanishing correlations result from MC. All $3\,520\,512$ weight-1 to weight-2 mask pairs also have zero correlation, as predicted by Theorem 1.

The dual statement bounds the weight on the other side, and is what the searches of Section 4 actually use as a pruning criterion.

Lemma 2 *Let a be a round input mask of weight 1. Then every one-round linear trail starting from a has an output mask of weight at least 12, and the value 12 occurs.*

Proof Let the active bit of a lie in row r_0; only the S-box of that row is active in the first layer, with some output mask $m \neq 0$. The first SC layer moves the bit of column j from row r_0 to row $r_0 - j$, so the active rows of the second layer form a subset $R \subseteq \{r_0 - 5, \ldots, r_0\}$ of six *consecutive* rows. Let z be the mask at the output of the second layer, so that $\mathrm{supp} z[j] \subseteq R$ for every column j. The second SC layer turns column j into $x^{-j} z[j]$, and the output mask is $b[j] = \alpha^{-1} x^{-j} z[j]$; multiplication by x^{-j} is a rotation, so $\mathrm{wt}(b[j]) = \mathrm{wt}(\alpha^{-1} z[j])$. Enumerating the 63 nonempty subsets S of an interval of six consecutive rows gives $\min_S \mathrm{wt}(\alpha^{-1} \mathbf{1}_S) = 12$, reached only when S is a pair of rows at distance 4. At least one column of z is nonzero, so $\mathrm{wt}(b) \geq 12$. Equality occurs for the trail whose first-layer output mask is $e_0 + e_4$ and whose two active second-layer S-boxes, in rows r_0 and $r_0 - 4$, both have output mask e_0. □

The two lemmas explain the interface mask weight sequence $2, 11, 2, 9, 1, 12$ in Figure 6 of [20]: after an interface mask of weight 1 (the fifth position) the next interface mask must have weight at least 12, and the figure indeed shows exactly 12. This pattern is forced by the structure of the round function. An independent enumeration derived from the tap set and the linear approximation table recovers the covering minimum, all 63 six-row interval weights, and the witness above.

3.5 Generalisation to nonlinear segments without intervening key additions

The algorithm of the preceding subsections was written for SPEEDY, but the property it relies on is independent of SPEEDY's parameters: whenever no key is added between two nonlinear layers, layer-wise multiplication of correlations can fail. This subsection states that result for nonlinear segments without intervening key additions. We give a two-sided characterisation of the error, a decidable sufficient exactness criterion, and a decomposition theorem that splits the problem into smaller ones. The examples below illustrate the scope of the class; none of the statements in this subsection depends on a SPEEDY-specific assumption.

Definition 1 Let the state be n bits. Call

$$F = \Lambda_\nu \circ S_\nu \circ \Lambda_{\nu-1} \circ \cdots \circ \Lambda_1 \circ S_1$$

a ν-layer *nonlinear segment without intervening key additions* if each S_t partitions the state into disjoint words and applies an invertible S-box on each word (the S-boxes may differ between words and between layers, and the word widths may differ), each Λ_t is an invertible $\mathbb{F}_2$-linear map, and no key is added between S_t and S_{t+1}. Keys may occur only outside the segment; constants may be added inside it.

One full round of SPEEDY is such a nonlinear segment with $\nu = 2$, $n = 6\ell$, $S_1 = S_2 = \mathsf{SB}$, $\Lambda_1 = \mathsf{SC}$ and $\Lambda_2 = \mathsf{MC} \circ \mathsf{SC}$; the last round has $\Lambda_2 = \mathrm{id}$. The round keys are added outside the segment, and it is exactly this intra-round structure with $\nu = 2$ that causes the phenomenon of Section 3.1. Allowing constants inside the segment matters: a constant only changes the signs of the individual terms, and unlike averaging over a uniform key it does not make the cross terms vanish, so "replacing the key by a constant" does not restore layer-wise multiplication.

Theorem 2 (Exact correlation of one nonlinear segment) *Let F be as in Definition 1 and $a, b \in \mathbb{F}_2^n$. Writing $u_0 = a$ and $u_\nu = b$,*

$$c_F(a,b) = \sum_{u_1,\ldots,u_{\nu-1} \in \mathbb{F}_2^n} \prod_{t=1}^{\nu} c_{S_t}\big(u_{t-1},\ \Lambda_t^T u_t\big), \qquad c_{S_t}(x,y) = \prod_i c_{S_{t,i}}\big(x|_i,\ y|_i\big),$$

where i runs over the words of layer t. Every term of the sum is a rational number that can be written with the power of 2 corresponding to the total width of its active S-boxes as its denominator; after cancellation the reduced denominator may be smaller.

Proof The correlation of an invertible linear map Λ takes values in $\{0, 1\}$: $c_\Lambda(m, m') = 1$ if and only if $m = \Lambda^T m'$, and 0 otherwise. Applying the composition rule for correlation matrices [7] to each of the 2ν factors of F gives a sum over all intermediate masks; the $\{0, 1\}$ property then eliminates, one by one, the intermediate masks introduced by the ν linear layers, leaving $u_1, \ldots, u_{\nu-1}$. The word-wise factorisation holds because S_t is a direct sum over words. Constants inside the segment multiply each term by ± 1 and do not change the set of terms. □

The expression itself is standard and we claim no novelty for it: it is implicit in the correlation-matrix formalism of [7]. Our contribution is to treat it as a *computable* object (the pruning of Section 3.1 makes the instance $n = 192$, $\nu = 2$ summable term by term), together with the complete characterisation below of how large the error is when one does not do this.

Let $\mathcal{U}(a, b)$ denote the set of tuples $(u_1, \ldots, u_{\nu-1})$ of intermediate masks for which the product in Theorem 2 is nonzero; we call its elements the *internal trails* of (a, b). Let $\hat{c}(a, b) = \max_{u \in \mathcal{U}(a,b)} \left| \prod_t c_{S_t} \right|$ be the *best-single-internal-trail estimate*, with the convention $\hat{c}(a, b) = 0$ when $\mathcal{U}(a, b) = \emptyset$. This is what layer-wise multiplication reports once the best internal trail has been selected; the value reported for SPEEDY in [20] is the contribution of one specific internal trail, which need not be the best one (Section 3.2).

Lemma 3 *Let $\nu = 2$, write $w = \Lambda_2^T b$ and put*

$$\mathcal{V}(a, b) = \left\{ u \in \mathbb{F}_2^n \;\middle|\; \begin{array}{ll} u|_j = 0 & \text{for every word } j \text{ with } w|_j = 0, \\ (\Lambda_1^T u)|_i = 0 & \text{for every word } i \text{ with } a|_i = 0 \end{array} \right\}.$$

Then $\mathcal{V}(a, b)$ is a linear subspace of $\mathbb{F}_2^n$ and $\mathcal{U}(a, b) \subseteq \mathcal{V}(a, b)$, so $|\mathcal{U}(a, b)| \leq 2^{\dim \mathcal{V}(a,b)}$. The dimension $\dim \mathcal{V}(a, b)$ is obtained by one Gaussian elimination over $\mathbb{F}_2$, at a cost independent of $|\mathcal{U}(a, b)|$.

Proof Since the S-boxes are invertible, row 0 and column 0 of their linear approximation tables vanish except at $(0, 0)$; that is, $c_S(x, y) \neq 0$ forces $x = 0 \iff y = 0$. Applying this word-wise to the second layer, $u \in \mathcal{U}(a, b)$ and $w|_j = 0$ imply $u|_j = 0$; applying it to the first layer, $a|_i = 0$ implies $(\Lambda_1^T u)|_i = 0$. Both families of conditions are $\mathbb{F}_2$-linear equations in u, and their solution set is precisely $\mathcal{V}(a, b)$. The converse implication ($w|_j \neq 0$ forces $u|_j \neq 0$) is not linear, which is why one obtains an inclusion rather than an equality. □

Corollary 1 *If $|\mathcal{U}(a, b)| = 1$ then $|c_F(a, b)| = \hat{c}(a, b)$ and layer-wise multiplication is exact. In particular $\dim \mathcal{V}(a, b) = 0$ implies $|\mathcal{U}(a, b)| \leq 1$, in which case the rule is necessarily exact.*

This criterion explains what is seen in Table 3: rounds 1, 3 and 5 of that five-round approximation each have a single internal trail ($|\mathcal{U}| = 1$), so the values of [20] are exactly right on those three rounds, whereas rounds 2 and 4 have 3042 and 15 of them and are the only two rounds in error. The criterion is decidable, and by Lemma 3 deciding $\dim \mathcal{V} = 0$ does not require enumerating $\mathcal{U}$.

Theorem 3 (The error of layer-wise multiplication is two-sided) *Let F be as in Definition 1 with $\nu = 2$. Then*

1. *(bounded underestimate) $|c_F(a, b)| \leq |\mathcal{U}(a, b)| \cdot \hat{c}(a, b) \leq 2^{\dim \mathcal{V}(a,b)} \hat{c}(a, b)$: whenever the exact correlation exceeds the best-single-internal-trail estimate, it does so by at most $\dim \mathcal{V}(a, b)$ bits;*

2. *(unbounded overestimate) no lower bound of the form* $|c_F(a,b)| \geq \epsilon\, \hat{c}(a,b)$ *with a constant* $\epsilon > 0$ *can hold, because there exist mask pairs with* $\hat{c}(a,b) > 0$ *and* $c_F(a,b) = 0$*; such pairs exist already for SPEEDY itself (*$\ell = 32$*, all* 192 *bits).*

Proof Item (i) is the triangle inequality combined with Lemma 3. For item (ii) we exhibit a witness, written column-wise with the j-th column an element of $\mathbb{F}_2^{32}$:

$$a = (\texttt{00000008}, 0, \texttt{00000020}, 0, 0, 0), \qquad b = (0, 0, 0, 0, \texttt{0584137c}, 0),$$

that is, $\mathrm{wt}(a) = 2$ (the active rows are rows 3 and 5) and $\mathrm{wt}(b) = 12$. Here $\mathcal{U}(a,b)$ has exactly two elements, whose contributions over the common denominator $2^{24} = 64^4$ are -16384 and $+16384$, so that $\hat{c}(a,b) = 2^{-10}$ while $c_F(a,b) = 0$. An independent exact evaluation gives the same value $0/64^4$. □

The mechanism behind item (ii) differs from that of Theorem 1 and the two must be distinguished: the zero of Theorem 1 comes from an *empty sum* (no compatible intermediate mask exists), whereas the zero here comes from *cancellation* between two nonzero contributions. Both make layer-wise multiplication report a positive value where the exact correlation is zero, but only the latter shows that the natural remedy of "keeping several internal trails and taking the largest" cannot recover the correct value in principle. A corresponding phenomenon occurs in differential cryptanalysis: [1] observe that omitted quasidifferential trails can drive the probability down to zero.

Our last result shows that the instances of the class in which one *must* sum are in fact rare. The linear layer at the end of the segment can always be absorbed into the output mask, since $c_F(a,b) = c_{F'}(a, \Lambda_\nu^T b)$ with $F' = S_\nu \circ \Lambda_{\nu-1} \circ \cdots \circ \Lambda_1 \circ S_1$, so it suffices to consider F'.

Theorem 4 (Dependency-graph decomposition) *Take as vertices the S-boxes of* F' *together with the layer they belong to, and join two vertices by an edge whenever the output of word* i *of layer* t *influences, through* Λ_t*, the input of word* j *of layer* $t+1$*; call the resulting graph the* dependency graph *of* F'*. Let* $K_1, \ldots, K_s$ *be its connected components and let* $B_k^{(t)}$ *be the set of bit positions occupied by the words of* K_k *in layer* t*. Then* $|B_k^{(1)}| = \cdots = |B_k^{(\nu)}| =: m_k$*, the map* F' *is the direct sum of* s *mutually independent invertible* m_k*-bit maps, and*

$$c_{F'}(a,b) = \prod_{k=1}^{s} c_{F'_k}\big(a|_{B_k^{(1)}},\, b|_{B_k^{(\nu)}}\big),$$

that is, correlations multiply exactly *across components; each component can be handled directly by a table of size* 2^{m_k}*, so that no joint summation over internal masks across components is needed. The correlations inside a single component are still obtained by a Walsh transform on* m_k *bits.*

Proof By the definition of the edges, if the input of word j of layer $t+1$ depends on the output of word i of layer t then the two vertices lie in the same component; hence Λ_t maps the subspace spanned by $B_k^{(t)}$ into the subspace spanned by $B_k^{(t+1)}$. Both $\{B_k^{(t)}\}_k$ and $\{B_k^{(t+1)}\}_k$ are partitions of all n bit positions and Λ_t is invertible, so the dimensions of the images sum to n; therefore every inclusion must be an equality and $|B_k^{(t)}| = |B_k^{(t+1)}|$. Consequently

Table 4 Examples covered by Definition 1.

Construction	Reference	ν	n	Components	Exact evaluation
One full round of SPEEDY ($\ell = 32$)	[15]	2	192	1×192	summation required
One full round of SPEEDY ($\ell = 4$, reduced)	This work	2	24	1×24	tabulable
Piccolo's F-function	[19]	2	16	1×16	tabulable
First half of MISTY1's FI	[16]	2	16	1×16	tabulable
Two unkeyed AES rounds	[3, 12, 14]	2	128	4×32	per component
one of its 32-bit components	[8]	2	32	1×32	tabulable

Note. The Components column gives the connected components of the dependency graph; $s \times m$ denotes s components of width m. "Tabulable" means that a complete Walsh transform over all 2^m inputs can be computed directly, without the sum in Theorem 2.

F' is block-diagonal with respect to that partition, i.e. a direct sum of s invertible m_k-bit maps. Correlation is multiplicative over direct sums (take the Walsh transform component by component), which gives the stated product. □

Theorem 4 identifies when the dependency-graph decomposition does not reduce the evaluation to smaller independent components: the dependency graph is connected. Whether direct tabulation is practical then depends on the component width. Table 4 lists several members of the class together with their dependency-graph components, obtained directly from the graph definition. For contrast, AES itself has a round key between two consecutive S-box layers, so by Definition 1 its nonlinear segment has $\nu = 1$ layer, the sum degenerates to a single term and layer-wise multiplication holds in the average-over-keys sense; Simpira v2, AESQ and Haraka v2 remove that key and keep only a constant, which places them inside the present class.

For the present paper this subsection has two consequences. First, the algorithm of Section 3.1 is not a SPEEDY-specific trick but an algorithm for the class of Definition 1. Among the listed instances, the full-round SPEEDY construction has the largest state and a connected dependency graph, so direct tabulation is infeasible at the real parameter size. Second, Theorem 3 shows that whenever a correlation over such a construction is estimated by layer-wise multiplication, the direction of the error cannot be assumed a priori: the five-round estimate of [20] happens to be too small, but this is a property of that particular set of masks, not of the rule.

As in Section 3.3, the results above are verified in layers, and entirely independently of SPEEDY:

1. On random members with $\nu = 2$, $n = 16$, four random invertible 4-bit S-boxes per layer and random invertible matrices Λ_t, six instances gave 9000 mask pairs in total, every one of which was compared against the exhaustive 2^{16} Walsh transform, with complete agreement (0 mismatches). Of these, 6095 pairs are compatible, 1784 have a single internal trail (the rule is exact, in accordance with Corollary 1) and 4311 have several: 1994 are underestimated (by at most 2.5850 bits) and 1722 are overestimated (by at most 3.0000 bits, including 560 total cancellations), while a further 595 have error exactly zero. The measured values satisfy $|\mathcal{U}| \leq 32$ and $\dim \mathcal{V} \leq 8$, in agreement with Lemma 3 throughout.

2. On random members with $\nu \in \{3, 4, 5\}$ (with $(n, \text{word width}, \nu)$ equal to $(8, 4, 3)$, $(9, 3, 3)$, $(6, 3, 4)$, $(8, 2, 4)$ and $(6, 2, 5)$), the multiple sum of Theorem 2 again reproduced the exhaustive Walsh transform exactly on 375 mask pairs. On a further 60 mask pairs the pruned enumeration for $\nu = 2$ and the generic summation for arbitrary ν returned the same rational, the same number of terms and the same largest term.
3. On the SPEEDY toy instance with $\ell = 4$ ($n = 24$), 400 mask pairs were taken for the full round and 400 for the last round, and each was compared with both an independent exact evaluation and the exhaustive 2^{24} Walsh transform, with no discrepancy in either comparison. Among the 400 full-round pairs, 229 have several internal trails and 12 cancel completely, with $|\mathcal{U}| \leq 1272$ and $\dim \mathcal{V} \leq 12$.
4. On the real instance with $\ell = 32$, the five-round trail of [20] and the one of Section 5 were evaluated independently round by round, and all ten rational correlations agree. For the former, the per-round values of $|\mathcal{U}|$ are $1, 3042, 1, 15, 1$ and those of $\dim \mathcal{V}$ are $3, 24, 3, 12, 2$; the five gains from "best single trail" to the exact value sum to exactly 0.6307 bits, matching the decomposition $2.9184 = 2.2877 + 0.6307$ of Section 3.2. For the latter, the gain on round 3 is -0.0365 bits: the true correlation is *smaller* than the best single internal trail, so the error is two-sided even within one five-round approximation.
5. The witness of Theorem 3(ii) is not isolated: on the real $\ell = 32$ round function, 30 000 mask pairs constructed in the configuration described by Lemma 2 yielded 3088 pairs with several internal trails, of which 1524 are underestimated (by at most 4.5443 bits), 1209 are overestimated (by at most 6.5850 bits) and 186 cancel completely to zero.
6. On the unkeyed AES super-box ($n = 32$, four active input bytes against one active output byte, 200 mask pairs), *no* pair has a single internal trail: 139 are underestimated (by at most 2.9104 bits) and 61 are overestimated (by at most 9.6618 bits). The error is therefore not a peculiarity of SPEEDY.

4 Lower bounds for five-round linear trails

This section derives lower bounds on the weight of 5-round linear trails of SPEEDY. The threshold from Section 2.4 is 94.7573 bits: if all 5-round trails could be proved to have weight above it, no full-round linear attack based on a 5-round distinguisher would be possible. We derive the bounds in two stages. We first characterise interface masks as codewords of a linear code and apply a layered dynamic-programming recurrence to the codeword catalogue (Section 4.1). We then add the endpoint constraints imposed by key recovery (Section 4.3). Section 4.2 shows that the corresponding summation-based extension from single trails to linear hulls gives only a trivial bound. Section 4.4 then observes that what really operates in the compatibility conditions is a *covering* relation rather than a counting one, and uses this to raise by five to six active S-boxes the universal lower bound that places no restriction whatever on the internal interface masks; that strengthening is what the negative result of Section 7 rests on.

Table 5 Weight distribution of the quasi-cyclic code $\mathcal{C}$.

Weight w	8	10	12	14	16
A_w	32	96	1 888	23 168	227 880
Cumulative ($\leq w$)	32	128	2 016	25 184	253 064

4.1 Codeword representation and a single-column bound

The value z of an interface mask on a single column, together with its image $q_T z$ under MC^T, determines the cost $|\mathrm{supp}(z)| + |\mathrm{supp}(q_T z)|$ of that column. The set $\mathcal{C} = \{(z, q_T z) : z \in \mathbb{F}_2^{32}\}$ is a $[64, 32]$ quasi-cyclic code whose weight distribution is given in Table 5. Its minimum distance is 8, an exact value rather than a bound. All A_w with odd w vanish, a structural fact that halves the search space. The minimum cost of a multi-column interface mask is 9 (see Section 7.2).

This codeword catalogue also yields a mask-level trail bound. Taking the codewords with $B \leq 14$ (25 184 of them) times the six column positions gives 151 104 candidate interface masks. A layered dynamic-programming computation over this set (round-by-round transitions, pruned by weight) shows that every 5-round linear trail *whose internal interface masks all belong to this catalogue* has weight at least 65.4316 bits. This bound has two restrictions: the interface masks are single-column codewords of weight $B \leq 14$, and the search additionally fixes $\mathrm{wt}(z_1) \leq 3$ on the first interface mask, so the figure does not by itself cover every first-round mask a_1. The endpoint-constrained sweep of Section 4.3 removes the second restriction, and the cardinality-level recursion of the same subsection removes both at the price of a weaker bound. For comparison, the corresponding bound in the SPEEDY design document is 58.02 bits, which our cardinality-level recursion reproduces as 58.0165. The 7.4-bit improvement therefore results from restricting the class of interface masks and is not a universal strengthening of the bound.

The complete catalogue is obtained independently by layered subset enumeration and by a $65\,536 \times 65\,536$ meet-in-the-middle enumeration of the full 2^{32} space. Both methods give the weight distribution of Table 5, including all 253 064 codewords with $B \leq 16$ used in Section 7.2.

4.2 A linear-hull bound and the limitation of summation-based recursion

The layered recurrence bounds a single trail. A direct extension to a whole linear hull replaces "take the optimum" by "sum" in the recursion. We show that this extension gives only a trivial bound for SPEEDY.

We first give a quantity that can be computed. Write $\mathrm{ELP}_k(a, b)$ for the expected linear potential of round k alone. Parseval's relation $\sum_a \mathrm{ELP}_4(a_1, a) = 1$ yields the reduction $\mathrm{ELP}_5(a_1, a_6) \leq \max_a \mathrm{ELP}_k(a, a_6)$, with k the last round of the

approximation. Define the one-round absolute value (L_1) bound

$$\Lambda(b) := \sum_{v} \prod_{i} \frac{|\mathrm{LAT}[v_i][w_i]|}{64} \cdot \prod_{i:\, u_i \neq 0} \frac{\mu(u_i)}{64},$$

where $w = \mathsf{SC}^T(\mathsf{MC}^T b)$ is the mask that the round output mask b induces at the output of the second SB layer and w_i is its 6-bit value in row i; v is the mask between the two SB layers, with row i equal to v_i, and the sum runs over all v such that $\mathrm{LAT}[v_i][w_i] \neq 0$ for every row with $w_i \neq 0$; u is the output mask of the first SB layer, collected from v bit by bit through SC, that is $u_i = \bigoplus_{j=0}^{5}(v_{i-j} \wedge e_j)$ with row indices modulo 32 and e_j the j-th coordinate mask; and $\mu(y) := \max_x |\mathrm{LAT}[x][y]|$ with $\mu(0) = 64$. Since the round input mask a is free, each active S-box of the first layer contributes at most $\mu(u_i)/64$, so $\Lambda(b) \geq \max_a |C(a,b)|$, i.e. $\max_a \mathrm{ELP}_k(a,b) \leq \Lambda(b)^2$. We factor $\Lambda(b)$ over the three clusters $\{0,1,5,9\}$, $\{15\}$, $\{21,26\}$ dictated by the coupling structure of the MC coefficients, and evaluate it per column:

$$\Lambda = (2.0256,\ 4.5285,\ 4.4838,\ 4.6283,\ 3.0692,\ 2.3161) \quad \text{(bits, per column)}.$$

An attacker would pick the smallest column, so the one-round weight of a linear hull is at least 2.0256, i.e. $\mathrm{ELP} \leq 2^{-4.0511}$. The factorisation itself was cross-checked against twelve independent instances, all matching exactly.

Now to the failure of the summation-type recursion. If the recursion sums over all intermediate masks, the total admits the lower bound

$$\Omega = \sum_{a \neq 0} 2^{-2\lambda \cdot \mathrm{cost}(a)} \ \geq\ 2^{189.74} \cdot 2^{-181.12} = 2^{8.61} > 1.$$

Now $\Omega > 1$ means that the upper bound produced by this recursion exceeds one and therefore carries no information. For it to be nontrivial, the linearity of the S-box would have to be at most 22; that of SPEEDY is 24, a difference of 0.0850 bits per S-box. Thus the SPEEDY S-box lies outside the parameter range in which this summation-based recursion is nontrivial.

Table 6 also delineates the scope of the negative result: our analysis does not prove that SPEEDY resists 5-round linear distinguishers. The strongest bound that holds for arbitrary interface masks is 72.1669 (Section 4.4), and the strongest bound we have for the single-column class is 74.5068; these fall short of the required threshold (94.7573) by 22.59 and 20.25 bits respectively. After incorporating the covering constraints, the gap between the universal and single-column bounds decreases from 10.83 bits to 2.34. What we can say is that the best trail we located in the restricted setting has weight 104.0628, exceeding the threshold by 9.31 bits. Thus, no feasible attack was found within the searched class (Section 6.4), but this observation does not constitute a proof for that class. Throughout the paper, we distinguish proved bounds from experimental search results and universal bounds from those restricted to single-column interface masks.

Table 6 Summary of five-round trail-weight bounds.

Bound	Value	Nature	Scope
One-round hull bound	2.0256	Proved, weak	Arbitrary
Summation-type recursion	—	Trivial	Arbitrary
Cardinality-level DP	53.7714	Proved	Arbitrary
with endpoint constraints	63.6767	Proved	Arbitrary
with covering constraints (Section 4.4)	62.2616	Proved	Arbitrary
with endpoint and covering constraints	72.1669	Proved	Arbitrary
Mask-level DP	65.4316	Proved	Single-column, and $\mathrm{wt}(z_1) \le 3$
with endpoint constraints	74.5068	Proved	Single-column
Threshold required for an attack	94.7573	—	—
Best trail found (Section 6.4)	104.0628	Measured	Single-column

Note. Values are in bits of weight. "Arbitrary" imposes no restriction on the internal interface masks; "single-column" requires every internal interface mask to be supported in one column with codeword weight $B \le 14$. Only the four cardinality-level entries are universal; the strongest two use the covering constraints of Section 4.4.

4.3 Trail bounds under key-recovery endpoint constraints

Key recovery imposes additional constraints on both ends of the trail: the number of outer key bits to be guessed is determined by the endpoint masks,

$$|k_0| = 6 \cdot |\mathrm{dil}_{\uparrow}(\mathrm{Rw}(a_1))| = 6U, \qquad |k_7| = 6 \cdot |\mathrm{dil}_{\downarrow}(\mathrm{Ra}(a_6))| = 6D.$$

A naive fast Walsh transform costs $l \cdot 2^l$ additions with $l = |k_0| + |k_7|$. One 7-round encryption amounts to $7 \times 2 \times 32 = 448$ S-box lookups and $\log_2 448 = 8.807$, so that cost is $l \cdot 2^l / 2^{8.807}$ equivalent encryptions; this constant is used for every addition-to-encryption conversion in the paper. Requiring it to stay below the 2^{192} time claim of SPEEDY-7-192 gives $l + \log_2 l \le 200.807$, hence $l \le 193$, and since $l = 6(U + D)$ is a multiple of six, $l \le 192$, i.e. $U + D \le 32$; at $l = 192$ the transform costs $2^{199.585}$ additions, that is $2^{190.778}$ equivalent encryptions. Hence a trail of very low weight is useless for an attack if its endpoints make $U + D$ too large; conversely, restricting $U + D$ increases the minimum attainable weight. It is this trade-off that the present subsection quantifies.

Complete 2^{32} enumeration at each endpoint gives the two exact lower-bound tables in Table 7. Two entries have a direct structural reading: $g[6] = 7 = |\mathrm{TAPS}|$ and $h[6] = 19 = \mathrm{wt}(q_T^{-1})$.

Applying these two tables in the layered recurrence yields the mask-level restricted bounds below. All four entries concern the single-column interface mask class of Section 4.1; the unconstrained entry carries in addition the start restriction $\mathrm{wt}(z_1) \le 3$, whereas the three constrained runs dispense with it and therefore hold for every first-round mask a_1.

Table 7 Minimum additional weight under endpoint constraints.

D (ciphertext side)	6–17	18–20	21–26	27–28	29–31	32			
$g[D]$	7	6	4	3	2	1			
U (plaintext side)	6	7–9	10–11	12	13–15	16–20	21–22	23–31	32
$h[U]$	19	14	12	10	9	6	5	2	1

Constraint	Bound (bits)	Increase over unconstrained
None	65.4316	—
Ciphertext side only	73.9218	+8.4902
Plaintext side only	66.0165	+0.5849
Both sides ($U = 26$, $D = 6$)	74.5068	+9.0752

Since the minimum weight is nonincreasing in both U and D, it suffices to sweep the boundary $U + D = 32$. Of the twenty-one splits (U, D), seventeen are nonempty and give bounds ranging from 74.51 to 85.58, the minimum 74.5068 being attained at $(26, 6)$, $(25, 7)$ and $(24, 8)$. The four empty splits are exactly those with $U \leq 9$, so $|k_0| \geq 60$ holds in any linear attack on SPEEDY whose 5-round distinguisher is built from single-column internal interface masks.

A cardinality-level dynamic-programming relaxation tracks the numbers of active S-boxes on both sides of each interface mask. For single-column interface masks it gives 58.0165 bits, unchanged by conditions (B)/(D) and the exact (p, q) spectrum. With no restriction whatever on the internal interface masks, the unrestricted and endpoint-constrained bounds are 53.7714 and 63.6767 bits, respectively. Quantitatively, the best trail used by the 6-round attack (Section 7.3, weight 82.1012) lies 16.67 bits above the mask-level unconstrained bound and 28.33 bits above the universal one, while in the endpoint-constrained setting the best trail we found (104.0628, Section 6.4) lies 29.56 bits above 74.5068 and 40.39 bits above 63.6767. We state these gaps explicitly as a quantitative account of the limitations of the relaxation. Section 4.4 strengthens the two universal bounds to 62.2616 and 72.1669, respectively.

4.4 Covering numbers and a stronger bound on internal interface masks

The cardinality-level relaxation of Section 4.3 has the same optimum for every plaintext-side constraint:

$$\big[\, P_1 = h[U],\ (2,7),\ (2,7),\ (2,7),\ (1,7),\ Q_6 = 1 \,\big], \qquad (P_t, Q_t) := \big(|\mathrm{Ra}(a_t)|,\ |\mathrm{Rw}(a_t)|\big),$$

with $h[U] + 36$ active S-boxes in total, so that $f(U) = 53.7714 + (h[U] - 2)\lambda$ holds exactly for every U in Table 7. All three internal interface masks of this state sequence take $(P, Q) = (2, 7)$, whose $B = 9$ is odd; since the code $\mathcal{C}$ has no codeword of odd weight (Table 5), they are necessarily multi-column interface masks. In other words,

what makes the universal bound weak is not that multi-column masks have gone unsearched, but that the cardinality shadow of conditions (A)/(D) is far too permissive on exactly those masks. This subsection introduces a quantity that still depends only on row sets, yet is strictly stronger than that shadow, and thereby raises the universal bound by five to six active S-boxes.

SC raises column j by j rows ($0 \le j \le 5$), so $\mathrm{dil}_{\downarrow}(V)$ is precisely the union, over $r \in V$, of the cyclic intervals $[r-5, r]$ of length six, and $\mathrm{dil}_{\uparrow}(V)$ is the union of the intervals $[r, r+5]$. Condition (A) therefore says that every row of $\mathrm{Rw}(a_{t+1})$ must lie in a cyclic interval of length six whose right end is a row of $\mathrm{Ra}(a_t)$. This suggests tracking the *cyclic covering number* of a row set,

$$\kappa(V) := \min\big\{\, |W| \;:\; W \subseteq \mathbb{Z}_{32},\ V \subseteq \textstyle\bigcup_{r\in W}[r, r+5] \pmod{32} \,\big\},$$

the least number of cyclic intervals of length six needed to cover V. Since six such intervals already cover all of $\mathbb{Z}_{32}$, we always have $\kappa(V) \le 6$; and clearly $\lceil |V|/6 \rceil \le \kappa(V) \le |V|$. Here $\mathbb{Z}_{32}$ denotes the row indices modulo 32, and the intervals $[r, r+5]$ are understood cyclically.

The quantity itself is not new: it coincides with the *partition Hamming weight* phw_5 of [22]. That paper partitions the active rows, in order, into sets of diameter at most five, and takes phw_5 to be the least number of such sets; since a partition into sets of diameter at most five corresponds exactly to a cover by cyclic intervals of length six, $\kappa = phw_5$. It uses the quantity to define a *partition branch number*, and thereby obtains the minimum active-S-box counts 13, 23, 35, 45, 57, 67 for differential trails over two to seven rounds. We should also point out that the matrix of MC is circulant, so the coordinate reversal $r \mapsto -r$ turns it into its transpose while leaving both wt and κ invariant; hence those counts apply to linear trails as well, within the same interface mask class. We should also point out that *pbn* is a per-column quantity: the x in its definition ranges over $\mathbb{F}_2^{32}$, that is, over one column of the state; to use it as a lower bound on the active S-boxes of two full rounds, that paper sets, "without loss of generality", one column of the input and of the output of MC as active. In the counting this is not without loss: distinct columns of a multi-column interface mask may share an active row, and a row is a single active S-box however many of its bits are active, so a per-column count overestimates the number of active S-boxes and cannot be added up into a universal bound. Our generalisation lies in two further points. First, we no longer require the internal interface masks to be single-column, so that the bound holds for arbitrary interface masks. This is necessary, because the three internal interface masks of the cardinality-optimal state sequence at the beginning of this section have $B = 9$, an odd number, and must therefore span several columns (Section 4.1). Second, instead of converting the count into an estimate of the form "a power of the per-S-box maximum", we carry κ as a state variable in the layered dynamic program, together with the per-S-box exact weight tables LAM_1 and LAM_2.

Lemma 4 *For any pair of adjacent interface masks a_t, a_{t+1}, the compatibility conditions (A) and (D) imply respectively*

$$|\mathrm{Ra}(a_t)| \;\ge\; \kappa\big(\mathrm{Rw}(a_{t+1})\big), \tag{K1}$$

$$|\mathrm{Rw}(a_{t+1})| \;\geq\; \kappa\big(\mathrm{Ra}(a_t)\big). \tag{K2}$$

Proof Condition (A) reads $\mathrm{Rw}(a_{t+1}) \subseteq \mathrm{dil}_{\downarrow}(\mathrm{Ra}(a_t)) = \bigcup_{r\in\mathrm{Ra}(a_t)}[r-5, r]$, whose right-hand side is a union of $|\mathrm{Ra}(a_t)|$ cyclic intervals of length six. Thus $\mathrm{Ra}(a_t)$ exhibits a cover of $\mathrm{Rw}(a_{t+1})$, whose size is at least that of a minimum cover, which is (K1). Condition (D) reads $\mathrm{Ra}(a_t) \subseteq \mathrm{dil}_{\uparrow}(\mathrm{Rw}(a_{t+1}))$ and gives (K2) in the same way. □

Because $\lceil |V|/6 \rceil \leq \kappa(V)$, the constraints (K1)/(K2) imply the cardinality shadow $Q_{t+1} \leq 6P_t$, $P_t \leq 6Q_{t+1}$ used in Section 4.3, so replacing the latter by the former loses nothing. The gap between the two can be large, and the worst case for κ is precisely the tap set of MC itself. By the count in Lemma 1, the only taps of TAPS $= \{0, 1, 5, 9, 15, 21, 26\}$ that share a cyclic interval of length six are the groups $\{0, 1, 5\}$ and $\{21, 26\}$, so any cover must use at least the four intervals $\{0, 1, 5\}$, $\{9\}$, $\{15\}$, $\{21, 26\}$:

$$\kappa(\mathrm{TAPS}) = 4 \quad \text{(and likewise for all 32 cyclic translates)},$$

whereas the counting bound gives only $\lceil 7/6 \rceil = 2$, off by a factor of two.

The lemma below shows that the internal interface masks of the cardinality-optimal sequence ($|\mathrm{Ra}(a)| = 2$, $|\mathrm{Rw}(a)| = 7$) necessarily satisfy $\kappa(\mathrm{Rw}(a)) = 4$, so (K1) forces the preceding interface mask to have $|\mathrm{Ra}(a_t)| \geq 4$, contradicting the $|\mathrm{Ra}(a_t)| = 2$ of that sequence. It is this incompatibility that excludes the cardinality-optimal sequence.

Lemma 5 *Let an interface mask a satisfy $|\mathrm{Ra}(a)| = 2$ and $|\mathrm{Rw}(a)| = 7$. Then $\mathrm{Rw}(a)$ is a cyclic translate of* TAPS*, and hence $\kappa(\mathrm{Rw}(a)) = 4$.*

Proof Write a column-wise as $z_0, \ldots, z_5$. Every active column satisfies $\mathrm{supp}(z_j) \subseteq \mathrm{Ra}(a)$, hence $\mathrm{wt}(z_j) \leq 2$. If some column had $\mathrm{wt}(z_j) = 2$, then $\mathrm{wt}(q_T z_j) \geq 8$ by Table 5 (the 32 codewords of weight 8 are exactly the 32 with $\mathrm{wt}(z) = 1$, so $\mathrm{wt}(z) = 2$ forces $B \geq 10$), but $\mathrm{supp}(\mathrm{rotl}(q_T z_j, j)) \subseteq \mathrm{Rw}(a)$ and rotation preserves weight, so $|\mathrm{Rw}(a)| \geq 8$, a contradiction. Every active column therefore has $\mathrm{wt}(z_j) = 1$, its image under MC^T is a full translate of TAPS with seven rows, and that translate is contained in the seven-row set $\mathrm{Rw}(a)$; the two are thus equal. □

Consequently the three $(2, 7)$ interface masks of this state sequence, legal though each is on its own, cannot be chained: by (K1) each of them demands $P \geq 4$ of the preceding interface mask, whereas the sequence supplies $P = 2$. The covering number must therefore be included in the dynamic-programming state so that (K1)/(K2) hold at every internal interface mask.

We enlarge the state from (P, Q) to the quadruple $\big(P, Q, \kappa(\mathrm{Ra}), \kappa(\mathrm{Rw})\big)$, impose (K1)/(K2) on transitions, and build a cost table $\mathrm{cost}[P][Q][k_A][k_W]$ in three regimes:

- $P + Q \leq 16$ with a single-column interface mask: we use the exact quadruple spectrum of the complete codeword catalogue for $B \leq 16$ (253 064 entries, Table 5), where (k_A, k_W) is literally $(\kappa(\mathrm{supp}z), \kappa(\mathrm{supp}q_T z))$, and take the cost to be the minimum over the lists $\mathrm{LAM}_1, \mathrm{LAM}_2$;

- $P + Q \leq 16$ with several columns: the cost is $(P + Q)\lambda$ subject to $P + Q \geq 9$ (the minimum cost of a multi-column interface mask, Section 7.2); the admissibility test is Lemma 6;
- $P + Q \geq 17$: no additional spectral restriction is imposed, the cost is taken to be $(P + Q)\lambda$, and κ is free in $[\lceil P/6 \rceil, \min(P, 6)] \times [\lceil Q/6 \rceil, \min(Q, 6)]$. This regime is crude, but $(P + Q)\lambda$ is a valid lower bound on the cost of any interface mask with $P + Q \geq 17$ active rows, and every pair (k_A, k_W) in the covering range is admitted, so no realisable trail is excluded.

The admissibility test of the second regime must make the admissible state set a *superset* of the realisable one, or the bound fails. The following lemma supplies the necessary condition we use.

Lemma 6 *Let an interface mask a satisfy $|\mathrm{Ra}(a)| = P$, $|\mathrm{Rw}(a)| = Q$, $\kappa(\mathrm{Ra}(a)) = k_A$, $\kappa(\mathrm{Rw}(a)) = k_W$ and $P + Q \leq 16$. Then there exists a codeword $(z, q_T z) \in \mathcal{C}$ satisfying all of*

$$\mathrm{wt}(z) \leq P, \quad \mathrm{wt}(q_T z) \leq Q, \quad \kappa(\mathrm{supp} z) \leq k_A, \quad \kappa(\mathrm{supp} q_T z) \leq k_W.$$

Proof Take any active column $z := z_j$ of a. From $\mathrm{supp}(z) \subseteq \mathrm{Ra}(a)$ we get $\mathrm{wt}(z) \leq P$; from $\mathrm{supp}(\mathrm{rotl}(q_T z, j)) \subseteq \mathrm{Rw}(a)$ and the invariance of weight under rotation, $\mathrm{wt}(q_T z) \leq Q$. Now κ is monotone under inclusion (if $V \subseteq V'$ then any cover of V' covers V) and invariant under cyclic translation, so $\kappa(\mathrm{supp} z) \leq \kappa(\mathrm{Ra}(a)) = k_A$ and $\kappa(\mathrm{supp} q_T z) = \kappa(\mathrm{supprotl}(q_T z, j)) \leq \kappa(\mathrm{Rw}(a)) = k_W$. Finally $B(z) = \mathrm{wt}(z) + \mathrm{wt}(q_T z) \leq P + Q \leq 16$, so the codeword lies in the catalogue. □

Hence "some catalogued codeword satisfies these four inequalities" is a necessary condition, and admitting states accordingly is a legitimate relaxation. The test itself is answered in $O(1)$ time by a table built from cumulative ORs along the four coordinates. As an immediate corollary, $(P, Q, k_A, k_W) = (2, 7, 1, k_W)$ is admissible only for $k_W = 4$: in the full spectrum the only codewords with $\mathrm{wt}(z) \leq 2$ and $\mathrm{wt}(q_T z) \leq 7$ are the 32 with $\mathrm{wt}(z) = 1$, and all of them have $\kappa(\mathrm{supp} q_T z) = 4$, which recovers Lemma 5.

We now write the layered recursion obtained after incorporating κ in explicit form, so that it can be checked term by term. For an interface mask a_t put $(P_t, Q_t, k_{A,t}, k_{W,t}) = \big(|\mathrm{Ra}(a_t)|, |\mathrm{Rw}(a_t)|, \kappa(\mathrm{Ra}(a_t)), \kappa(\mathrm{Rw}(a_t))\big)$, let Θ be the set of quadruples declared admissible by the three regimes above, and let $c(\cdot)$ be the value of the cost table on it.

Lemma 7 (Covering-number layered recursion) *Let $p_{\min}, q_{\min} \geq 1$. Put*

$$F_0(P, k_A) = \lambda \cdot \min\big\{Q : \ Q \geq \max(q_{\min}, k_A, 1), \ \lceil Q/6 \rceil \leq \min(P, 6)\big\},$$

and for $1 \leq m \leq 4$

$$F_m(P, k_A) = \min_{\substack{(P', Q', k'_A, k'_W) \in \Theta \\ k'_W \leq \min(P, 6), \ Q' \geq \max(k_A, 1)}} \Big\{ c(P', Q', k'_A, k'_W) + F_{m-1}(P', k'_A) \Big\}.$$

Then every five-round linear trail whose endpoints satisfy $P_1 \geq p_{\min}$ and $Q_6 \geq q_{\min}$ has weight at least

$$w_{\min}(p_{\min}, q_{\min}) = \min_{\substack{P_1 \geq p_{\min} \\ \lceil P_1/6 \rceil \leq k_{A,1} \leq \min(P_1,6)}} \{\lambda P_1 + F_4(P_1, k_{A,1})\}.$$

Taking $p_{\min} = h[U]$ and $q_{\min} = g[D]$ (or 1 at an unconstrained end) gives Table 8 and the two-sided endpoint-constrained bound in the sense of Section 4.3.

Proof In four steps. First, the weight decomposition. $\mathrm{Ra}(a_t)$ is the set of active rows of the first SB layer of round t and $\mathrm{Rw}(a_t)$ that of the second SB layer of round $t-1$, so round t has $P_t + Q_{t+1}$ active S-boxes and the five rounds have

$$N = P_1 + \sum_{t=2}^{5} (P_t + Q_t) + Q_6$$

of them in total; the remaining endpoint quantities Q_1 and P_6 fall in the guessed outer rounds and do not enter the weight of the trail. Second, the cost of the internal interface masks. For $2 \leq t \leq 5$ the true quadruple of a_t lies in Θ through one of the three regimes, namely the first (single column, exact quadruple spectrum), the second (several columns, Lemma 6) or the third ($P + Q \geq 17$, no spectral restriction), and the weight associated with a_t is at least $c(P_t, Q_t, k_{A,t}, k_{W,t})$. Third, the transition constraints. By Lemma 4 adjacent interface masks satisfy $k_{W,t+1} \leq P_t$ and $Q_{t+1} \geq k_{A,t}$, and $\kappa \leq 6$ gives $k_{W,t+1} \leq \min(P_t, 6)$; these are the two constraints in F_m, while $\lceil Q/6 \rceil \leq \min(P, 6)$ in F_0 is the feasibility form of the same constraint between Q_6 and $k_{W,6}$. Fourth, the endpoints. For a_1 and a_6 only the trivial range $\lceil \cdot/6 \rceil \leq \kappa \leq \min(\cdot, 6)$ is used and no spectral restriction is imposed, so all of the sharpening comes from the four internal interface masks; and $P_1 \geq h[U]$, $Q_6 \geq g[D]$ are supplied by the two endpoint tables of Table 7. The state set traversed by the recursion is therefore a superset of the realisable one, and its minimum is a lower bound on the weight. □

One feature of the recursion deserves a remark. Since $\mathrm{LAM}_1[0] = \mathrm{LAM}_2[0] = \lambda$, the per-column minimum in the first regime is always attained at $j = 0$ and equals $(P + Q)\lambda$. Checking the table entry by entry, all 10 601 finite entries equal $(P + Q)\lambda$ without exception: the three regimes differ only in which quadruples they declare admissible, not in the cost they assign. That is why every entry of Table 8 is an integer multiple of λ, the strengthening contributed by the covering number being purely combinatorial. Evaluating the recursion, the unconstrained value is $w_{\min}(1, 1) = 62.2616 = 44\lambda$, attained at

$$\big[\, P_1 = 2,\ (6,6,2,2),\ (2,7,1,4),\ (6,6,2,2),\ (1,7,1,4),\ Q_6 = 1 \,\big];$$

under the two-sided setting $U + D \leq 32$ the minimum is $72.1669 = 51\lambda$, attained at every split with $h[U] = 2$ and $g[D] = 7$, that is at $(U, D) = (26, 6)$, $(25, 7)$, $(24, 8)$ and $(23, 9)$, with optimal state sequence

$$\big[\, P_1 = 2,\ (6,6,2,2),\ (2,7,1,4),\ (6,6,2,2),\ (2,7,1,4),\ Q_6 = 7 \,\big].$$

Table 8 gives the universal lower bounds obtained after incorporating κ. Replacing (K1)/(K2) by their cardinality relaxation recovers the bounds of Section 4.3, including

Table 8 Universal five-round trail bounds by plaintext-side key-guessing width.

U	6	7–9	10–11	12	13–15	16–20	21–22	23–32
$\lvert k_0\rvert$	36	42–54	60–66	72	78–90	96–120	126–132	138–192
Cardinality-only	77.8271	70.7519	67.9218	65.0917	63.6767	59.4316	58.0165	53.7714
Covering-number	84.9022	77.8271	74.9970	72.1669	70.7519	66.5068	65.0917	62.2616
Gain	+7.08	+7.08	+7.08	+7.08	+7.08	+7.08	+7.08	+8.49
Active S-boxes	60	55	53	51	50	47	46	44

Note. The key-guessing width is $\lvert k_0\rvert = 6U$. The cardinality-only bound is from Section 4.3; the covering-number bound incorporates (K1)/(K2). Neither bound restricts the internal interface masks in any way.

53.7714 without endpoint constraints and 63.6767 with constraints on both ends. Independently, exhaustive subset enumeration agrees with the computed covering number on all 4 531 tested row masks.

The gain is five active S-boxes for $U \le 22$ and six for $U \ge 23$. With neither end constrained the bound rises from 53.7714 (38 S-boxes) to 62.2616 (44); under the two-sided endpoint setting $U + D \le 32$ of Section 4.3 it rises from 63.6767 to 72.1669, the optimum being every split with $h[U] = 2$ and $g[D] = 7$, for instance $U = 26$, $D = 6$. The latter figure still falls 22.59 bits short of the attack threshold 94.7573 (previously 31.08), so the negative conclusion of Section 6.4 about full-round SPEEDY-7-192 is unaffected. As an external reference point, the five-round count of [22] for single-column interface masks is $pbn_5 = 45$ active S-boxes, that is $45\lambda = 63.68$ bits: the 44 S-boxes of this section are slightly fewer in count, but the corresponding bound holds for arbitrary interface masks; and the mask-level bound obtained in Section 4.3 under the same restriction is 65.4316 bits (about 46 S-boxes, with the start restriction $\mathrm{wt}(z_1) \le 3$), already above that count.

The optimum retained after incorporating κ shows which state sequences are excluded by the covering-number constraints. For $U = 16$, the optimal chain is

$$\big[\, P_1 = 6,\ (2,7,1,4),\ (6,6,2,2),\ (4,7,1,4),\ (1,7,1,4),\ Q_6 = 1 \,\big],$$

the quadruples being ordered as $(P, Q, \kappa(\mathrm{Ra}), \kappa(\mathrm{Rw}))$, for a total of $6 + 9 + 12 + 11 + 8 + 1 = 47$ active S-boxes. The optimal chain contains a $(6, 6)$ interface mask between any two low-cost $(2, 7)$ states: the former has six active rows on each side and κ equal to 2 on both, costing $12\lambda = 16.9804$ bits. Among the admissible states that supply $P \ge 4$ to their predecessor, it has the minimum cost. Such interface masks do exist and are single-column: the catalogue contains 32 codewords with $(6, 6, 2, 2)$, for instance $z =$ `00001c31` and $q_T z =$ `4c034000`, whose row sets are $\{0, 4, 5, 10, 11, 12\}$ and $\{14, 16, 17, 26, 27, 30\}$.

Three consistency checks are independent of the lower-bound computation. First, (K1)/(K2) hold on the constructed trails: round by round, the 5-round chain of

Section 5.3 gives

$$(P, \kappa(\mathrm{Ra}), Q, \kappa(\mathrm{Rw})) = (2, 2, 8, 4),\ (11, 4, 3, 2),\ (3, 2, 7, 4),\\ (9, 4, 3, 2),\ (1, 1, 7, 4),\ (12, 4, 2, 1).$$

Conditions (A), (D), (K1), and (K2) all hold with slack; the two-column input mask of variant B of Section 7.3 passes as well. Notably, that chain pays $P = 11$ and $P = 9$ at exactly the interface masks whose successor has $Q = 7$, which is the structure captured by the covering-number bound. Second, comparing the direct bitwise test of (A)/(D) against (K1)/(K2): of 60 318 pairs of interface masks, 7 528 are compatible, and the number violating (K1)/(K2) is zero. Third, the covering-number bound must not exceed the weight of any trail we constructed: the 5-round chain of Table 9, at its own endpoint width $|k_0| = 144$, faces the bound 62.2616, a slack of 22.25 bits; variant B at $|k_0| = 156$ has a slack of 19.84 bits.

It should be stated what this strengthening does not cover. κ uses only conditions (A) and (D), that is, only the covering relation between row sets; it still does not use the linear structure of (B)/(C), nor does it track how the columns of a multi-column interface mask are aligned with one another, and both the $P + Q \geq 17$ regime and the two ends of the trail are still handled by coarse relaxations. The figures in Table 8 are therefore provable lower bounds and not attainable values: against the weight 82.1012 actually constructed in Section 7.3, the bound 62.2616 at $U = 26$ is 19.84 bits low. Extending the exact mask-level analysis to multi-column interface masks is a natural approach to tightening the bound (Section 9).

5 An improved five-round linear distinguisher

The previous section showed that the provable lower bounds still fall short of the attack threshold. This gap motivates a direct search for improved distinguishers. The search below targets linear hulls rather than single trails and concludes with an exhaustive enumeration confirming that the resulting trail is optimal under any single-position modification.

5.1 Search method and results

The *distinguisher search* starts from a seed trail. The interface mask at one position is replaced by every compatible candidate in turn, each candidate is evaluated with the exact algorithm of Section 3, and every candidate whose weight lies within 6 bits of the seed weight is retained (at most 16 per position and pass). After each pass, if the best trail in the product of the candidate sets is lighter than the seed, it becomes the new expansion centre and a further pass is run. The search performs six passes with retention sizes $|S_2| = 51$, $|S_3| = 186$, $|S_4| = 191$, $|S_5| = 35$; the ELP is obtained by summing over the entire product of these candidate sets, and 283 663 exact one-round evaluations are performed in total.

The best trail found by this search for the reported external masks is given in Table 9 and drawn in Figure 2. Shading marks the active column at each interface mask, and the edge labels give the exact round weights. These weights sum to 10.4150+

Table 9 Best five-round trail found by the search for the reported external masks; it is locally optimal under every single-position modification.

Position	Column	z	wt	Exact weight of that round
a_1	3	`08010000`	2	10.4150
a_2	3	`cc312310`	11	28.8779
a_3	4	`04400800`	3	12.1296
a_4	5	`82660842`	9	28.2636
a_5	0	`00080000`	1	4.8301
a_6	0	`26f80b08`	12	—

Note. The variable z is the 32-bit value of the interface mask on the indicated column, written in hexadecimal.

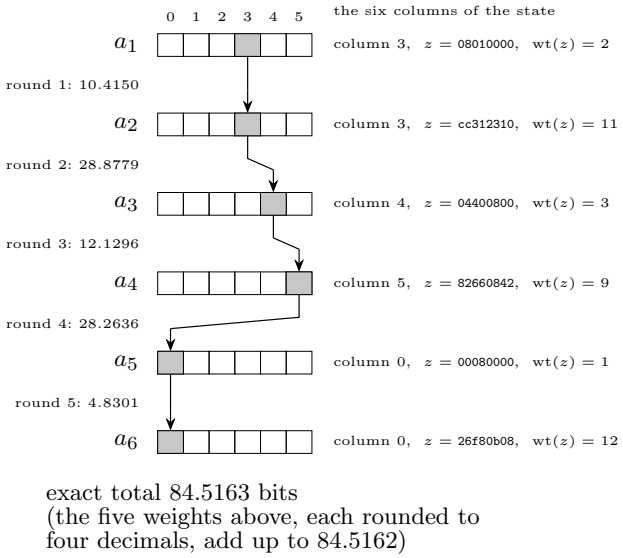


Fig. 2 A 5-round linear trail for SPEEDY.

$28.8779 + 12.1296 + 28.2636 + 4.8301 = 84.5162$. The evolution of the distinguisher from the published value to ours is summarised in Table 10.

5.2 Multi-column input masks at position 1

The published search considers positions 2 through 5. Extending the exhaustive enumeration to position 1 yields the following improved multi-column input mask:

$$a_1 = (\texttt{10000000},\ \texttt{00000000},\ \texttt{10000000},\ \texttt{00010000},\ \texttt{00000000},\ \texttt{10000000}),$$

Table 10 Evolution of the 5-round distinguisher.

Version	Trail correlation	Potential	Saving vs. published
Value reported in [20]	$2^{-93.0147}$	LP $= 2^{-186.0294}$	—
Exact value, same mask sequence (Sec. 3.2)	$2^{-90.0962}$	LP $= 2^{-180.1925}$	5.84 bits
Best single trail (reported external masks)	$2^{-84.5163}$	LP $= 2^{-169.0326}$	17.00 bits
Linear hull (reported external masks)	—	ELP $\geq 2^{-168.9034}$	17.13 bits
Best single trail (a_1 released)	$2^{-83.1943}$	LP $= 2^{-166.3887}$	19.64 bits
Linear hull (a_1 released)	—	ELP $\geq 2^{-166.3865}$	19.64 bits

Note. The second column reports correlations of single trails only. It is left blank on the two hull rows: under the independent round key assumption the ELP is an average of squared correlations over the round keys, so its square root ($2^{-84.4517}$ and $2^{-83.1933}$, respectively) is not the correlation of the hull for any fixed key. The data savings are differences of potential weights, the data complexity being inversely proportional to the potential. The ELP of the fourth row comes from the distinguisher search of this section, which retains $51 \cdot 186 \cdot 191 \cdot 35 = 63\,413\,910$ trail sequences; the ELP of the last row comes from the hull-gain control experiment of Section 5.2 (a single-pass scan, weight tolerance 14 bits, at most 250 candidates per position), which retains $3 \cdot 5 \cdot 33 \cdot 5 = 2\,475$. The two runs are therefore not of the same calibre: the bound on the last row is the more loosely searched of the two, so the 19.64 bits it reports is a conservative figure.

with $|\mathrm{Ra}(a_1)| = 2$. The weight of round 1 decreases from 10.4150 to 9.0931 and the total weight to $9.0931 + 28.8779 + 12.1296 + 28.2636 + 4.8301 = 83.1943$.

The modification above changes the input mask and therefore defines a *different* distinguisher; it is not merely a more accurate evaluation of the published one. For the published mask pair, the correct improvement figure is 17.13 bits; the 19.64 bits pertain to the modified mask.

How much does the linear hull itself contribute? The comparison has to be made between two quantities of the same kind, so we state it in potentials: the *potential* of a single trail is c_τ^2, the square of its correlation, and the ELP is the sum of the potentials of all trails in the hull. For this we run a separate *hull-gain control experiment* (a single-pass scan, weight tolerance 14 bits, at most 250 candidates per position) with identical parameters for the two chains: for the old a_1, the best single trail has potential $2^{-169.0326}$ (correlation $2^{-84.5163}$) and the ELP is at least $2^{-169.0304}$; for the new a_1, the best single trail has potential $2^{-166.3887}$ (correlation $2^{-83.1943}$) and the ELP is at least $2^{-166.3865}$. The hull thus contributes only 0.0022 bits of potential in both cases. In other words, in this region the 5-round linear hull of SPEEDY is dominated by a single trail, and the improvement comes almost entirely from better trails rather than from hull effects. This deserves to be stated explicitly, as it runs counter to experience with many SPN ciphers.

5.3 Exhaustive single-position searches

The observation of Section 2.3, that interface masks satisfying conditions (B) and (C) form a linear space, lets us enumerate each of the five positions exhaustively rather than heuristically; see Table 11.

Table 11 Exhaustive enumeration of the five positions (5-round distinguisher).

Position	Dimension	Leaves	Passing (A)+(D)	Baseline wt.	Result
1	90	417 691	71 442	9.0931	found[a]
2	48	376	257	37.9710	none
3	78	4 282	1 687	41.0075	none
4	96	9 461	3 769	40.3933	none
5	60	330	177	33.0937	none

Note. Dimension is the dimension of the corresponding linear space. The baseline weight of position t is the sum of the exact weights of the two rounds adjacent to that position in the reference trail (position 1 is adjacent to round 1 only); the reference trail here is 9.0931, 28.8779, 12.1296, 28.2636, 4.8301, i.e. the trail after the improvement of Section 5.2. The "Result" column states whether the enumeration contains a mask that strictly lowers that sum. [a] The improvement is described in Section 5.2.

We may therefore assert that the current trail is optimal under any single-position modification. The only remaining possibility is a joint modification of two or more interface masks, which is the main open problem of this paper (Section 9).

6 Reassessment of Wang et al.'s full-round SPEEDY-7-192 linear key-recovery attack

The attack of [20] covers full-round SPEEDY-7-192. Section 2.4 of [5] already identified an error in its key-recovery step and stated that "the actual complexity exceeds 2^{260}". We refine this observation by computing the affine dimension of the Walsh-spectrum support exactly and obtain $d = 324$. The scope of the resulting conclusion is limited: the unpruned fast Walsh transform costs $2^{323.53}$ equivalent encryptions ($2^{332.34}$ additions), and the affine pruning of [9] cannot reduce that, because the smallest affine subspace containing the spectrum is the whole space. Footnote 8 of [5] suggests that puncturing "may reduce the complexity slightly"; the full-dimension result itself says nothing about puncturing, so we give in Section 6.5 a closed form for the retained Walsh energy and use it to prove that, within the class of affine subspaces aligned with the 6-bit blocks, neither puncturing nor a union of subspaces reaches the 2^{192} claim. Accordingly, we verify the published counts and identify the misused step in the complexity formula (Section 6.1), compute the misused quantity exactly (Section 6.2), evaluate the resulting complexity and compare it with the literature (Section 6.3), evaluate an endpoint-compression variant (Section 6.4), and finally treat puncturing and unions of affine subspaces (Section 6.5).

6.1 Reported parameters and the complexity formula

Direct recomputation confirms the published key-recovery parameters: $|k_0| = 144$, $|k_7| = 180$, $l_{0,7} = 139$, 185 candidate bits, 7 remaining bits, and data $2^{188.51}$ (published: $2^{188.50}$). There is no error at the level of the counts; the discrepancy arises from how $l_{0,7}$ enters the complexity formula.

The verified counts isolate the discrepancy to the use of $l_{0,7}$ in the complexity formula. The published time complexity reads "With the FWT approach in [7], the time complexity is $2(\rho_M + \rho_A)2^{|k_0|+|k_7|-l_{0,7}}$", that is, $2^{|k_0|+|k_7|}$ divided by $2^{l_{0,7}}$. But $l_{0,7}$ describes the dependency between k_0 and k_7 induced by the linear key schedule, and what it reduces is the number L of independent candidate keys; the cost term of the fast Walsh transform is $L_{\text{int}} \cdot l \cdot 2^l$, where l is the dimension of the domain of the transform, equal to the length $|k_0| + |k_7| = 324$ of the concatenated plaintext and ciphertext parts, and this does not shrink because the keys are linearly related. Key dependencies can only reduce the number of candidates to be tested at the end; they cannot reduce the size of the transform itself.

6.2 Dimension of the Walsh-spectrum support $d = 324$

The argument of the previous subsection is qualitative. To turn it into a verifiable value we need the dimension d of the affine space spanned by the support of the Walsh spectrum that is transformed, because the cost of affine pruning [9] is exactly $d \cdot 2^d$. We fix the objects once and for all: the Boolean function used in the key-recovery step is a *Boolean* function $g : \mathbb{F}_2^n \to \mathbb{F}_2$ with $n = |k_0| + |k_7| = 324$, and what the transform is applied to is its sign function $f = (-1)^g : \mathbb{F}_2^n \to \{\pm 1\}$, whose spectrum

is $\hat{f}(u) = \sum_x f(x)(-1)^{\langle u,x\rangle}$. Enumerating 2^{324} spectral values is infeasible, so we use the identity below instead.

Lemma 8 (Walsh dimension identity) *Let $g : \mathbb{F}_2^n \to \mathbb{F}_2$, let $f = (-1)^g$, and let*

$$V = \{v \in \mathbb{F}_2^n : g(x \oplus v) = g(x) \text{ for all } x\}$$

be its group of invariant translations; equivalently $V = \{v : f(x \oplus v) = f(x) \text{ for all } x\}$. Then $\operatorname{span}(\operatorname{supp}\hat{f}) = V^\perp$, *and hence* $\dim \operatorname{span}(\operatorname{supp}\hat{f}) = n - \dim V$.

Proof Let $v \in V$ and substitute $y = x \oplus v$:

$$\hat{f}(u) = \sum_x f(x \oplus v)(-1)^{\langle u,x\rangle} = (-1)^{\langle u,v\rangle}\,\hat{f}(u),$$

so $\hat{f}(u) \neq 0$ forces $\langle u, v\rangle = 0$. Hence $\operatorname{supp}\hat{f} \subseteq V^\perp$ and $W := \operatorname{span}(\operatorname{supp}\hat{f}) \subseteq V^\perp$. Conversely, let $v \in W^\perp$. Inverse transformation gives

$$f(x \oplus v) = 2^{-n} \sum_{u \in \operatorname{supp}\hat{f}} \hat{f}(u)(-1)^{\langle u,x\rangle}(-1)^{\langle u,v\rangle} = f(x),$$

because every u occurring in the sum is orthogonal to v. Thus $W^\perp \subseteq V$, i.e. $V^\perp \subseteq W$, and the two spaces coincide. Taking dimensions and using $\dim V^\perp = n - \dim V$ finishes the proof. □

Lemma 8 gives the dimension of the *linear* span of the support, whereas the cost of affine pruning is governed by the dimension of its *affine* hull, and the two need not agree. The identity below supplies the latter and identifies the invariant that controls it.

Lemma 9 (Affine dimension identity) *In the notation of Lemma 8, let*

$$\mathrm{LS}(g) = \{v \in \mathbb{F}_2^n : g(x \oplus v) \oplus g(x) = \mathit{const}\}$$

be the space of linear structures of g, and let $D = \operatorname{span}\{u \oplus u' : u, u' \in \operatorname{supp}\hat{f}\}$ be the difference space of $\operatorname{supp}\hat{f}$. *Then*

$$D^\perp = \mathrm{LS}(g), \qquad d := \dim \operatorname{aff}(\operatorname{supp}\hat{f}) = n - \dim \mathrm{LS}(g).$$

Moreover $V \subseteq \mathrm{LS}(g)$ and $\dim V \le \dim \mathrm{LS}(g) \le \dim V + 1$, so d and $\dim \operatorname{span}(\operatorname{supp}\hat{f})$ *differ by at most one.*

Proof Parseval's relation $\sum_u \hat{f}(u)^2 = 2^{2n}$ gives $\operatorname{supp}\hat{f} \neq \emptyset$. Fix $u_0 \in \operatorname{supp}\hat{f}$; then $\operatorname{aff}(\operatorname{supp}\hat{f}) = u_0 \oplus D$, so $\dim \operatorname{aff}(\operatorname{supp}\hat{f}) = \dim D$. Let $v \in \mathbb{F}_2^n$ and $c \in \mathbb{F}_2$. The condition $g(x \oplus v) \oplus g(x) = c$ for all x is equivalent to $f(x \oplus v) = (-1)^c f(x)$; taking Walsh transforms of both sides and using $\widehat{f(\cdot \oplus v)}(u) = (-1)^{\langle u,v\rangle}\hat{f}(u)$, it is in turn equivalent to

$$(-1)^{\langle u,v\rangle}\,\hat{f}(u) = (-1)^c \hat{f}(u) \quad \text{for all } u,$$

that is, to $\langle u, v\rangle = c$ for every $u \in \operatorname{supp}\hat{f}$. Hence $v \in \mathrm{LS}(g)$ if and only if $\langle u, v\rangle$ is constant on $\operatorname{supp}\hat{f}$, if and only if $\langle u \oplus u', v\rangle = 0$ for all $u, u' \in \operatorname{supp}\hat{f}$, i.e. $v \in D^\perp$. Taking dimensions gives $\dim D = n - \dim \mathrm{LS}(g)$. Finally V is the part of $\mathrm{LS}(g)$ with $c = 0$, which either equals $\mathrm{LS}(g)$ or is a subgroup of index two in it, so $\dim V \le \dim \mathrm{LS}(g) \le \dim V + 1$. □

That one bit of slack is enough to change the conclusion: if $\operatorname{supp}\hat{f}$ consists of a single nonzero point u_0, the linear dimension is one while the affine dimension is zero, and indeed $g(x) = \langle u_0, x\rangle \oplus \epsilon$ is then affine and the transform can be carried out on a single coset. What has to be excluded is therefore $\mathrm{LS}(g) \neq \{0\}$, not merely $V \neq \{0\}$, and Theorem 5 is stated for $\mathrm{LS}(g)$. Two further points are worth noting. First, g is the sum of a plaintext-side and a ciphertext-side function in disjoint variables, $g = g_{\mathrm{p}} \oplus g_{\mathrm{c}}$, and each difference depends only on its own group of variables, so $\mathrm{LS}(g) = \mathrm{LS}(g_{\mathrm{p}}) \times \mathrm{LS}(g_{\mathrm{c}})$; this is what licenses treating the 144 and 180 sides separately and adding the dimensions. Second, the identity is stated for the sign function f: it is f that is transformed, while it is the Boolean g whose variables are eliminated iteratively through derivative constraints.

Theorem 5 *For the function g of the full-round linear attack of [20],*

$$d = 324 = |k_0| + |k_7| \text{ (exactly)}, \qquad \mathrm{LS}(g) = \{v : g(x \oplus v) \oplus g(x) = \mathit{const}\} = \{0\}.$$

Proof We determine the translation-invariance group by the following four-step elimination.

(i) Local Boolean functions. Let R be the set of active rows of the fixed far-side mask and j^* the column of a_1 (plaintext side) resp. a_6 (ciphertext side). Then

$$g(x) = \bigoplus_{r \in R} \varphi_r(x), \qquad \varphi_r(x) = \rho\big(b_r(x)\big), \quad b_r(x)_j = \big[S(x_{\mathrm{row}(r,j)})\big]_j, \quad \rho(\nu) = [S(\nu)]_{j^*},$$

with $\mathrm{row}(r, j) = r + j$ on the plaintext side and $r - j$ on the ciphertext side, and S the first-layer S-box (resp. S^{-1}). Each function φ_r therefore reads exactly the six consecutive text rows $\mathrm{row}(r, 0), \ldots, \mathrm{row}(r, 5)$, and the sum is over $\mathbb{F}_2$.

(ii) The derivative constraint. Put $\psi_r(v; x) = \varphi_r(x \oplus v) \oplus \varphi_r(x)$. Then $v \in V \iff \bigoplus_{r \in R} \psi_r(v; \cdot) \equiv 0$, while $v \in \mathrm{LS}(g)$ if and only if $\bigoplus_{r \in R} \psi_r(v; \cdot)$ is constant. Writing $b = b_r(x)$ and $\delta_j = [S(x_i)]_j \oplus [S(x_i \oplus v_i)]_j$ with $i = \mathrm{row}(r, j)$, we have $\psi_r = \rho(b \oplus \delta) \oplus \rho(b)$, so ψ_r depends on x_i only through the attainable pair set $P_j(v_i) = \{([S(x)]_j, [S(x)]_j \oplus [S(x \oplus v_i)]_j) : x \in \mathbb{F}_2^6\} \subseteq \{0, 1\}^2$. Since the rows read by a local function are distinct, its attainable set is the product $P_0 \times \cdots \times P_5$, and ψ_r depends on v only through that tuple; each column admits only a handful of distinct sets, so the condition is decided by a finite enumeration and is checked exhaustively rather than by sampling.

(iii) No linear structure. $P_j(0) = \{(0, 0), (1, 0)\}$ always. We swept all 63×63 pairs (v, c) for both S and S^{-1} and found no nonzero linear structure, i.e. $c \cdot (S(x) \oplus S(x \oplus v))$ is constant for no $v \neq 0$. Hence for $v_i \neq 0$ the set $P_j(v_i)$ is neither $P_j(0)$ nor $\{(0, 1), (1, 1)\}$, so "$P_j \neq P_j(0)$" is *equivalent* to "$v_i \neq 0$". This is the step that makes the iterative elimination possible, and it is exactly the condition that the identity-S-box comparison does not satisfy.

(iv) Iterative elimination. Call r *active* if $\psi_r \not\equiv 0$. If a text row i occurs in exactly one active function r, then varying x_i alone changes only ψ_r, so the constancy of $\bigoplus_r \psi_r$ forces ψ_r to be independent of x_i; by step (ii) this is a condition on the tuple $(P_0, \ldots, P_5)$ alone, and by step (iii) it is a condition on the zero pattern of that local function alone. Lemma 10 reduces everything this step can yield to a finite enumeration of 64 patterns: it forces $v_i = 0$ at least on the row carrying x_i, and on the ciphertext side it forces all six rows of that function to zero. Each newly forced row removes further active functions and exposes rows occurring in one remaining function, and the elimination is iterated to completion. At termination every row is forced to $v_i = 0$: on the plaintext side 8 local functions covering 24 rows close after six passes, five of which force new rows, whereas the 12 ciphertext functions covering 30 rows

close after three, two of which force new rows. The enumeration uses only the constancy of $\bigoplus_r \psi_r$ and never the value of the constant, so what it bounds is exactly $\mathrm{LS}(g)$: hence $\mathrm{LS}(g) = \{0\}$, a fortiori $V = \{0\}$, and $d = n = 324$ by Lemma 9. □

The strength of step (iv) must be stated exactly, since it is easy to overstate. Fix a local function r, write $i_j = \mathrm{row}(r, j)$ for the text row read by its j-th column, and for $Z \subseteq \{0, \ldots, 5\}$ let $\mathcal{T}(Z)$ be the tuple with $P_j = P_j(0)$ for $j \in Z$ and $P_j = \{0,1\}^2$ for $j \notin Z$. Call Z *flat at column* j if ψ_r evaluated on $\mathcal{T}(Z)$ does not depend on column j.

Lemma 10 (Zero-pattern exclusion) *For both S and S^{-1} and every column j, the following hold.*

1. *$P_j(0) = \{(0,0),(1,0)\}$, and $P_j(v_i) = \{0,1\}^2$ for every $v_i \neq 0$. Consequently the attainable tuple of a local function is determined by its zero pattern $Z = \{j : v_{i_j} = 0\}$ alone, and the only tuples that can occur are the $2^6 = 64$ tuples $\mathcal{T}(Z)$.*
2. *If $j \notin Z$ then Z is not flat at column j. An independence constraint at column j therefore forces $v_{i_j} = 0$, that is, it forces a single row to zero and not necessarily all six rows of the function.*
3. *Let Z_0 be the set of columns whose row is already known to satisfy $v_i = 0$, let I be the set of columns whose row occurs in exactly one active function, and let $\mathcal{S}(Z_0, I)$ be the set of those Z with $Z \supseteq Z_0$ that are flat at column j for every $j \in I$. Then the true zero pattern lies in $\mathcal{S}(Z_0, I)$, and hence $v_{i_j} = 0$ for every $j \in \bigcap_{Z \in \mathcal{S}(Z_0,I)} Z$.*
4. *For S^{-1} with $j^* = 0$ (the ciphertext side), no $Z \neq \{0, \ldots, 5\}$ is flat at any column, so a single row occurring in exactly one active function forces all six rows of that function to zero. For S with $j^* = 3$ (the plaintext side) there are exactly 25 pairs (Z, j) with $j \in Z$, $Z \neq \{0, \ldots, 5\}$ that are flat at column j, so in general only part of a function's rows is forced and the elimination must iterate.*

Proof (1) The first identity is immediate from the definition of $P_j(0)$. The second is step (iii): if some $v_i \neq 0$ had $P_j(v_i) \subsetneq \{0,1\}^2$, then $[S(x)]_j \oplus [S(x \oplus v_i)]_j$ would be constant in x, contradicting the absence of nonzero linear structures of S. (2) and (4) are finite enumerations: for the six choices of j^*, the six of j and the two S-boxes, that is 72 combinations, all 64 zero patterns are traversed. The instances to be checked in (2) are the pairs (Z, j) with $j \notin Z$, that is $12 \times 6 \times 32 = 2304$ of them, and there is no exception; the counts 25 and 0 in (4) come from the same enumeration. (3) If row i_j occurs in exactly one active function then, by the argument of step (iv), ψ_r must be independent of x_{i_j}, so the true zero pattern is flat at column j; it also contains Z_0, hence lies in $\mathcal{S}(Z_0, I)$ and therefore contains $\bigcap_{Z \in \mathcal{S}(Z_0,I)} Z$. □

By (2) we have $I \subseteq \bigcap_{Z \in \mathcal{S}(Z_0,I)} Z$, so (3) is at least as strong as (2); on the plaintext side the containment is strict, and it is (3) that each pass of the implementation performs. The first pass already shows the difference: the row-by-row rule (2) alone yields nine rows, whereas (3) yields fifteen.

Table 12 gives a finite certificate for the theorem: for each part and each elimination pass, it lists the remaining active functions and the individual text rows newly forced to $v_i = 0$. The certificate has two relevant properties. First, the per-column catalogue is exact: for both S and S^{-1} every column admits exactly *one* distinct pair set $P_j(v_i)$

Table 12 Derivative constraints certifying Theorem 5.

Pass	Active functions	Rows newly forced to $v_i = 0$	Total forced
Plaintext side: S, $j^* = 3$, 8 functions, 24 active rows, $n = \lvert k_0 \rvert = 144$			
1	8	$0, 1, 8, 10, 13, 20$–$25, 28$–31	15/24
2	6	$2, 3$	17/24
3	5	4	18/24
4	4	$5, 6, 7$	21/24
5	2	$9, 11, 12$	24/24
6	0	none	24/24 (complete)
Ciphertext side: S^{-1}, $j^* = 0$, 12 functions, 30 active rows, $n = \lvert k_7 \rvert = 180$			
1	12	0–$3, 6$–$11, 14$–$19, 24$–31	24/30
2	8	$4, 5, 20$–23	30/30
3	0	none	30/30 (complete)

Note. Each pass applies the deduction of Lemma 10(3) to every remaining local function, with I the set of its columns whose row occurs in exactly one remaining function. On both the plaintext and ciphertext parts, all active text rows are forced to $v_i = 0$; hence $\mathrm{LS}(g) = \{0\}$ and $d = 144 + 180 = 324$.

for $v_i \neq 0$, and it differs from $P_j(0)$, which is what step (iii) needs. Second, the two sides close at different speeds, and Lemma 10(4) says why: on the ciphertext side no pattern other than the all-zero one is flat at any column, so one row forces all six rows of its function and two productive passes suffice, whereas on the plaintext side the 25 flat cases let a pass peel off only part of a function's rows, and five are needed. The elimination is therefore not a formality: on the plaintext side the first pass forces only 15 of the 24 rows, and the remaining nine are forced a few at a time in passes 2 to 5.

We validate the full-rank conclusion by four complementary checks.

1. The Boolean function used in the key-recovery step agrees pointwise with an independent bitwise evaluation on 576 inputs.
2. Exact small-scale spectrum: for a subproblem restricted to three rows we computed the complete 2^{18} Walsh transform, whose dimension matches the certificate method.
3. Replacing the S-box by the identity permutation gives $\dim V = 143$ with none of the 24 rows forced, whereas three random 6-bit permutations force all 24 rows. These controls distinguish the SPEEDY result from the degenerate linear case.
4. At the borderline where the certificate criterion no longer applies, an exact 2^{24} computation on four rows gives the same result.

6.3 Complexity of unpruned FWT-based methods

With $l = 324$, $d = 324$ and $L_{\mathrm{int}} = 1$, Table 13 can be filled in. Note that $\log_2 324 = 8.3399$, so $324 \cdot 2^{324} = 2^{332.34}$ additions, which by the conversion constant $2^{8.807}$ of Section 4.3 is $2^{323.53}$ equivalent encryptions. The column "Cost here" of Table 13 follows the model of [11] and is stated in additions; dividing by $2^{8.807}$ gives equivalent encryptions.

Table 13 FGNP-family costs for the attack of [20].

Method	General cost	Cost here
Matsui's Alg. 2	$O(NL)$	$2^{373.5}$
Alg. 2 + distillation table	$O(N) + O(2^l L)$	2^{509}
Fast Walsh transform	$O(N) + O(L_{\text{int}} \cdot l \cdot 2^l)$	$324 \cdot 2^{324} = 2^{332.34}$
Affine pruning [9]	$O(N) + O(L_{\text{int}} \cdot d \cdot 2^d)$	$d = 324 = l$, so the dimension is not reduced: $2^{332.34}$
Puncturing (one subspace)	$C = N_0/\theta + L_{\text{int}} \cdot d \cdot 2^d$	Restricted to 6-bit-aligned blocks; the data ceiling forces $d \geq 282$: $\geq 2^{290.14}$ (Section 6.5)
Union of T subspaces [11]	$C = TN + \sum_i L_{\text{int},i} d_i 2^{d_i} + TL$	Minimum normalized cost $2^{208.78}$ at $d = 180$, $T = 2^{16.70}$ (Section 6.5)

Note. The parameters are $l = 324$, $d = 324$, and $L_{\text{int}} = 1$. Puncturing and union-of-subspaces costs use the model of Section 6.5, in which the data follow the law $N = N_0/\theta$ with $N_0 = 2^{188.51}$ (θ being the retained share of the Walsh energy); the N in the last row is computed by this law.

A natural question is whether the effective transform dimension $l = 324$ can be reduced. We evaluate three possible strategies.

1. Guess one side first and transform the other. Building the ciphertext-side distillation table for a fixed k_0 already costs $2^{144} \cdot 2^{180}$ operations; doing it once for every k_0 is precisely the plaintext-side transform itself. The total remains $(144 + 180) \cdot 2^{324}$.
2. Fold onto the subspace of legitimate keys. Since the key schedule is linear, the legitimate pairs (k_0, k_7) form a 185-dimensional linear subspace W. Convolution on W can be written as $\text{cor}|_W(k) = \sum_r (a|_{r+W} *_W f|_{r+W})(k)$, which requires ranging over 2^{139} cosets at a cost of $2^{139} \cdot 185 \cdot 2^{185} = 2^{331.5}$ additions, i.e. a saving of at most 0.8 bits. The spectral viewpoint explains why: this approach requires knowing $h = \hat{f} \cdot \hat{a}$ on a spanning set of $\text{supp}(\hat{f})$, and $d = 324$ says exactly that this spanning set is the whole space.
3. Exploit sparsity of the distillation table. Three objects have to be kept apart here: the size 2^{324} of the index space of the table, the number of its nonzero counters, and the number of output points the transform has to produce. Only the second is small: the data complexity is $2^{188.5} \ll 2^{324}$, so at most $2^{188.5}$ counters are nonzero and the table can be stored sparsely. The transform, however, has to deliver the correlation at every one of the 2^{324} candidate key values, and a sparse input does not reduce that output grid.

Among the three strategies examined, the lowest estimated cost is $2^{322.69}$ equivalent encryptions ($2^{331.5}$ additions), against $2^{323.53}$ equivalent encryptions ($2^{332.34}$ additions) for the unpruned fast Walsh transform; none of them approaches 2^{260}. We do not claim that $2^{322.69}$ is an infimum over all possible transform-based strategies, because the analysis covers only these three cases.

The relation to [5] can be read as follows. That paper writes "exceeds 2^{260}", i.e. gives a lower bound, and $2^{332.34} > 2^{260}$, so there is no contradiction. As lower bounds on the same quantity, the larger one is tighter: ours is tighter by 72.34 bits. The three

numbers differ in strength and should be distinguished. The value 2^{324} is unconditional and unit-free (the domain of the dense transform is $\mathbb{F}_2^{324}$, so its output grid has 2^{324} points); $2^{332.34}$ is the leading cost of the unpruned 324-dimensional fast Walsh transform, and by $d = l$ it is also what affine pruning [9] costs, since the spectrum admits no affine-dimension reduction; 2^{260} is an assertion for which no derivation is given. Conversely, if 2^{260} were read as an achievable cost for a pruned transform, then $O(d \cdot 2^d)$ would require $d \approx 252$, contradicting the exact value $d = 324$. Combined with the word "exceeds" in the original, the reading we find self-consistent is that [5] gives an unoptimised lower bound while we give the cost of the unpruned fast Walsh transform exactly.

The following distinctions clarify the units and scope of this comparison.

- Units. With the conversion constant $2^{8.807}$ of Section 4.3, $2^{332.34}$ additions correspond to about $2^{323.53}$ equivalent encryptions, still 63.53 bits above 2^{260}.
- Scope. Both bounds cover only the family of methods based on linear distillation and the fast Walsh transform. A looser bound is more robust to the modelling assumptions, which is the one respect in which 2^{260} is preferable to $2^{332.34}$.
- What is measured and what is proved. The statement that "the spectrum is dense" is a measurement: the figure 94.8% comes from a full 2^{24} transform on a four-row restricted spectrum on the ciphertext side, not from a global measurement over 2^{324}. It does not affect the value $d = 324$, and after Section 6.5 it no longer carries any conclusion either: the last two rows of Table 13 come from Theorem 7, whose only modelling assumption is the data law $N = N_0/\theta$, the one most favourable to the attacker.
- What $d = 324$ does and does not exclude. It excludes exactly one thing: restricting the transform to a single affine subspace, which is what [9] does, because the smallest affine subspace containing $\operatorname{supp}\hat{f}$ is the whole space. By *itself* it does not exclude a union of several subspaces [11], nor a *punctured* transform that discards a subset of the spectrum and accepts the resulting loss of correlation. These two strategies are analysed separately in Section 6.5, where neither reaches the 2^{192} claim within the class of affine subspaces aligned with the 6-bit blocks. What remains unexcluded is a subspace not aligned with the blocks, and a key-recovery arrangement that is not of the distillation-plus-transform structure at all. Automated searches for the best decomposition of a key-recovery attack [13] are precisely the line of work in which such an arrangement, if one exists, would be found; we make no claim about the outcome.

6.4 Evaluation of an endpoint-compression variant

An alternative is to use a distinguisher with smaller endpoint key masks, thereby reducing $|k_0| + |k_7|$ until the unpruned fast Walsh transform becomes feasible ($|k_0| + |k_7| \leq 192$). We evaluate every endpoint allocation on the ciphertext side that this budget admits. Although they satisfy the transform-time constraint, their data complexity exceeds the full codebook. The search is exhaustive only within the class defined below.

Fixing $a_1, \ldots, a_4$ to the single-column trail of Table 9 gives $|k_0| = 144$ on the plaintext side, so the ciphertext-side budget is $|k_7| \leq 48$, that is $D = |\mathrm{dil}_{\downarrow}(\mathrm{Ra}(a_6))| \leq$

8. For a single-column a_6 whose support has circular span t, $\mathrm{dil}_{\downarrow}$ covers $t+6$ rows, so $D \le 8$ is equivalent to a span of at most 2: each column contributes 32 single-row masks ($D=6$), 32 adjacent two-row masks ($D=7$), and 32 two-row plus 32 three-row masks of span 2 ($D=8$), giving 768 masks over the six columns. Enumerating these a_6 one by one, we filtered a_5 by the compatibility conditions (29 340 candidates per a_6), pre-sorted by the bound from the linear approximation table, and evaluated rounds 4 and 5 exactly (223 986 evaluations in total).

A single-row a_6 makes $w_5 = \mathsf{SC}^T\mathsf{MC}^T a_6$ a translated copy of the coefficient set TAPS, whose gaps (a lower bound on the covering circular interval) force z_5 to have several active rows; relaxing to two-row and three-row masks of span 2 only makes the support of w_5 heavier. The optimum is therefore still attained at $D=6$: optimal total weight 104.0628 bits (round 4 contributing 29.5053 and round 5 contributing 23.1349, with $\mathrm{wt}(z_5)=7$), $|k_0|=144$ and $|k_7|=36$, so $180 \le 192$ and the time is feasible, while the optima for $D=7$ and $D=8$ degrade to 114.7624 and 118.5151 bits. But at $D=6$ the data complexity is

$$N = 5.6/\mathrm{ELP} = 2^{210.61} \quad \text{(that is, 18.61 bits beyond the full codebook).}$$

The constant used here is already 5.6, the one most favourable to the attacker, and the data requirement still exceeds the full codebook. Thus, this variant does not yield a valid attack in the searched class. In this class, the first four masks $a_1, \dots, a_4$ were held fixed at the single-column trail of Table 9, and only the last two were varied, over all 768 single-column a_6 with $D \le 8$ and, for each of them, all 29 340 compatible a_5. We have not searched distinguishers with different masks $a_1, \dots, a_4$ or endpoint compression on the plaintext side. Consequently, this is a negative experimental result for the specified class, not an impossibility result for the complete framework.

6.5 Spectral puncturing and unions of affine subspaces

Theorem 5 establishes that the smallest affine subspace containing $\mathrm{supp}\,\hat{f}$ is the whole space, so the pruning of [9] cannot reduce the dimension. It does not answer the two remaining questions, and neither of them is about whether the dimension is large enough; both are about how much data it costs to compute less of the spectrum. The first is the union of T subspaces of [11], the second is *puncturing*: deliberately discarding part of the spectrum, trading correlation for dimension. This subsection computes that trade-off exactly in terms of the retained *spectral energy*. The support size and the largest spectral coefficient alone give insufficient lower bounds, as quantified at the end of the subsection.

Write $f = (-1)^g$ and $c(u) = 2^{-n}\hat{f}(u)$, so that Parseval's identity reads $\sum_u c(u)^2 = 1$. The $n = 324$ variables split by text row into 54 blocks of 6 bits (24 rows on the plaintext side, 30 on the ciphertext side); let u_i denote the block of u on row i. For a set of rows A and a fixed t whose nonzero blocks lie outside A, put

$$U_{A,t} = \{u \in \mathbb{F}_2^{324} : u_i = t_i \text{ for all } i \notin A\}, \qquad \dim U_{A,t} = 6|A|,$$

which we call an affine subspace *aligned* with the 6-bit blocks; for $t = 0$ it is the coordinate subspace U_A. The retained fraction of the energy is

$$\theta(A,t) = \sum_{u \in U_{A,t}} c(u)^2 \in [0,1], \qquad \theta(\mathbb{F}_2^{324}) = 1.$$

Theorem 6 (Closed form for the retained energy) *Keep the notation of the proof of Theorem 5: the set R indexes all $8+12=20$ local Boolean functions, and $\mathrm{row}(r,j)$ is the text row read by function r in column j. The column j^* and the S-box used (S on the plaintext side, S^{-1} on the ciphertext side) are taken according to the side r belongs to; the text rows of the two sides are disjoint, so the two sides can be merged into a single product, and we write c_S for the S-box correlation of either side. If every nonzero block of t sits on a text row read by a* single *local function, then*

$$\theta(A,t) = \prod_{r \in R} \Big(\prod_{j \in Z_r} c_S\big(t_{\mathrm{row}(r,j)}, e_j\big)^2 \Big) \cdot \sigma_{J_r, Z_r}, \qquad \sigma_{J,Z} = \sum_{Z \subseteq \mathrm{supp}(m) \subseteq J \cup Z} c_S(m, e_{j^*})^2,$$

where e_j is the j-th unit mask,

$$J_r = \{j : \mathrm{row}(r,j) \in A\}, \qquad Z_r = \{j : \mathrm{row}(r,j) \notin A,\ t_{\mathrm{row}(r,j)} \neq 0\}.$$

In particular, for a coordinate subspace ($t = 0$, so $Z_r = \emptyset$),

$$\theta(A,0) = \prod_{r \in R} \sigma(J_r), \qquad \sigma(J) = \sum_{m \neq 0,\ \mathrm{supp}(m) \subseteq J} c_S(m, e_{j^*})^2, \qquad \sigma(\{0,\dots,5\}) = 1.$$

Proof Expand the sign of each local Boolean function by the Walsh transform:

$$(-1)^{\rho(b_r(x))} = \sum_{m_r \in \mathbb{F}_2^6} c_S(m_r, e_{j^*})\, (-1)^{\langle m_r, b_r(x) \rangle}.$$

Since $\langle m_r, b_r(x) \rangle = \bigoplus_j m_{r,j} \big[S(x_{\mathrm{row}(r,j)})\big]_j$, collecting their contributions by text row yields one mask $w_i(m)$ per row, whose j-th bit equals $m_{r,j}$ for the function r with $\mathrm{row}(r,j) = i$ (and is 0 if there is none). For a fixed pair (i,j) there is at most one such r, so $m \mapsto \big(w_i(m)\big)_i$ is a bitwise rearrangement, and in particular injective. Hence

$$c(u) = \sum_m \prod_{r \in R} c_S(m_r, e_{j^*}) \prod_i c_S\big(u_i, w_i(m)\big).$$

Square this and sum over $u \in U_{A,t}$, examining the coupling factor of (m, m') row by row. For $i \in A$, u_i ranges over $\mathbb{F}_2^6$ and the orthogonality of the linear approximation table gives $\sum_{u_i} c_S(u_i, v) c_S(u_i, v') = \delta_{v,v'}$, forcing $w_i(m) = w_i(m')$. For $i \notin A$, $u_i = t_i$ is fixed and the factor is $c_S(t_i, w_i(m))\, c_S(t_i, w_i(m'))$: if $t_i = 0$, invertibility of S gives $c_S(0,v) = \delta_{v,0}$, forcing $w_i(m) = w_i(m') = 0$; if $t_i \neq 0$, the same invertibility gives $c_S(t_i, 0) = 0$, and since that row occurs in a single function one has $w_i(m) \in \{0, e_j\}$, so $w_i(m) = w_i(m') = e_j$ is forced and a factor $c_S(t_i, e_j)^2$ is left behind. All three cases pin the corresponding bits of m and m' to each other, and injectivity gives $m = m'$: the cross terms vanish. In the remaining single sum, m_r ranges exactly over $Z_r \subseteq \mathrm{supp}(m_r) \subseteq J_r \cup Z_r$, and factorising by local function gives the stated product. Finally, $c_S(0, e_{j^*}) = 0$ because S is invertible, so $\sigma(\{0,\dots,5\}) = \sum_m c_S(m, e_{j^*})^2 = 1$ is Parseval's identity. □

Table 14 Dimension–energy frontier for affine subspaces aligned with the 6-bit blocks.

$\lvert A\rvert$	$d = 6\lvert A\rvert$	$\log_2 \theta^*$ (U_A)	$\log_2 \theta^*$ (affine)	$\log_2 T$	$\log_2$ add.	$\log_2$ enc.
6	36	$-\infty$	−89.9536	86.46	278.47	269.67
12	72	−61.3822	−61.0912	57.60	249.61	240.81
18	108	−44.0491	−44.0491	40.56	232.57	223.76
24	144	−30.6868	−30.6868	27.20	219.21	210.40
30	180	−20.1929	−20.1929	16.70	208.78	**199.97**
36	216	−12.2429	−12.2429	8.75	232.51	223.70
42	252	−6.4559	−6.4559	2.97	262.94	254.14
47	282	−3.0043	−3.0043	0	290.14	281.33
54	324	0	0	0	332.34	323.53

Note. The quantity θ^* is the largest retained energy fraction in the given dimension, and $T = 2^{-3.49}/\theta^*$ is the number of subspaces required to reach the full codebook. The last two columns give total time in additions and encryptions under the model of [11], converting $2^{8.807}$ additions into one encryption; the minimum occurs at $\lvert A\rvert = 30$.

Theorem 6 turns the energy-versus-dimension trade-off into a finite combinatorial optimisation: θ depends only on which of three states each row is in: free (in A, contributing 6 bits of dimension), frozen at 0, or frozen at some nonzero value; and the weight of each local function depends only on the states of the six rows it reads. Since each function reads six consecutive rows, the optimal frontier

$$\theta^*(s) = \max\bigl\{\theta(A,t) : |A| = s\bigr\}$$

can be computed exactly by a transfer matrix cycling through the 32 rows: the state need only remember the last five rows (3^5 possibilities), and 3^5 end configurations are enumerated to close the cycle. The resulting frontier is given in Table 14. Freezing rows at nonzero values increases the retained energy only for $|A| \le 12$ (a local function with no free row contributes 0 on a coordinate subspace, but at most $\prod_j \gamma_j \cdot c_S(\mathbf{1}, e_{j^*})^2$ on an affine one, where $\mathbf{1}$ is the all-one mask and $\gamma_j = \max_{a\neq 0} c_S(a, e_j)^2 \le (\mathrm{Lin}(S)/64)^2 = 2^{-2.830}$), and the cost at $|A| \le 12$ is already dominated by T; for $|A| \ge 18$ the two frontiers coincide.

Checking this closed form requires being able to evaluate a single spectral value $c(u)$ exactly, which the dependency graph of g provides: joining local functions that share a text row splits the 8 plaintext functions into $7 + 1$ and the 12 ciphertext functions into $8 + 4$. The four parts read disjoint sets of text rows, and therefore $\hat{f}$ is a tensor product of four factors, $144 + 180 = 108 + 36 + 96 + 84 = 324$. This is Theorem 4 instantiated on this particular function. Each part is evaluated exactly by one contraction along its rows, with a peak intermediate state of 2^{20}. We validate the closed form and the frontier in four ways: (i) on the real 36-bit and 84-bit parts, for random A with $|A| = 2$ and random t, all 2^{12} spectral values were computed pointwise and squared-summed, and agree with the closed form exactly (including nonzero cases such as 2.789497×10^{-5}); (ii) the transfer-matrix recurrence agrees

exactly with exhaustive enumeration of all 3^l state strings on 12 small cyclic instances (once for each of the two families); (iii) on the real instances, the exhaustive search over $|A| \leq 3$ agrees with the recurrence; (iv) among 200 000 random state strings per side, none has θ exceeding the frontier. Exact spectral values were additionally checked by a full Walsh transform on small-ℓ instances, direct enumeration over m, and the independent exact correlation computation of Section 3 on the full 192-bit round.

Converting the frontier into a cost requires one more rule, linking θ to the data complexity. We adopt the form most favourable to the attacker: when a fraction θ of the energy is retained, the data complexity is

$$N = N_0/\theta, \qquad N_0 = 2^{188.51},$$

i.e. we assume the retained energy can be exploited fully coherently (the best case for multiple or multidimensional linear cryptanalysis). The resulting cost bound therefore holds throughout the defined class of affine subspaces aligned with the 6-bit blocks. Since $N \leq 2^{192}$ (the full codebook), it follows immediately that

$$\theta \geq 2^{188.51-192} = 2^{-3.49}.$$

This constraint is severe: the data complexity of the attack is only 3.49 bits below the full codebook to begin with, so *almost no energy can be discarded.*

Theorem 7 *Consider the class of affine subspaces aligned with the* 6*-bit blocks, with nonzero frozen values only on rows occurring in a single local function. Let the attacker use* T *subspaces* $U_1, \ldots, U_T$ *of that class, of dimensions* $d_1, \ldots, d_T$*. Following the terms in the model of [11], define the normalized operation count*

$$C(T, \{d_i\}, N, L) = TN + \sum_i d_i 2^{d_i} + TL,$$

where $L = 2^{185}$ *is the number of surviving key candidates and* $2^{8.807}$ *normalized operations count as one* 7*-round encryption (that is,* 448 *S-box lookups). Then, under this normalized cost model:*

(a) For $T = 1$*, i.e. the pruning of [9], necessarily* $d \geq 282$*, and* C *is at least* $2^{290.14}$ *operations, i.e.* $2^{281.33}$ *encryptions,* 89.33 *bits above the* 2^{192} *claim. Compared with the unpruned* $2^{332.34}$*, pruning saves at most* 42.20 *bits anywhere in the class.*

(b) For any T *and any combination of dimensions,* C *is at least* $2^{208.78}$ *operations, i.e.* $2^{199.97}$ *encryptions,* 7.97 *bits above the claim. The cheapest point is* $|A| = 30$*,* $d = 180$*,* $\theta^* = 2^{-20.1929}$*,* $T = 2^{16.70}$*,* $N = 2^{192}$*, with binding term* $TN = 2^{208.70}$*.*

Proof Let U_i retain energy $\theta_i \leq \theta^*(d_i/6)$ and put $\Theta = \sum_i \theta_i$. The energy of the union is at most the sum of the parts, so $N \geq N_0/\Theta$; combined with $N \leq 2^{192}$ this gives $\Theta \geq 2^{-3.49}$. The total cost is $\sum_i \big(d_i 2^{d_i} + N + L\big)$, which we bound term by term:

$$\sum_i \big(d_i 2^{d_i} + N + L\big) \geq \sum_i \theta_i \cdot \frac{d_i 2^{d_i} + N + L}{\theta^*(d_i/6)} \geq \Theta \cdot \min_s \frac{6s \cdot 2^{6s} + N + L}{\theta^*(s)}.$$

Substituting $N = N_0/\Theta$, the right-hand side equals $\min_s\big(\Theta\,(6s\cdot 2^{6s}+L)+N_0\big)/\theta^*(s)$, which is nondecreasing in Θ and hence minimal at $\Theta = 2^{-3.49}$. Mixing different dimensions therefore cannot beat a single dimension, and it suffices to minimise over the rows of Table 14; the minimum is at $|A| = 30$, which is (b). Part (a) is the case $T = 1$: $\theta^*(s) \geq 2^{-3.49}$ gives $s \geq 47$ from the frontier, and the cost is dominated by $d \cdot 2^d$. □

The binding term in Theorem 7(b) is TN, the cost of evaluating all T subspaces on the data. If that term were waived altogether (which is not legitimate, since the T subspaces span all 324 bits and cannot share a single distillation table), the remaining $T\,d\,2^d + TL$ is minimized at $2^{203.41}$ normalized operations, i.e. $2^{194.60}$ encryptions, at $|A| = 29$, $d = 174$, $\theta^* = 2^{-21.7779}$, $T = 2^{18.29}$, still 2.60 bits above the 2^{192} claim. Even an accounting that charges nothing for reading the data therefore fails to reach the security claim.

Two simpler bounds are insufficient. First, the covering bound $\sum_i 2^{d_i} \geq |\mathrm{supp}\hat{f}|$ is weak because the spectral support of g is in fact *thin*: the measured fractions of random masks lying in the support of the four parts are $0/512$, $2^{-19.806}$ (this part has a single local function, so it can be counted exactly: $|\mathrm{supp}| = 74\,970 = 2^{16.194}$), $2^{-2.093}$ and $2^{-4.541}$; a thinner support decreases the corresponding union-of-subspaces lower bound. Second, deriving $|\mathrm{supp}\hat{f}| \geq c_{\max}^{-2}$ from Parseval requires an upper bound on $c_{\max}$, but g has precisely the property of admitting very sparse high-correlation approximations (masks placed on the few rows read by many local functions). The product of the best single trails of the four parts is $2^{-55.6176}$, which yields only $|\mathrm{supp}\hat{f}| \geq 2^{111.24}$, far too weak to support any conclusion. Theorem 6, by contrast, is exact and indifferent to how thin the support is.

What remains uncovered is the general affine subspace *not* aligned with the 6-bit blocks, and the freezing of a nonzero value on a text row read by two or more local functions (15 of the 24 rows on the plaintext side, 19 of the 30 on the ciphertext side). Both fail at the same place: the cross terms in the proof of Theorem 6 no longer vanish, θ becomes a double sum over (m, m'), and the transfer-matrix state grows from 2^{20} to 2^{40}, which we did not compute. The conclusion of this subsection is therefore about the class of subspaces aligned with the 6-bit blocks, not an impossibility result for every transform-based key recovery.

7 Linear cryptanalysis of SPEEDY-6-192

The complexity in the previous section is dominated by the ciphertext-side outer-key mask k_7, which increases the affine dimension of the Walsh-spectrum support to 324. In SPEEDY-6-192, the absence of MC from the last round eliminates this ciphertext-side term. We use this property to construct a 6-round linear key-recovery attack and then derive lower bounds for the corresponding dominant-trail FGNP/Walsh class.

Sections 7.1–7.3 construct a valid 6-round linear key recovery with $2^{169.84}$ known plaintexts, time $2^{170.20}$ and memory 2^{156}, all of whose correlations come from the exact algorithm of Section 3. Section 7.4 then formalises the class this attack belongs to as Definition 2 and, using the universal lower bound of Section 4.4, proves under the dominant-trail estimate that the class contains no attack inside the design security claim of SPEEDY-6-192 (data and time both $\leq 2^{128}$), and gives a normalized time

lower bound of $2^{136.302}$. Section 7.5 compares the same class with the time and memory complexities of the existing 6-round differential attack and excludes that complexity range as well.

7.1 Last-round structure and maximum correlation

The last round of SPEEDY omits MC, so for an r-round version $R_{r-1} = \mathsf{Ak}_r \circ \mathsf{SB} \circ \mathsf{SC} \circ \mathsf{SB} \circ \mathsf{Ak}_{r-1}$ contains no inter-column diffusion. This has two immediate consequences.

1. $|k_6| = 0$: no key bits need to be guessed on the ciphertext side, because the output mask of the distinguisher can be placed directly on the ciphertext (for $r = 6$, k_6 is the outer key of the last round). The dominant $l = |k_0| + |k_7| = 324$ of Section 6 (the case $r = 7$) is reduced here to $l = |k_0|$.
2. The weight of round 5 decreases from 4.8301 to 3.4150, one layer of MC^T diffusion being absent.

Together, these two make a 6-round linear key recovery feasible. The scope must be emphasised: the feature relies on the absence of MC in the last round, so it applies to SPEEDY-6-192 itself and not to SPEEDY-7-192 truncated to six rounds, because round 6 of the latter still contains MC. This is why, in Section 1, we said that our result and the 6-round attack of [21] are not interchangeable.

The same last-round structure also permits an exact maximisation of its correlation. Since the last round has no MixColumns, its correlation admits a closed analytic expression whenever the input mask has weight one, and the exact per-column optima are

$$(3.4150,\ 2.8301,\ 3.0931,\ 2.8301,\ 3.4150,\ 3.3561) \quad \text{(bits, per column)},$$

each entry having been checked individually against exact evaluation. We verified two further points: restricting single-column masks to weight at most 2 preserves optimality because the optimum already lies in that set; and among all 29 340 compatible masks a_5, only 60 satisfy all four compatibility conditions, of which only one has weight one.

7.2 Exhaustive single-position searches in the six-round setting

As in Section 5.3, we enumerate the five positions of the 6-round setting exhaustively; see Table 15. The masks $a_2, \ldots, a_5$ are already optimal at any cost and any number of columns.

Classifying the 152 improvements at position 1 by $|k_0|$, the case $|k_0| = 156$ gives round weight 9.4150 and yields variant B below. The cases $|k_0| = 174$ and $|k_0| = 192$, with round weights 9.4150 and 9.0931, respectively, do not yield a key-recovery attack below exhaustive search.

We also verified a structural fact: the cost distribution of multi-column interface masks is cost $= 9 \mapsto 480$, $10 \mapsto 640$, $11 \mapsto 480$, $12 \mapsto 1216$, $13 \mapsto 2624$, $14 \mapsto 24\,256$, and cost 8 is unreachable because $|\mathrm{TAPS}| = 7$ is not a multiple of any element of $32/\gcd(d, 32) \in \{8, 16, 32\}$.

Extending the catalogue of interface masks from $B \leq 14$ to $B \leq 16$ (1 518 384 single-column masks) yields no improvement for $a_2, \ldots, a_5$ (the baseline weights

Table 15 Exhaustive enumeration in the 6-round setting.

Position	Dimension	Leaves	Baseline weight	Improvements
1	90	417 691	10.4150	152
2	48	733	39.2929	0
3	78	4 282	41.0075	0
4	96	9 461	40.3933	0
5	36	142	31.6787	0

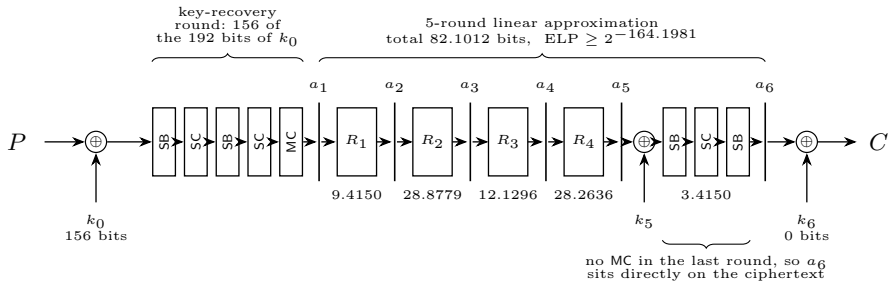
Note. The baseline weight is defined as in Table 11; the reference trail here is 10.4150, 28.8779, 12.1296, 28.2636, 3.4150 (the last round without MC, position 1 not yet improved). The number of improvements counts the masks that strictly lower the sum of the two adjacent round weights: the best of the 152 improvements at position 1 is 9.0931, while Section 7.3 uses the value 9.4150 among them because of the key-guessing width.

39.2929, 41.0075, 40.3933, 31.6787 remain unchanged). Thus the 0.4150-bit improvement comes from including position 1.

7.3 Key-recovery variants and complexities

Applying the FGNP framework to the best trail above yields two variants, differing in whether the input mask is single- or multi-column. Both variants take their three middle interface masks directly from the improved trail of Section 5 (a_2, a_3, a_4, 69.2711 bits in total), so the 19.64-bit data saving that Section 5 achieves over the published value is realised here rather than in a 7-round attack. The last-round weight decreases from 4.8301 to 3.4150 because MC is absent, and this 1.4151-bit reduction exceeds the 0.3219 bits lost on the input mask, so the 6-round setting reaches ELP $\geq 2^{-164.1981}$, which is 2.19 bits better than the 5-round value $2^{-166.3865}$ of Section 5. The lowest-weight input mask of Section 5 (round 1 at 9.0931 bits) is not applicable here: it forces $|k_0| = 192$ and therefore gives no improvement over exhaustive key search. Consequently, Section 7.2 uses the second-best mask at position 1.

- Variant A (single-column input mask). Taking $a_1 =$ `08010000` on column 0 gives $w_1 = 10.0000$ and total weight $10.0000 + 28.8779 + 12.1296 + 28.2636 + 3.4150 = 82.6862$, with linear hull ELP $\geq 2^{-165.3681}$ (the hull contributing only 0.0043 bits); $|k_0| = 144$ and $|k_6| = 0$; the fast Walsh transform costs $144 \cdot 2^{144} = 2^{151.17}$ additions, that is $2^{142.36}$ equivalent encryptions, with memory 2^{144}; taking advantage $a = 23$, the data is $N = 2^{170.96}$, the final exhaustive step $2^{192-23} = 2^{169}$, and the total time about $2^{171.29}$.
- Variant B (multi-column input mask; the result of this section). Taking $a_1 =$ `10020000` on both column 1 and column 2 gives $w_1 = 9.4150$ and total weight $9.4150 + 28.8779 + 12.1296 + 28.2636 + 3.4150 = 82.1012$, with linear hull ELP $\geq 2^{-164.1981}$ (the hull again contributing only 0.0043 bits); $|k_0| = 156$ and $|k_6| = 0$; the fast Walsh transform costs $156 \cdot 2^{156} = 2^{163.29}$ additions, that is $2^{154.48}$ equivalent encryptions, with memory 2^{156}; taking advantage $a = 24$, the data is $N = 2^{169.84}$



fast Walsh transform over $l = |k_0| = 156$ bits: $156 \cdot 2^{156} = 2^{163.29}$ additions, $= 2^{154.48}$ equivalent encryptions; memory 2^{156}

advantage $a = 24$: data $N = 2^{169.84}$ known plaintexts, final step $2^{192-24} = 2^{168}$, total $\approx 2^{170.20}$

Fig. 3 A 6-round known-plaintext attack on SPEEDY-6-192.

known plaintexts, the final exhaustive step $2^{192-24} = 2^{168}$, and the total time $\approx 2^{169.84} + 2^{154.48} + 2^{168} \approx 2^{170.20}$.

Variant B is better than variant A in time ($2^{170.20}$ against $2^{171.29}$) and is taken as the result of this subsection: a linear key recovery on SPEEDY-6-192 with $2^{169.84}$ known plaintexts, time about $2^{170.20}$ and memory 2^{156}. Variant A is retained here because it is the best choice available under a single-column input mask and because its memory is lower (2^{144} against 2^{156}): the two variants together give two endpoints of the memory-time trade-off, which shows that the order of 2^{170} persists across the enumerated mask space.

The attack is drawn in Figure 3 and follows the standard steps of the FGNP framework. The values below the rounds are exact weights; omitted key additions and round constants affect only the sign of the correlation. The final exhaustive search is stated separately to distinguish the two counting stages.

1. Data collection: collect $N = 2^{169.84}$ known plaintext-ciphertext pairs.
2. Distillation: compress each plaintext onto the 156 bits involved in k_0 and accumulate into a counter table of size 2^{156}. This table determines the dominant memory cost.
3. Transform and filtering: apply one 156-dimensional fast Walsh transform to that table ($156 \cdot 2^{156} = 2^{163.29}$ additions, that is $2^{154.48}$ equivalent encryptions) and keep the $2^{|k_0|-a} = 2^{156-24} = 2^{132}$ values of the 156-bit outer key with the largest absolute correlation.
4. Final exhaustion: for each of those 2^{132} outer candidates, complete the remaining $192-156 = 36$ master-key bits exhaustively and test the resulting key. The number of full keys tested is $2^{132+36} = 2^{168} = 2^{192-a}$, as it must be.

Two accounting details affect the complexity. First, the size of the final exhaustive step is 2^{192-a}, not $2^{192-|k_0|}$: the advantage a determines the reduction in the number of surviving key candidates. Replacing it by $|k_0|$ would repeat the complexity-accounting error analysed in Section 6. Second, the term $2^{169.84}$ in the total consists of N memory accesses rather than encryptions, so $2^{170.20}$ is a conservative time estimate.

7.4 Lower bounds for the dominant-trail attack class

Section 7.3 gives a concrete attack whose time $2^{170.20}$ lies far above the 2^{128} claim for SPEEDY-6-192. We next bound the best attack of this kind under the same dominant-trail data estimate. This subsection defines that model and derives the corresponding lower bounds.

Definition 2 A linear key-recovery attack against SPEEDY-6-192 belongs to *class* $\mathcal{A}$ if it consists of the following three parts: (i) a selected dominant 5-round linear trail covering rounds 1 to 5, of correlation weight w, whose output mask is placed directly on the ciphertext (Section 7.1, hence $|k_6| = 0$); (ii) the FGNP/Walsh key recovery of Section 2.4, using a distillation table of size $2^{|k_0|}$ and applying the fast Walsh transform over the $|k_0| = 6U$ outer-key bits involved in round 0; and (iii) the dominant-trail form of Selçuk's statistical estimate, with advantage a and success probability $P_S \geq 1/2$. The data estimate, memory, and normalized time are

$$N = c(a) \cdot 2^{2w}, \qquad M = 2^{|k_0|}, \qquad T = \max\big(N,\ |k_0| \cdot 2^{|k_0|}/2^{8.807},\ 2^{192-a}\big),$$

where $c(a) = \big(\Phi^{-1}(P_S) + \Phi^{-1}(1 - 2^{-(a+1)})\big)^2$. The three time terms are all in equivalent encryptions, the transform being converted by the constant $2^{8.807}$ of Section 4.3.

Both variants of Section 7.3 belong to class $\mathcal{A}$, as does the attack in Section 5.2 of [20] when transposed to six rounds.

Theorem 8 *Under the dominant-trail estimate and the cost model of Definition 2, class* $\mathcal{A}$ *contains no attack with estimated data* $\leq 2^{128}$ *and normalized time* $\leq 2^{128}$ *simultaneously. More precisely, any attack in the class whose normalized time does not exceed* 2^{128} *has estimated data complexity at least* $2^{130.91}$.

Proof Suppose an attack has time $T \leq 2^{128}$. From $T \geq 2^{192-a}$ we get $a \geq 64$. Since $c(a)$ is increasing in a, we have $c(a) \geq c(64)$; at $P_S = 0.95$, $\log_2 c(64) = 6.8660$, and $c(64)$ is increasing in P_S as well, so $\log_2 c(64) \geq 6.3892$ for every $P_S \geq 1/2$. On the other hand, the universal lower bound of Table 8 holds for arbitrary internal interface masks and for every endpoint width, and its minimum over all U is $w \geq 62.2616$. Hence

$$\log_2 N = \log_2 c(a) + 2w \ \geq\ 6.3892 + 2 \cdot 62.2616 = 130.9124 \ >\ 128,$$

contradicting $N \leq 2^{128}$. □

The proof uses only two facts: the final exhaustive step forces $a \geq 64$, and the universal 5-round trail bound. It needs no accounting for $|k_0|$ whatsoever, and is therefore immune to objections of the form "what is the true constant of the fast Walsh transform". If the distillation table is charged to the time as well, then $2^{|k_0|} \leq 2^{128}$ forces $U \leq 21$ and the universal bound rises to 65.0917; charging in addition the linear factor of the transform, $|k_0| \cdot 2^{|k_0|}/2^{8.807} \leq 2^{128}$, that is $|k_0| + \log_2 |k_0| \leq 136.807$, still gives $U \leq 21$ (at $U = 22$ that term is $2^{130.24}$), so the two accounting conventions yield the same universal bound. Table 16 lists the resulting data lower bounds for two

Table 16 Margins of Theorem 8.

P_S	$\log_2 c(64)$	No table charge $w \geq 62.2616$	Table charged ($U \leq 21$) $w \geq 65.0917$
0.95	6.8660	$2^{131.389}$ (+3.39)	$2^{137.049}$ (+9.05)
0.80	6.6430	$2^{131.166}$ (+3.17)	$2^{136.826}$ (+8.83)
0.50	6.3892	$2^{130.913}$ (+2.91)	$2^{136.573}$ (+8.57)

Note. Each entry is the minimum data complexity of an attack in class $\mathcal{A}$ with time at most 2^{128}; parentheses give the margin over 2^{128} in bits. The final two columns respectively omit and charge the distillation table; in equivalent encryptions, charging the table size $2^{|k_0|}$ and charging $|k_0|2^{|k_0|}/2^{8.807}$ impose the same constraint $U \leq 21$, so they coincide in a single column.

accounting conventions and three success probabilities; all six entries exceed 2^{128}, with margins from 2.91 to 9.05 bits.

Removing the 2^{128} ceiling on the time, the same dominant-trail accounting gives a normalized time lower bound for class $\mathcal{A}$.

Corollary 2 *Under the model of Definition 2, no attack in class $\mathcal{A}$ has normalized time below $2^{136.302}$, which is 8.30 bits above the 2^{128} claim for SPEEDY-6-192. The lower-bound expression is minimized at $|k_0| = 138$ ($U = 23$), with trail-weight lower bound $w \geq 62.2616$ and advantage $a = 56$; the resulting time lower bound is $2^{136.302}$, in which the fast Walsh transform term is $2^{136.302}$, the final exhaustive-search term is 2^{136} and the data term is $2^{131.219}$.*

Proof For each U, substitute the lower bound $f(U)$ of Table 8 into the three time terms of Definition 2 and optimise over the advantage a; the resulting lower-bound expression is minimized at $U = 23$. For smaller U the data term $c(a)2^{2f(U)}$ dominates (already $2^{175.03}$ at $U = 6$), for larger U the fast Walsh transform term $|k_0|2^{|k_0|}/2^{8.807}$ dominates ($2^{154.48}$ at $U = 26$), and the two cross between $U = 21$ and $U = 23$; at the minimum $U = 23$ the transform term $2^{136.302}$ and the final exhaustive-search term 2^{136} are almost equal. □

Corollary 2 does not depend on the cost model of the distillation table either. Deleting the term $|k_0| \cdot 2^{|k_0|}/2^{8.807}$ altogether (that is, positing some distillation method of negligible cost) leaves the time as $\max(N, 2^{192-a})$, whose lower-bound expression has minimum $2^{131.328}$ at $U \geq 23$, $w \geq 62.2616$ and $a = 61$, still 3.33 bits above the 2^{128} claim. In other words, the emptiness of the region inside 2^{128} needs only two ingredients: that the data is counted as $c(a)2^{2w}$, and that the remaining key must be exhausted.

Finally, we state the scope of this negative result. Theorem 8 is a statement about the dominant-trail class $\mathcal{A}$, not about the whole of linear cryptanalysis of SPEEDY-6-192. The following three kinds of attack fall outside its scope. (i) Arrangements with a distinguisher shorter than five rounds and key recovery on both sides, for instance

a 4-round approximation over rounds 1 to 4 together with two key-recovery rounds 0 and 5: applying the recurrence of Section 4.4 for four rounds gives 50.9413 at $U = 16$, corresponding to data of only $2^{108.75}$, so this arrangement is *not* excluded by the present bound; deciding it requires the optimal ciphertext-side masks and their key-guessing widths for a last round without MC, which we have not computed (Section 9). (ii) Full-hull optimisation, multiple linear cryptanalysis, or multidimensional linear cryptanalysis, whose data is not represented by the selected-trail expression $c(a)2^{2w}$. (iii) Arrangements without a final exhaustive step, that is, ones that determine the remaining key by some means other than 2^{192-a} trial decryptions. We stress that the cost model of the distillation table is *not* on this list: by the discussion above, a cheaper table does not affect the conclusion: even a table of zero cost still leaves a 3.33-bit margin. Of the three, we consider (i) the most worthwhile to pursue next.

7.5 Comparison with differential cryptanalysis and limitations

We next compare class $\mathcal{A}$ with the time and memory complexities of the 6-round differential attack, whose revised parameters are $2^{126.75}$ chosen ciphertexts, time $2^{156.77}$, and memory 2^{42} [2]. The comparison concerns only the numerical complexity range: the differential attack uses a chosen-ciphertext model, whereas class $\mathcal{A}$ uses known plaintexts.

A linear key-recovery attack in this range must satisfy

$$2^{|k_0|} \le 2^{42}, \qquad T \le 2^{156.77}.$$

The memory condition gives $U := |\mathrm{dil}_{\uparrow}(\mathrm{Rw}(a_1))| \le 7$. Table 8 then gives $w \ge 84.9022$ for $U = 6$ and $w \ge 77.8271$ for $U = 7$; with $N = 5.6 \cdot 2^{2w}$, the constant most favourable to the attacker from Section 2.4, the corresponding data lower bounds are $2^{172.29}$ and $2^{158.14}$, respectively. Both exceed the time threshold $2^{156.77}$; hence no parameter tuple in class $\mathcal{A}$ satisfies the time and memory bounds of the differential attack. This conclusion holds for arbitrary internal interface masks.

An independent exhaustive enumeration restricted to single-column internal interface masks gives the stronger class-specific trail bound $w \ge 83.8271$ for $|k_0| \le 42$. This result is consistent with, but is not needed for, the universal exclusion above. By contrast, exhaustive key search at time 2^{192} is not a lower bound for this attack class: the attack of Section 7.3 already has time $2^{170.20}$.

Together with the preceding bounds, this comparison characterises the defined dominant-trail FGNP/Walsh setting for 6-round SPEEDY. Within this setting, the results yield two conclusions:

- To the best of our knowledge, the attack of Section 7.3 is the first published dominant-trail-based linear key-recovery result for 6-round SPEEDY. The earlier candidate in Section 5.2 of [20] was invalidated by [5] and has unpruned-FWT cost $2^{323.53}$ equivalent encryptions according to Section 6. Every correlation used in the new attack is computed exactly by the algorithm of Section 3.
- Together with Theorem 8, the attack bounds this modelled class from both sides: under Definition 2, no attack can be faster than $2^{136.302}$ (Corollary 2), while what we actually construct is $2^{170.20}$, a difference of 33.90 bits. That difference comes

entirely from the gap between the trail bound and the trails we have constructed, not from the key-recovery accounting (end of Section 4.4).

Four limitations delimit the scope of these conclusions.

- It does not threaten the security claim of SPEEDY-6-192. The designers claim only 128-bit security for that instance (with a data limit of 2^{128} as well), whereas our attack takes time $2^{170.20}$ and data $2^{169.84}$. With respect to the design goal, SPEEDY-6-192 remains secure; Theorem 8 shows further that the defined dominant-trail class $\mathcal{A}$ cannot reach that range under the stated data and cost models.
- It uses known plaintexts, whereas the existing 6-round differential attacks use chosen ciphertexts. Their complexities are therefore not directly comparable.
- The memory of 2^{156} is substantial. This is inherent to attacks of the fast Walsh transform type (the distillation table has size $2^{|k_0|}$); and the exclusion established earlier in this section shows that, when memory is restricted to the 2^{42} of the differential attack, no trail meets the corresponding weight threshold, whatever its internal interface masks may be. Within that class, the high memory is therefore inherent to the attack parameters rather than a computational overhead.
- The structural feature it exploits is the absence of MC in the last round, so it cannot be transferred to the first six rounds of SPEEDY-7-192; likewise, Theorem 8 is a statement about SPEEDY-6-192.

8 Re-evaluation of a four-round differential-linear distinguisher

The exact algorithm of Section 3 is not confined to purely linear cryptanalysis. In this section we apply it to the differential-linear distinguisher of [20] and obtain a small but unambiguous correction.

Section 5.4 of [20] presents a 4-round differential-linear attack on SPEEDY-5-192 (2^{61} chosen plaintexts, time 2^{105}, memory 2^{105}), listed in the third row of Table 2 among the published SPEEDY-5-192 results relevant to this section. Its distinguisher correlation is estimated at $2^{-23.0950}$ and we recomputed the three components of that estimate; as recorded in footnote a of Table 2, the key-recovery part is not recomputed and the reported complexity is retained.

We use the standard three-part decomposition. The cipher is written as $E = E_2 \circ E_m \circ E_1$; p is the probability of the differential $\Delta_{\text{in}} \to \Delta_1$ over E_1, q is the correlation of the linear approximation $(\mu_1, \mu_{\text{out}})$ over E_2, and $r = \mathrm{Cor}_x\big[\langle \mu_1, E_m(x) \oplus E_m(x \oplus \Delta_1)\rangle\big]$ is the correlation of the middle part. Under the usual assumptions that the differential over E_1 is dominated by the single difference Δ_1 and that the three parts combine independently, the correlation of the distinguisher is estimated by

$$\mathrm{Cor}\big[\langle \mu_{\text{out}}, E(x) \oplus E(x \oplus \Delta_{\text{in}})\rangle\big] \approx p\, q^2\, r, \tag{1}$$

in which q enters squared (so its sign is irrelevant), while the signs of p and r carry through. We call the right-hand side the differential-linear estimate. Of the three

Table 17 The 4-round differential-linear distinguisher. The uncertainty on r is one standard error of the mean over 52 keys.

Quantity	[20]	This paper
Middle correlation r	$2^{-8.85}$	$2^{-8.4459 \pm 0.0130}$
Distinguisher (estimate)	$2^{-23.0950}$	$2^{-22.6909}$
Difference	—	0.4041 bits (about 0.81 bits of data)

factors, p is a probability in $[0, 1]$, while q and r are correlations in $[-1, 1]$; none of them is a bias. The figure quoted for the distinguisher is $|p\,q^2 r|$.

Both the differential and the linear part can be computed exactly with the method of Section 3:

$$p = 2^{-7.4150}, \qquad q = -2^{-3.4150}.$$

Here p comes from the entries 4 and 6 of the difference distribution table and q from the entries -16 and $+24$ of the linear approximation table; note that q is negative, which by (1) does not affect the result.

The middle part cannot be estimated reliably by trail-based methods and is measured over 52 random master keys with 2^{25} pairs per key. The resulting mean is $\bar{r} = +0.00286761$, i.e. $|\bar{r}| = 2^{-8.4459}$. The quoted uncertainty is one standard error of this mean, expressed in the exponent domain: with $s = 1.8718 \cdot 10^{-4}$ the observed standard deviation across the 52 keys, the standard error of the mean is $s/\sqrt{52} = 2.596 \cdot 10^{-5} = 0.905\%$ of $\bar{r}$, and $\log_2(1 + 0.00905) = 0.0130$, whence

$$r = 2^{-8.4459 \pm 0.0130} \quad \text{(one standard error; } \pm 0.0261 \text{ is the corresponding 95\% interval).}$$

Sampling error and key-to-key variation are not added separately: s is the total observed spread and therefore already contains both. Substituting into (1),

$$|p\,q^2 r| = 2^{-(7.4150 + 2 \cdot 3.4150 + 8.4459)} = 2^{-22.6909},$$

against the published $2^{-23.0950}$. Since r is measured rather than computed, this figure is an estimate and not an exact correlation. The results are summarised in Table 17.

The measured middle correlation is supported by the following checks.

- The middle map agrees with an independent bitwise evaluation at three levels.
- All 52 keys give a positive sign; the stability of r across keys passes a χ^2 test at 59.96 (51 degrees of freedom, $p = 0.183$), consistent with identical distributions across keys. The observed spread of the 52 per-key values is 6.53% of $\bar{r}$, against $1/\sqrt{N} = 6.02\%$ expected from the sampling of a single fixed correlation; the ratio of the two is 1.084, so subtracting the sampling component in quadrature leaves an inter-key component of at most 2.52% of $\bar{r}$, i.e. 0.036 bits. The key dependence of the middle part is therefore well below the 0.404-bit gap that the distinguisher exhibits.
- The worst of the 52 keys gives $2^{-8.6504}$, still 0.200 bits above the published value.
- An independent measurement with 32 keys and 2^{23} pairs per key gives $2^{-8.4487}$.

We did not recompute the full key-recovery part of that attack. The 2^{105} time complexity reported in Section 5.4 of [20] is therefore retained in Table 2; whether the 0.81 bits of data saving can be converted into an improvement in time is left open (Section 9).

9 Conclusion

The central finding of this work is that evaluating the intra-round correlation exactly changes the assessment of the published linear cryptanalysis of SPEEDY. Establishing it required three main contributions. First, it gives and independently verifies an exact one-round correlation algorithm and develops the corresponding framework for nonlinear segments without intervening key additions (Section 3). Second, it corrects the published 5-round distinguisher and reassesses the dimension of the Walsh-spectrum support, the cost of the unpruned fast Walsh transform, and affine-subspace strategies restricted to 6-bit-aligned blocks for the associated SPEEDY-7-192 key recovery (Sections 5 and 6). Third, it introduces covering numbers into the trail search, constructs a 6-round SPEEDY-6-192 attack, and derives lower bounds for the defined dominant-trail FGNP/Walsh class (Sections 4.4 and 7). We also apply the exact correlation algorithm to the differential-linear correction of Section 8.

All ELP values are stated in the standard independent-round-key model, and the key-recovery data estimates use the dominant-trail approximation. Under the first model, the corrected distinguisher gives ELP $\geq 2^{-166.3865}$, 19.64 bits above the published estimate, and exhaustive enumeration excludes every single-position improvement. For SPEEDY-7-192, the affine dimension of the Walsh-spectrum support is 324; under the normalized cost of Theorem 7, affine-subspace pruning restricted to 6-bit-aligned blocks costs at least $2^{281.33}$ encryptions for one subspace and $2^{199.97}$ for a union.

Two methodological points are central. Within a nonlinear segment without intervening key additions, differential contributions are unsigned but linear contributions are signed; the latter may reinforce or cancel, so the direction and magnitude of a layer-wise error must be determined by exact evaluation. Separately, key-schedule dependencies reduce the number of candidate keys but do not automatically reduce the domain of the Walsh transform. This distinction is what changes the reassessed SPEEDY-7 complexity.

The scope is correspondingly limited. The trail bounds remain below the 94.7573-bit attack threshold and do not prove general linear resistance. Theorem 8 and Corollary 2 exclude attacks only within Definition 2; they do not cover full-hull optimisation, multiple or multidimensional linear cryptanalysis, shorter distinguishers with key recovery on both sides, or alternative data formulas. Theorem 7 covers only affine subspaces aligned with the 6-bit blocks and reports normalized costs under the coherent retained-energy assumption. Finally, the constructed SPEEDY-6-192 attack exceeds the 128-bit security claim and uses known plaintexts, whereas the existing differential attacks use chosen ciphertexts.

Declarations

Funding. This work was supported by the National Cyber Security-National Science and Technology Major Project (No. 2026ZD1500700) and by the National Natural Science Foundation of China (Nos. 62172427 and 61702537).

Competing interests. The authors declare no competing interests.

Data availability. The programs and data that reproduce the computational results of this paper are submitted with the manuscript as `supplementary_code.zip`, so that the reviewers and the editors can check them; they are also available from the authors on reasonable request.

References

[1] Beyne T., Neyt A.: Note on the cryptanalysis of SPEEDY. IACR Cryptology ePrint Archive 2024/262 (2024)

[2] Beyne T., Neyt A.: Improved differential cryptanalysis of SPEEDY. IACR Trans. Symmetric Cryptol. 2025(3), 444–474 (2025). ePrint 2025/915

[3] Biryukov A., Khovratovich D.: PAEQ: parallelizable permutation-based authenticated encryption. In: ISC 2014, vol. 8783, pp. 72–89. LNCS. Springer (2014)

[4] Boura C., David N., Heim Boissier R., Naya-Plasencia M.: Better steady than speedy: full break of SPEEDY-7-192. In: EUROCRYPT 2023, vol. 14007, pp. 36–66. LNCS. Springer (2023). ePrint 2022/1351

[5] Boura C., Derbez P., Germon B., Heim Boissier R., Naya-Plasencia M.: SPEEDY: caught at last. In: ASIACRYPT 2025, Part I, vol. 16245, pp. 189–220. LNCS. Springer (2026). ePrint 2025/890

[6] Boura C., Derbez P., Germon B., Heim Boissier R., Naya-Plasencia M.: A note on a recent attack against SPEEDY-7-192. IACR Cryptology ePrint Archive 2025/1265 (2025)

[7] Daemen J., Govaerts R., Vandewalle J.: Correlation matrices. In: FSE 1994, vol. 1008, pp. 275–285. LNCS. Springer (1995)

[8] Daemen J., Rijmen V.: Understanding two-round differentials in AES. In: SCN 2006, vol. 4116, pp. 78–94. LNCS. Springer (2006)

[9] Flórez-Gutiérrez A.: Optimising linear key recovery attacks with affine Walsh transform pruning. In: ASIACRYPT 2022, Part IV, vol. 13794, pp. 447–476. LNCS. Springer (2022)

[10] Flórez-Gutiérrez A., Naya-Plasencia M.: Improving key-recovery in linear attacks: application to 28-round PRESENT. In: EUROCRYPT 2020, Part I, vol. 12105, pp. 221–249. LNCS. Springer (2020)

[11] Flórez-Gutiérrez A., Todo Y.: Improving linear key recovery attacks using Walsh spectrum puncturing. In: EUROCRYPT 2024, Part I, vol. 14651, pp. 187–216. LNCS. Springer (2024). Extended version: J. Cryptol. 38(4), article 34 (2025). ePrint 2024/151

[12] Gueron S., Mouha N.: Simpira v2: a family of efficient permutations using the AES round function. In: ASIACRYPT 2016, Part I, vol. 10031, pp. 95–125. LNCS. Springer (2016)

[13] Hou C., Wang S., Liu M.: Walsh spectrum puncturing revisited: toward automated linear key recovery attacks. IACR Trans. Symmetric Cryptol. 2026(1), 318–344 (2026)

[14] Kölbl S., Lauridsen M.M., Mendel F., Rechberger C.: Haraka v2: efficient short-input hashing for post-quantum applications. IACR Trans. Symmetric Cryptol. 2016(2), 1–29 (2016). ePrint 2016/098

[15] Leander G., Moos T., Moradi A., Rasoolzadeh S.: The SPEEDY family of block ciphers: engineering an ultra low-latency cipher from gate level for secure processor architectures. IACR Trans. Cryptogr. Hardw. Embed. Syst. 2021(4), 510–545 (2021). ePrint 2021/960

[16] Matsui M.: New block encryption algorithm MISTY. In: FSE 1997, vol. 1267, pp. 54–68. LNCS. Springer (1997)

[17] Rohit R., Sarkar S.: Cryptanalysis of reduced round SPEEDY. In: AFRICACRYPT 2022, vol. 13503, pp. 133–149. LNCS. Springer (2022). ePrint 2022/612

[18] Selçuk A.A.: On probability of success in linear and differential cryptanalysis. J. Cryptol. 21(1), 131–147 (2008). Preliminary version: SCN 2002, vol. 2576, pp. 174–185. LNCS (with Biçak A.)

[19] Shibutani K., Isobe T., Hiwatari H., Mitsuda A., Akishita T., Shirai T.: Piccolo: an ultra-lightweight blockcipher. In: CHES 2011, vol. 6917, pp. 342–357. LNCS. Springer (2011)

[20] Wang J., Niu C., Liu Q., Li M., Preneel B., Wang M.: Cryptanalysis of SPEEDY. In: ACISP 2023, vol. 13915, pp. 124–156. LNCS. Springer (2023). Full version: ePrint 2023/612

[21] Yu Q., Jia K., Zou G., Zhang G.: Differential cryptanalysis of round-reduced SPEEDY family. In: Inscrypt 2022, vol. 13837, pp. 272–291. LNCS. Springer

(2022)

[22] Zhang L.: Observations on the branch number and differential analysis of SPEEDY. Des. Codes Cryptogr. 92(5), 1175–1199 (2024)