Analyzing the Linear Keystream Biases in AEGIS

Maria Eichlseder, Marcel Nageler and Robert Primas

Graz University of Technology, Graz, Austria
maria.eichlseder@iaik.tugraz.at
marcel.nageler@student.tugraz.at
rprimas@gmail.com

Abstract. AEGIS is one of the authenticated encryption designs selected for the final portfolio of the CAESAR competition. It combines the AES round function and simple Boolean operations to update its large state and extract a keystream to achieve an excellent software performance. In 2014, Minaud discovered slight biases in the keystream based on linear characteristics. For family member AEGIS-256, these could be exploited to undermine the confidentiality faster than generic attacks, but this still requires very large amounts of data. For final portfolio member AEGIS-128, these attacks are currently less efficient than generic attacks. We propose improved keystream approximations for the AEGIS family, but also prove upper bounds below 2^{-128} for the squared correlation contribution of any single suitable linear characteristic.

Keywords: Authenticated encryption · CAESAR · AEGIS · Linear cryptanalysis

1 Introduction

AEGIS [WP16, WP13] is a family of authenticated ciphers with excellent performance in high-speed software applications thanks to an AES-based state update function combined with bitwise operations And and Xor to extract a keystream from the large internal state. The AEGIS family, consisting of family members AEGIS-128, AEGIS-256, and AEGIS-128L, is a finalist of the CAESAR competition and in 2019, family member AEGIS-128 was elected to be part of the final CAESAR portfolio [CAE19].

Surprisingly, very few cryptanalytic results on AEGIS have been published so far. The designers provide a general security analysis of AEGIS [WP16] including comments on generic attacks, nonce misuse attacks, and differential attacks on the initialization, state update, and finalization. Based on a very conservative bound on active AES rounds and active S-boxes, they argue that an attacker trying to inject differences via the message to produce internal collisions will only obtain differential characteristics with a probability of less than 2^{-156} (for AEGIS-128 and AEGIS-256) or less than 2^{-150} (for AEGIS-128L), leading to attack complexities higher than generic forgery attempts. Vaudenay and Vizár [VV18] and Kales et al. [KEM17] further analyze the security of AEGIS under nonce misuse and propose state recovery attacks.

The most notable third-party cryptanalysis result on AEGIS was published by Minaud [Min14] soon after the beginning of the CAESAR competition. He proposes keystream approximations based on linear characteristics of the round function. For AEGIS-128, the attack is based on a characteristics with squared correlation contribution 2^{-154}, and Minaud estimates an attack would require about 2^{140} blocks of data, which is still above the security claim. For AEGIS-256, however, the resulting keystream distinguisher with 2^{178} blocks of data is significantly better than generic attacks. While these attacks have very high data requirements, the data may be collected across different secret keys as long as the plaintexts remain known or constant. AEGIS-256 was not selected as part of the

final CAESAR portfolio and remains a finalist. The third family member and the designers' primary recommendation, AEGIS-128L, was not discussed in Minaud's analysis, but is also not part of the final portfolio: in the announcement of the finalists in 2018 [Ber18], it was already declared that at most one of the two members AEGIS-128 and AEGIS-128L would proceed to the final portfolio. To the best of our knowledge, no further improvements or bounds on similar attacks have since been proposed.

Related work. Minaud's analysis of AEGIS [Min14] has also inspired similar keystream bias attacks on another CAESAR finalist, MORUS: Ashur et al. [AEL+18] first proposed a distinguisher based on a linear characteristic found by hand and discussed in more detail how such keystream correlations could be exploited in practice, referring to TLS attacks that exploit the biased keystream of RC4 [IOWM13, ABP+13]. Shi et al. [SSS+19] then succeeded in substantially improving the distinguishers by applying Mixed-Integer Linear Programming (MILP) solvers to search for better characteristics. While the state update functions of MORUS and AEGIS are quite different and the ideas introduced by Shi et al. are not directly relevant for AEGIS, their results still illustrate the advantages of off-the-shelf solvers for finding better attacks. AES and (tweakable) block ciphers with a similar structure have been among the first [MWGP11] and most popular targets of MILP-based cryptanalysis [CHP+17, BJK+16, Ava17]. For strongly aligned designs like AES, models are usually truncated to cell-level, since bitwise models of large S-boxes [AST+17] are usually too costly; but for other designs, bitwise models have been applied for linear cryptanalysis and other attack vectors [SHW+14, TIM+18, SSS+19]. Besides improving cryptanalytic attacks, MILP models have been particularly popular with designers for proving bounds on the maximum possible differential probability of differential characteristics or the maximum possible correlation contribution of linear characteristics. However, somewhat surprisingly, neither attempts at improved attacks nor upper bounds on attacks similar to Minaud's have been published. For AEGIS-128 in particular, the latter would be relevant in order to better understand its security against an attacker as defined by the 128-bit security claim.

Contribution. We search for better linear characteristics, as well as upper bounds on the best possible correlation. We observe that straightforward truncated models of linear characteristics of AEGIS only produce very weak bounds since they fail to capture connections and constraints that follow from dependencies in the AEGIS state update function. To obtain tighter bounds and consistent solutions, we identify additional constraints on the differences and higher-order differences of the linear masks and propose an improved truncated model. This model yields much better results, including consistent solutions for AEGIS-128, but still shows a significant gap between the bounds and the best found characteristics, mainly due to the Boolean output function. We propose a partially bitwise model to close this gap. As a result, for all AEGIS family members, we derive upper bounds below 2^{-128} for the squared correlation contribution of any single suitable linear characteristic. Finally, we apply Constraint Programming (CP) to find consistent characteristics and obtain improved attacks for all members. Table 1 details our results.

Table 1: Bounds for the inverse squared correlation contribution c^{-2} of the best suitable linear characteristics of AEGIS. Lower bounds are derived using MILP models, upper bounds are based on best found characteristics (see Subsection 4.3 on their accuracy).

	AEGIS-128	AEGIS-256	AEGIS-128L
Manual analysis [Min14]	$c^{-2} \leq 2^{154}$	$c^{-2} \leq 2^{178}$	
Truncated model (Sect. 3.2)	$2^{92} \leq c^{-2}$	$2^{116} \leq c^{-2}$	$2^{114} \leq c^{-2} \leq 2^{172}$
Improved model (Sect. 3.3)	$2^{102} \leq c^{-2} \leq 2^{140}$	$2^{120} \leq c^{-2}$	
Bitwise model (Sect. 3.4)	$2^{132} \leq c^{-2} \leq 2^{140}$	$2^{152} \leq c^{-2} \leq 2^{162}$	$2^{140} \leq c^{-2} \leq 2^{152}$

Outline. We recall the design and previous analysis of the AEGIS family in Section 2. In Section 3 we develop several successively refined MILP models of truncated linear characteristics for AEGIS and derive upper bounds on the squared correlation contribution of linear characteristics. In Section 4 we reuse the best truncated MILP solutions as a basis for searching consistent characteristics with CP solvers and find improved attacks. Our source code is available on https://extgit.iaik.tugraz.at/krypto/aegis_linear_trails.

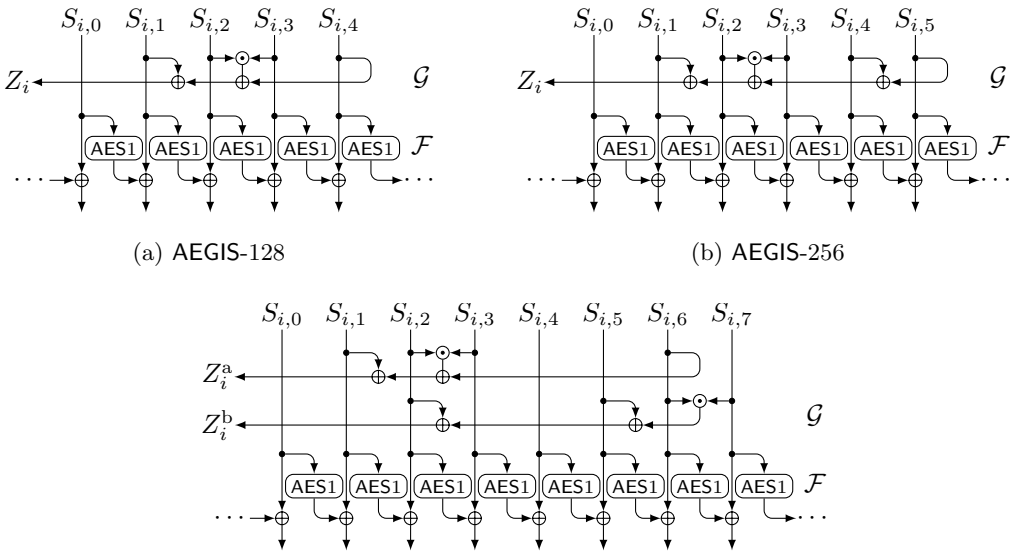
2 Description and Previous Analysis of AEGIS

2.1 The AEGIS Family of Authenticated Ciphers

AEGIS [WP16] is a stream cipher design that achieves high performance by utilizing hardware support of the AES round function that is nowadays available on a large variety of devices. The final version of the submission document specifies 3 versions of AEGIS: AEGIS-128, AEGIS-128L, and AEGIS-256, with their main difference being their state and key sizes. While AEGIS-128 uses a 640-bit state that is comprised of 5×128-bit AES states, AEGIS-256 uses a 768-bit state (6×128-bit), and AEGIS-128L uses a 1024-bit state (8×128-bit) to fully utilize the 8-staged pipeline of AES instructions on modern desktop CPUs. The key sizes are 128-bit for AEGIS-128 and AEGIS-128L, and 256-bit for AEGIS-256.

The schemes encrypt a message M of arbitrary length to a ciphertext C of the same length plus a 128-bit authentication tag T, both depending on a 128-bit (for AEGIS-$\{128, 128\text{L}\}$) or 256-bit (for AEGIS-256) key K and nonce IV as well as associated data A of arbitrary length. The encryption and decryption procedures of AEGIS are split into 4 stages: initialization, absorbing associated data, encryption or decryption of message blocks, and the finalization.

For AEGIS-128, these are defined as follows. During the initialization, the inner state $S_{-10,0}, \ldots, S_{-10,4}$ is initialized with the key K, nonce IV, and some constants, followed



(a) AEGIS-128

(b) AEGIS-256

(c) AEGIS-128L

Figure 1: State update function of the AEGIS family members.

by 10 iterations of the state update function $\mathcal{F}(S_i, x_i)$ illustrated in Figure 1:

$$\begin{aligned} S_{i+1,0} &= \mathsf{AES1}(S_{i,4}) \oplus S_{i,0} \oplus m_i \\ S_{i+1,1} &= \mathsf{AES1}(S_{i,0}) \oplus S_{i,1} \\ S_{i+1,2} &= \mathsf{AES1}(S_{i,1}) \oplus S_{i,2} \\ S_{i+1,3} &= \mathsf{AES1}(S_{i,2}) \oplus S_{i,3} \\ S_{i+1,4} &= \mathsf{AES1}(S_{i,3}) \oplus S_{i,4}\,, \end{aligned}$$

where AES1 is the AES round function without key addition and m_i denotes a data block, which is derived from K and IV during initialization. The associated data is processed by repeatedly calling $\mathcal{F}$ using blocks A_i of the associated data A as m_i until all blocks are absorbed. AEGIS then encrypts each plaintext block M_i by extracting a keystream block Z_i from the large internal state with the output function $\mathcal{G}$, where $\oplus$ denotes bitwise Xor and $\odot$ is bitwise And of substates:

$$Z_i = S_{i,1} \oplus S_{i,4} \oplus (S_{i,2} \odot S_{i,3})$$

to compute the ciphertext block $C_i = M_i \oplus Z_i$ and updating the internal state with $\mathcal{F}$:

$$S_{i+1} = \mathcal{F}(S_i, M_i)\,.$$

During the finalization, 7 state updates are performed with a message block $m = S_{i,3} \oplus (|A| \parallel |M|)$ containing the associated data and message lengths $|A|$ and $|M|$ as 64-bit integers:

$$S_{i+1} = \mathcal{F}(S_i, m)\,.$$

Then 128-bit tag T can be derived as $T = \bigoplus_{j=0}^{4} S_{i,j}$.

The descriptions for AEGIS-256 and AEGIS-128L are similar, the main differences are due to the different state and key sizes. Most notably, the number of initialization rounds for AEGIS-256 is increased to 16, the output function $\mathcal{G}$ in AEGIS-256 is defined as:

$$Z_i = S_{i,1} \oplus S_{i,4} \oplus S_{i,5} \oplus (S_{i,2} \odot S_{i,3})\,,$$

whereas $\mathcal{G}$ in AEGIS-128L is defined as:

$$\begin{aligned} Z_i &= S_{i,1} \oplus S_{i,6} \oplus (S_{i,2} \odot S_{i,3}) \\ Z_{i+1} &= S_{i,2} \oplus S_{i,5} \oplus (S_{i,6} \odot S_{i,7})\,, \end{aligned}$$

and thus produces two ciphertext blocks at once. For a full specification, we refer to the design papers [WP16, WP13].

2.2 Minaud's Analysis of AEGIS

Minaud [Min14] showed that an attacker can exploit biased linear approximations involving only the keystream $Z_0, Z_1, \ldots$ to undermine the confidentiality of AEGIS. The attack requires collecting sufficient amounts of data with known or constant plaintext blocks.

The complexity of the attacks depends on the correlation of the bits selected by the keystream masks. We consider linear characteristics [Mat93] and evaluate their correlation contribution c [DGV94, DR02], computed as the product of the correlations $2p - 1$ of the individual approximations of nonlinear operations (S-boxes, And) using their probability p. Assuming that the correlation contribution is a good estimate for the actual correlation, we expect a data complexity proportional to the inverse squared correlation contribution c^{-2} to distinguish the keystream based on the approximation.

Minaud proposes linear characteristics covering 2 iterations of $\mathcal{F}$, corresponding to 3 keystream blocks Z_i, Z_{i+1}, Z_{i+2}. For AEGIS-128, the squared correlation contribution is $c^2 = 2^{-154}$, and for AEGIS-256, $c^2 = 2^{-178}$. In an appendix, he observes that based on an exhaustive evaluation of larger blocks of the characteristic, the squared correlation appears to be higher by a factor of about 2^{10}, i.e., about 2^{-144}. Furthermore, he estimates that by using permuted variants of the same characteristic to obtain 16 different approximations, the data requirements could be as low as about 2^{140} instead of about 2^{154}.

3 Upper Bounds for the Keystream Bias in AEGIS

In this section, we develop several successively refined Mixed-Integer Linear Programming (MILP) models of truncated linear characteristics for AEGIS and derive upper bounds on the squared correlation contribution of linear characteristics. We first provide some context and introduce the necessary notation in Subsection 3.1. In Subsection 3.2, we define a simple first model following the usual modeling approach for AES-like designs but obtain only very weak bounds and no consistent solutions. In Subsection 3.3, we identify several reasons for these inconsistencies and consequently extend the simple model with additional constraints on differences and second-order differences of linear masks to obtain tighter bounds as well as consistent truncated solutions. Since there is still a significant gap between the truncated bounds and the best found bitwise solutions, we proceed with a bitwise model of all linear operations in Subsection 3.4. This last model improves the bounds significantly and proves that, under Markov assumptions, no individual linear characteristics for attacks similar to Minaud's have a squared correlation contribution better than 2^{-128} for AEGIS-128, AEGIS-256, or AEGIS-128L.

3.1 Notation for Linear Approximations of the AEGIS Keystream

We aim to find better approximations with a higher bias for more efficient attacks than Minaud's manually constructed approximations, as well as upper bounds on the best possible bias to better evaluate the security margin of AEGIS. To find and evaluate approximations, we model linear characteristics for the state update function $\mathcal{F}$ and the output function $\mathcal{G}$ with MILP. We denote the number of rounds involved as k. Thus we consider the keystream blocks Z_i to Z_{i+k}, denoted Z_0 to Z_k in the following for the sake of simplicity. Similar to the notation for MORUS by Shi et al. [SSS+19], we denote the linear masks for the i-th output function call $\mathcal{G}$ by (γ_i, λ_i) and those for the following state update $\mathcal{F}$ by (α_i, β_i), as illustrated in Figure 2.

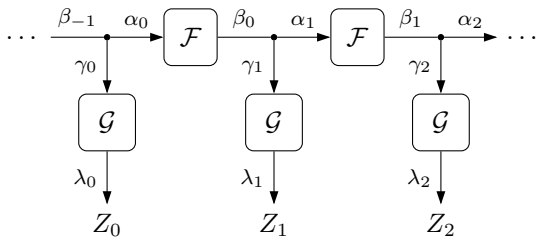


Figure 2: Linear masks $\alpha_i, \beta_i, \gamma_i, \lambda_i$ for 3 consecutive keystream blocks Z_i of AEGIS.

Additionally, we need masks for the inputs of the SubBytes, ShiftRows, and MixColumns layers, denoted as σ_i, ρ_i, and μ_i, respectively. The output mask of MixColumns must equal the corresponding round function output mask β_i due to the following XOR operation.

Besides the round number i, masks like $\alpha_{i,j}[r,c,b]$ are indexed as follows, where index computations for j,r,c are always modulo the corresponding set size (see Figure 3):

$$\begin{aligned}
&\text{Round number} && i \in \begin{cases} \mathcal{I} = \{0,\dots,k-1\}, & \text{for round functions } \mathcal{F} \\ \mathcal{I}' = \{0,\dots,k\}, & \text{for output functions } \mathcal{G} \end{cases} \\
&\text{Substate index} && j \in \mathcal{J} = \begin{cases} \{0,\dots,4\}, & \text{for AEGIS-128} \\ \{0,\dots,5\}, & \text{for AEGIS-256} \\ \{0,\dots,7\}, & \text{for AEGIS-128L} \end{cases} \\
&\text{AES Row} && r \in \mathcal{R} = \{0,1,2,3\} \\
&\text{AES Column} && c \in \mathcal{C} = \{0,1,2,3\} \\
&\text{Bit position} && b \in \mathcal{B} = \{0,\dots,7\}\,.
\end{aligned}$$

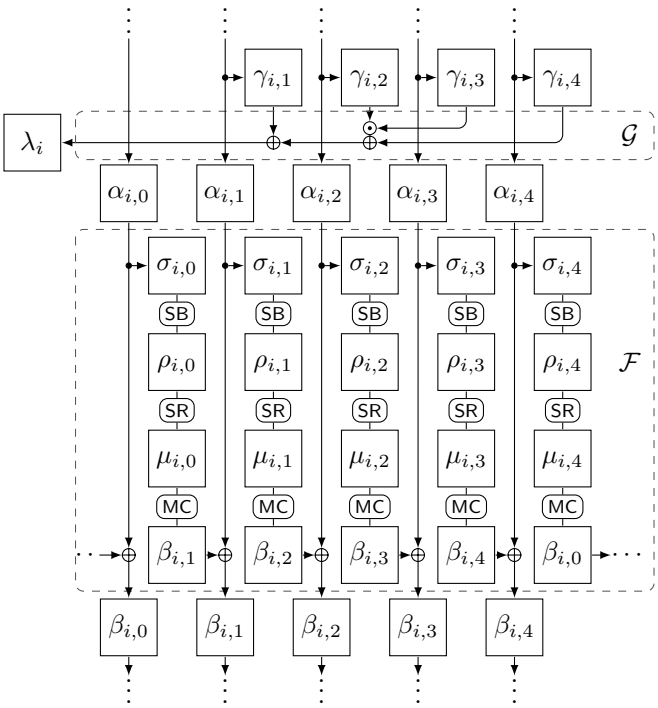


Figure 3: Notation for linear substate masks in round i of AEGIS-128.

3.2 Simple Truncated Model

We want to find a lower bound on the number of active S-boxes and byte-AND-operations using a MILP model of the truncated linear behavior of AEGIS. This truncated model defines which byte-sized cells of the state are active (i.e., have non-zero masks) and which ones are not for compatible linear characteristics. We refer to a truncated characteristic as consistent or valid if there is at least one compatible linear characteristic with non-zero correlation contribution (estimated under Markovian independence assumptions), and as inconsistent or contradictory otherwise. We denote the truncated mask by adding a bar to the above cell mask names; for example, the truncated mask $\bar{\alpha}_{i,j}[r,c]$ for byte mask $\alpha_{i,j}[r,c]$ is active if any of its bits $\alpha_{i,j}[r,c,b]$ is active: $\bar{\alpha}_{i,j}[r,c] = \bigvee_{b\in\mathcal{B}} \alpha_{i,j}[r,c,b]$, with $\vee$ denoting logical OR. The MILP model includes a binary decision variable for each truncated mask corresponding to the masks in Figure 3.

In the following, we describe the constraints imposed on $\bar{\alpha}$, $\bar{\beta}$, $\bar{\gamma}$, and $\bar{\lambda}$ by the state update function $\mathcal{F}$ and the output function $\mathcal{G}$, which are very similar to other AES-based truncated models inspired by the original model of Mouha et al. [MWGP11]. In particular, many of the steps involving some N truncated mask variables $x_1, \ldots, x_N$ are defined by their branch number B, i.e., either 0 or $\geq B$ of the variables must be active. This can be modeled with two $\mathbb{R}$-linear MILP equations using a binary dummy variable d that is active iff any of the x_n is active as $B \cdot \mathrm{d} \leq \sum_{n=1}^{N} x_n \leq N \cdot \mathrm{d}$.

3.2.1 Constraints for the State Update Function $\mathcal{F}$ and Output Function $\mathcal{G}$

Model of the Mode: Branches and Boundary Constraints For the branch between $\mathcal{F}$ and $\mathcal{G}$ that links masks β_{i-1}, α_i, and γ_i, each bit position must be active in either 0 or 2 of the 3 masks. However, we are modeling bytes, so within one byte some bits may come from the first part of the branch while other bits are modeled using the other side of the branch. This leaves us with the options of 0, 2, or 3 bytes active in the truncated model, i.e., $\forall i \in \mathcal{I}', j \in \mathcal{J}, r \in \mathcal{R}, c \in \mathcal{C} : \bar{\beta}_{i-1,j}[r,c] + \bar{\alpha}_{i,j}[r,c] + \bar{\gamma}_{i,j}[r,c] \in \{0,2,3\}$. This corresponds to a linear function with branch number 2 and can be modeled using a binary dummy variable $\mathrm{d}_{i,j}^{\vdash\bar{\gamma}}[r,c] \in \{0,1\}$ for each cell:

$$2\,\mathrm{d}_{i,j}^{\vdash\bar{\gamma}}[r,c] \leq \bar{\beta}_{i-1,j}[r,c] + \bar{\alpha}_{i,j}[r,c] + \bar{\gamma}_{i,j}[r,c] \leq 3\,\mathrm{d}_{i,j}^{\vdash\bar{\gamma}}[r,c] \quad \forall i \in \mathcal{I}', j \in \mathcal{J}, r \in \mathcal{R}, c \in \mathcal{C}\,.$$

Alternatively, we can eliminate the dummy variable by disallowing all invalid transitions, which leads to significantly reduced solving runtimes:

$$\begin{aligned}
\bar{\beta}_{i-1,j}[r,c] + \bar{\gamma}_{i,j}[r,c] &\geq \bar{\alpha}_{i,j}[r,c] && \forall i \in \mathcal{I}', j \in \mathcal{J}, r \in \mathcal{R}, c \in \mathcal{C}\,,\\
\bar{\alpha}_{i,j}[r,c] + \bar{\gamma}_{i,j}[r,c] &\geq \bar{\beta}_{i-1,j}[r,c] && \forall i \in \mathcal{I}', j \in \mathcal{J}, r \in \mathcal{R}, c \in \mathcal{C}\,,\\
\bar{\alpha}_{i,j}[r,c] + \bar{\beta}_{i-1,j}[r,c] &\geq \bar{\gamma}_{i,j}[r,c] && \forall i \in \mathcal{I}', j \in \mathcal{J}, r \in \mathcal{R}, c \in \mathcal{C}\,.
\end{aligned}$$

The alternative model provided significant speedups for the bitwise model which will be discussed in Section 3.4.

Since the internal state of the cipher is unknown, we cannot use bits of the internal state before and after our characteristic as part of our approximation. Hence, for a k-round characteristic, we require

$$\bar{\beta}_{-1} = 0, \quad \bar{\alpha}_k = 0\,.$$

Finally, to exclude all-zero characteristics, we add a non-triviality constraint on $\bar{\lambda}_0$; a similar constraint can optionally be added for the last keystream block $\bar{\lambda}_k$ to set a precise number of keystream blocks instead of a maximum number:

$$\sum_{\substack{r \in \mathcal{R}\\ c \in \mathcal{C}}} \bar{\lambda}_0[r,c] \geq 1\,.$$

Model of the State Update Function $\mathcal{F}$ The state update function takes the input substates (masks α_i), branches each substate to apply one AES round (input masks σ_i), and XORs the result to the neighbouring substate. Thus, the masks for the output of $\mathcal{F}$, outputs of MixColumns, and the state after branching the AES round input must all equal $\beta_i = \alpha_i \oplus \sigma_i$. We model the branching in exactly the same way as the one for $\mathcal{G}$, either by using a binary dummy variable $\mathrm{d}_{i,j}^{\vdash\bar{\sigma}}[r,c] \in \{0,1\}$ for each cell as follows or with the alternative model:

$$2\,\mathrm{d}_{i,j}^{\vdash\bar{\sigma}}[r,c] \leq \bar{\alpha}_{i,j}[r,c] + \bar{\beta}_{i,j}[r,c] + \bar{\sigma}_{i,j}[r,c] \leq 3\,\mathrm{d}_{i,j}^{\vdash\bar{\sigma}}[r,c] \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, r \in \mathcal{R}, c \in \mathcal{C}\,.$$

For SubBytes, each output byte is active iff the input byte is active. For ShiftRows, we require the equality of the variables according to the specification of the ShiftRows step. For the truncated model, both steps boil down to a renaming of variables:

$$\begin{aligned} \bar{\rho}_i &= \bar{\sigma}_i && \forall i \in \mathcal{I}\,, \\ \bar{\mu}_{i,j}[r,c] &= \bar{\rho}_{i,j}[r, r+c] && \forall i \in \mathcal{I}, j \in \mathcal{J}, r \in \mathcal{R}, c \in \mathcal{C}\,. \end{aligned}$$

The truncated linear behavior of MixColumns is defined by its branch number of 5 [DR98], i.e., the sum of active truncated input and output masks must be in $\{0, 5, 6, 7, 8\}$. Using a binary dummy variable $\mathrm{d}_{i,j}^{\mathsf{MC}\bar{\mu}}[*, c] \in \{0, 1\}$ per column, we get

$$5\,\mathrm{d}_{i,j}^{\mathsf{MC}\bar{\mu}}[*,c] \leq \sum_{r\in\mathcal{R}} \bar{\beta}_{i,j+1}[r,c] + \sum_{r\in\mathcal{R}} \bar{\mu}_{i,j}[r,c] \leq 8\,\mathrm{d}_{i,j}^{\mathsf{MC}\bar{\mu}}[*,c] \qquad \forall i \in \mathcal{I}, j \in \mathcal{J}, c \in \mathcal{C}\,.$$

Model of the Output Function $\mathcal{G}$ The output functions for AEGIS-128 and AEGIS-256 compute the Xor of several substates and the And of two substates. The Xor implies

$$\begin{cases} \bar{\lambda}_i = \bar{\gamma}_{i,1} = \bar{\gamma}_{i,4} & \text{for AEGIS-128} \\ \bar{\lambda}_i = \bar{\gamma}_{i,1} = \bar{\gamma}_{i,4} = \bar{\gamma}_{i,5} & \text{for AEGIS-256} \end{cases} \qquad \forall i \in \mathcal{I}'\,.$$

The And operation is the same for both variants of the cipher. Its output can be approximated using any input mask, but an active input mask implies an active output:

$$\bar{\gamma}_{i,2}[r,c] \leq \bar{\lambda}_i[r,c] \qquad \text{and} \qquad \bar{\gamma}_{i,3}[r,c] \leq \bar{\lambda}_i[r,c] \qquad \forall i \in \mathcal{I}', r \in \mathcal{R}, c \in \mathcal{C}\,.$$

Furthermore, because the leftmost substate is not part of the output function, $\bar{\gamma}_{i,0} = 0$.

For AEGIS-128L, the output function computes two outputs with masks $\lambda_i^{\mathrm{a}}, \gamma_i^{\mathrm{a}}$ and $\lambda_i^{\mathrm{b}}, \gamma_i^{\mathrm{b}}$ with similar functions, where $\gamma_i = \gamma_i^{\mathrm{a}} \oplus \gamma_i^{\mathrm{b}}$. The truncated model is analogous, with the additional requirements that $\bar{\gamma}_{i,0}^{\mathrm{a}} = \bar{\gamma}_{i,4}^{\mathrm{a}} = \bar{\gamma}_{i,5}^{\mathrm{a}} = \bar{\gamma}_{i,7}^{\mathrm{a}} = \bar{\gamma}_{i,0}^{\mathrm{b}} = \bar{\gamma}_{i,1}^{\mathrm{b}} = \bar{\gamma}_{i,3}^{\mathrm{b}} = \bar{\gamma}_{i,4}^{\mathrm{b}} = 0$.

3.2.2 Objective Function

The quality of our truncated linear characteristic is determined by (an upper bound on) the maximum possible correlation contribution c of a compatible characteristic, which we want to maximize. By assuming the independence of the parts of the characteristic, we can model the overall correlation contribution as the product of individual correlations [DGV94, DR02], similar to the piling-up lemma [Mat93] for the bias $\varepsilon = c/2$. Then, the inverse of the squared correlation contribution, c^{-2}, is an estimate for the necessary data to exploit the corresponding approximation.

To obtain a linear objective function, we equivalently work with the cost function corresponding to the logarithm of the squared correlation contribution, $w = -2\log_2|c|$. In order to derive an upper bound, we want to rate each active nonlinear component with the upper bound of its correlation. The best correlation for an active S-box is $c = \pm 2^{-3}$, thus we rate it with $w = 6$. An active byte in the And-operation at the output function requires at least one active bit, thus the maximum absolute correlation is given by $c = \pm 2^{-1}$ and we rate it with $w = 2$. The resulting objective function is

$$\text{minimize} \quad 6 \sum_{\substack{i\in\mathcal{I}\\ j\in\mathcal{J}\\ r\in\mathcal{R}\\ c\in\mathcal{C}}} \bar{\sigma}_{i,j}[r,c] + 2 \sum_{\substack{i\in\mathcal{I}'\\ r\in\mathcal{R}\\ c\in\mathcal{C}}} \bar{\lambda}_i[r,c]\,.$$

3.2.3 Results for the Truncated Model

We can consider models for a varying number of keystream blocks. There are no solutions for only one round of the state update function and two blocks. Bounds for more than two rounds are significantly higher, e.g., 2^{-158} for 3-round AEGIS-128 and 2^{-192} for 3-round AEGIS-256. Thus, we focus on the best case of two rounds and three blocks in the following.

Solving this model quickly yields an upper bound for the squared correlation contribution of 2^{-92} for AEGIS-128 and 2^{-116} for AEGIS-256, along with several truncated patterns meeting this bound. Trying to find compatible characteristics with bitwise masks, however, fails for all "optimal" and "near-optimal" results in the solution pool: they are apparently inconsistent. An exemplary invalid solution is listed in Figure 4.

For AEGIS-128L, the resulting bound for two rounds is 2^{-114}, and the best truncated characteristics are indeed consistent; a short heuristic search with the methods of Section 4 yields an example characteristic with squared correlation contribution 2^{-172}.

3.3 Improved Truncated Model

To understand why the solutions are inconsistent despite the apparent validity of the patterns for individual AES rounds, it is necessary to consider pairs of AES rounds either in the same AEGIS round or in consecutive rounds. Such cross-round inconsistencies have previously been observed in related-key differential characteristics for AES [GLMS18].

3.3.1 Limitations of the Simple Truncated Model

In the invalid example in Figure 4, two sources of conflicts are highlighted: One within the first round across substates in blue and green, the second within one substate across the first two rounds in yellow and green.

Consider the bitwise mask for the last column of $\alpha_{0,1} = \gamma_{0,1} = \gamma_{0,4} = \alpha_{0,4}$, highlighted in blue. Recall that for consistent linear characteristics, any three bitwise masks for the inputs and output of an Xor ($\oplus$) operation must be identical and that any three masks for the input and two outputs of a variable branch ($\bullet$) must Xor-sum to zero. Based on the definition of the output function $\mathcal{G}$ and the requirement for all-zero initial masks β_{-1}, these columns must have identical masks. Furthermore, since the corresponding last columns in $\sigma_{0,1}$ and $\sigma_{0,4}$ are both all-zero except for the first cell of $\sigma_{0,4}$, the bottom three cells of $\beta_{0,1}$ (green) and $\beta_{0,4}$ must also be identical. Their first cell, on the other hand, must be different. These columns correspond to the outputs of MixColumns in the first-round AES round functions. The corresponding input masks before MixColumns in the diagonals of $\rho_{0,0}$ and $\rho_{0,3}$ must be all-zero and thus identical in the bottom three cells, while the first cell may or may not be different.

We thus need a *pair of masks with difference* $(\texttt{**}, \texttt{00}, \texttt{00}, \texttt{00})$ at the input and a pair with difference $(\texttt{xx}, \texttt{00}, \texttt{00}, \texttt{00})$ at the output, where $\texttt{xx} \neq \texttt{00}$. This is, however, not possible due to the MDS property of the MixColumns matrix A [DR02]. Since A is MDS, all its square submatrices are invertible, implying that the transposed matrix A^T is also MDS and thus has a differential branch number of 5. This contradicts our requirements of mask differences with either 1 or 2 active cells.

The situation is similar for the yellow columns: The difference between the MixColumns output masks $\beta_{0,1}$ in the first round and $\beta_{1,1}$ in the second round must be precisely $\gamma_{1,1} \oplus \sigma_{1,1}$ and thus have exactly 2 active cells, while the corresponding input mask difference in the diagonal of $\rho_{0,0} \oplus \rho_{1,0}$ may have at most one active cell, contradicting the differential branch number of A^T.

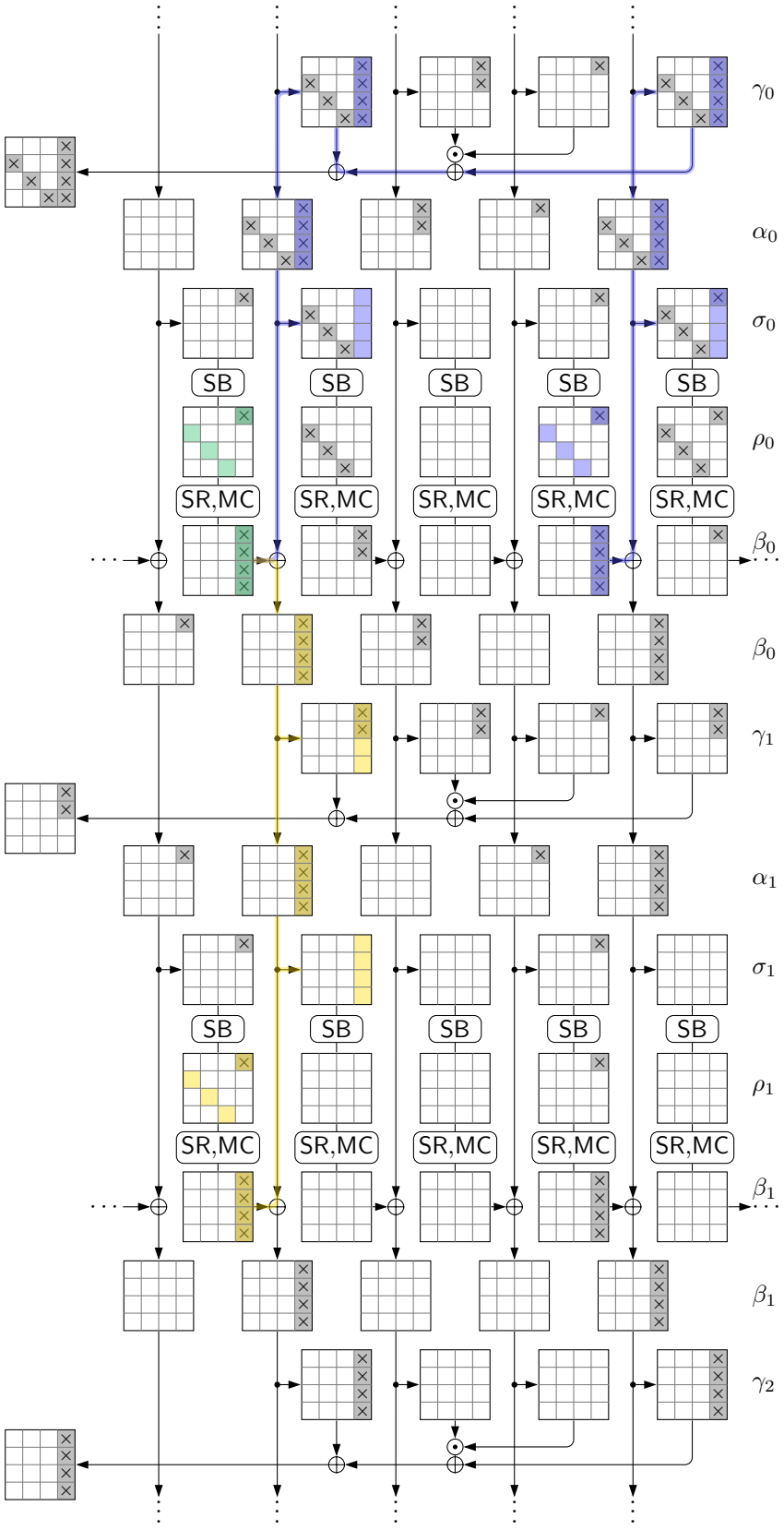


Figure 4: Inconsistent truncated linear characteristic for AEGIS-128 (11 S-boxes, 13 ANDs). Highlighted columns in blue+green or in yellow+green permit no bitwise consistent masks.

3.3.2 Additional Constraints for an Improved Truncated Model

More generally, the problem arises from linear constraints between the MixColumns output masks of a pair of AES rounds that are not reflected in the local truncated descriptions. In the following, we propose a refined MILP model with additional constraints to reflect the differential branch number for relevant pairs of masks. Such relevant pairs are linked by (undirected) paths with linear operations (Xor, branches), and their difference is constrained according to the difference patterns in those branches.

For each relevant pair, we introduce decision variables corresponding to the truncated difference between masks:

1. For the first active round (blue+green example): MixColumns input difference $\Delta\rho_{0,(0,3)} = \rho_{0,0} \oplus \rho_{0,3}$ (or, equivalently, $\Delta\mu_{0,(0,3)}$ after ShiftRows), output difference $\Delta\beta_{0,(1,4)} = \beta_{0,1} \oplus \beta_{0,4}$, constrained by the identical branch difference $\Delta\beta_{0,(1,4)} = \sigma_{0,1} \oplus \sigma_{0,4}$.

2. For each active round i (similarly to the blue+green example, but with the following output function): MixColumns input difference $\Delta\rho_{i,(0,3)} = \rho_{i,0} \oplus \rho_{i,3}$, output difference $\Delta\beta_{i,(1,4)} = \beta_{i,1} \oplus \beta_{i,4}$, constrained by the identical branch difference $\Delta\beta_{i,(1,4)} = \alpha_{i+1,1} \oplus \alpha_{i+1,4}$.

3. For substate j of two consecutive rounds $i, i+1$ (yellow+green example): MixColumns input difference $\Delta\rho_{(i,i+1),j} = \rho_{i,j} \oplus \rho_{i+1,j}$, output difference $\Delta\beta_{(i,i+1),j} = \beta_{i,j} \oplus \beta_{i+1,j}$, constrained by the identical branch difference $\Delta\beta_{(i,i+1),j} = \gamma_{i+1,j} \oplus \sigma_{i+1,j}$ (for $j \neq 0$) or $\Delta\beta_{(i,i+1),0} = \sigma_{i+1,0}$ (for $j = 0$).

4. For any two consecutive rounds $i, i+1$, we can combine the above constraints to obtain second-order differential constraints: The difference of two consecutive MixColumns input differences $\Delta^2\rho_{(i,i+1),(0,3)} = \Delta\rho_{(i,i+1),0} \oplus \Delta\rho_{(i,i+1),3} = \Delta\rho_{i,(0,3)} \oplus \Delta\rho_{i+1,(0,3)}$ and output differences $\Delta^2\beta_{(i,i+1),(1,4)} = \Delta\beta_{(i,i+1),1} \oplus \Delta\beta_{(i,i+1),4} = \Delta\beta_{i,(1,4)} \oplus \Delta\beta_{i+1,(1,4)}$, constrained by the identical branch difference $\Delta^2\beta_{(i,i+1),(1,4)} = \sigma_{i+1,1} \oplus \sigma_{i+1,4}$, since $\gamma_{i+1,1} \oplus \gamma_{i+1,4} = 0$.

In the truncated MILP model, both the required MDS branch number for the MixColumns input and output difference variables and the Xor-based definition of the differences based on the original masks are only modeled by their branch numbers (5 for differential MixColumns, 2 for Xor). These can be translated to MILP constraints in exactly the same way as the linear models of MixColumns and branches in Subsection 3.2, using yet another set of activity helper variables.

In total, for the two-round model of AEGIS-128, this adds $16 \times 2 \times (1+2+5+1) = 288$ new binary variables for the differences plus slightly less than $(4 + 16 \times 3) \times (1 + 2 + 5) + (4 + 16 \times 5) \times 1 = 500$ new binary helper variables for MixColumns and Xors, as well as about 2×500 linear constraints. This is in addition to the 1741 variables and 1610 constraints needed for the simple truncated model. For the larger variants AEGIS-256 and AEGIS-128L, the conditions can be easily adapted to the extended state size and different output functions; for example, for AEGIS-256, the Xor of three substates appears in the output function, so the conditions for substate pair $(1, 4)$ above need to be applied for all pairs $(1, 4)$, $(1, 5)$, and $(4, 5)$. The objective function remains unchanged.

3.3.3 Results for the Improved Model

Solving this refined truncated model improves the upper bound for the squared correlation contribution to 2^{-102} for 2 rounds of AEGIS-128. The resulting optimal truncated solutions appear to be consistent, and corresponding bitwise characteristics can be found easily (see Section 4). On the downside, there is a significant gap between this bound and the best

found bitwise characteristic, whose squared correlation contribution is 2^{-140}. For 2 rounds of AEGIS-256, the bound is slightly improved to 2^{-120}, but the obtained optimal solution still appears to be inconsistent.

3.4 Bitwise Model

Although the best truncated linear characteristics for AEGIS-128 found with the improved truncated model of Subsection 3.3 are consistent, there is a significant gap between the resulting bounds and the best found solutions. Most notably, while the bound suggests potential attacks with complexity below 2^{128}, the best found attacks are above this bound. The main reason for this gap is not the correlation of non-optimal S-box transitions, which are only slightly worse than the optimal ones, but the contribution of the output function and its AND-gates. While the bound assumes the optimal case of only 1 active bit per byte and thus a squared correlation of 2^{-2}, this may rise up to all 8 active bits and a squared correlation of 2^{-16}.

To improve the tightness of the bound, we chose to include a full bitwise model of the linear operations in the MILP model alongside the truncated constraints. The goal of this model is to capture constraints on the number of active bits in the nonlinear output functions and thus better estimate their correlation. Running the truncated and non-truncated models in parallel allows the MILP solver to find better bounds faster.

Linked Truncated and Bitwise Models We propose a partial bitwise model of the state update function to model each bit of the masks using MILP. For each binary decision variable corresponding to a truncated mask, we add 8 binary decision variables to model the activity of each bit, i.e., whether the mask is 0 or 1. For example, the truncated variable $\bar{\alpha}_{i,j}[r,c]$ is connected with its corresponding individual bitwise mask variables $\alpha_{i,j}[r,c,b]$, $b \in \mathcal{B}$, as follows:

$$\bar{\alpha}_{i,j}[r,c] \;\leq\; \sum_{b\in\mathcal{B}} \alpha_{i,j}[r,c,b] \;\leq\; 8\,\bar{\alpha}_{i,j}[r,c] \qquad \forall i \in \mathcal{I}, j \in \mathcal{J}, r \in \mathcal{R}, c \in \mathcal{C}\,.$$

Simple variable renaming steps, such as ShiftRows or the mask equality constraints corresponding to XOR operations, can be directly applied to the mask variables in the same fashion as for the truncated variables.

Modeling AES SubBytes Although there has been some work done regarding the modeling of large S-boxes in MILP [AST+17], such approaches are impractical for our AEGIS model due to the sheer number of S-boxes involved, combined with the density of the Linear Approximation Table (LAT) of the AES S-box. Consequently, we simply do not add any additional constraints for the S-box transitions and assume that any S-box transition is possible with the optimal squared correlation of 2^{-6}. Indeed, about 93 % of all transitions are possible for the AES S-box, with squared correlation between 2^{-6} and 2^{-12}.

Modeling AES MixColumns The MixColumns step is linear over $\mathbb{F}_2$ and can thus be represented as a matrix-vector multiplication over $\mathbb{F}_2$. We denote the corresponding 32×32 matrix as A. Thus if we denote the input of AES MixColumns as m and the output as b, with 32-bit columns denoted by m_c and b_c, respectively, then $\forall c \in \mathcal{C}:\ b_c = A\,m_c$. The corresponding columns of our linear masks are:

$$\begin{aligned}
\mu_{i,j}[*,c] &:= \mu_{i,j}[0,c] \parallel \mu_{i,j}[1,c] \parallel \mu_{i,j}[2,c] \parallel \mu_{i,j}[3,c] && \text{for } m_c\,,\\
\beta_{i,j}[*,c] &:= \beta_{i,j}[0,c] \parallel \beta_{i,j}[1,c] \parallel \beta_{i,j}[2,c] \parallel \beta_{i,j}[3,c] && \text{for } b_c\,.
\end{aligned}$$

Then the linear masks must satisfy:

$$\mu_{i,j}[*,c] = A^T \beta_{i,j}[*,c] \qquad \forall i \in \mathcal{I}, j \in \mathcal{J}, c \in \mathcal{C}\,,$$

where the matrix multiplication is over $\mathbb{F}_2$. This way, we obtain Xor-equations involving either 6 or 12 variables. To model these Xor-equations as $\mathbb{R}$-linear equations for MILP, we simply require that the integer sum of all involved variables equals $2 \cdot d$ for a new integer dummy variable d.

Modeling Branches and the Output Function $\mathcal{G}$ The bitwise model of And operations as well as the equality constraints corresponding to Xor operations are analogous to the truncated model described in Section 3.2.1. The Xor constraints corresponding to variable branches can be modeled the same way as the Xor equations in MixColumns, i.e., by using dummy variables. There is, however, a much better model possible: Similar to the truncated model we can disallow all invalid transitions using constraints. Thus for each triplet of bits (a, b, c) and for each branch we require:

$$a + b + c \leq 2\,, \qquad b + c \geq a\,, \qquad a + c \geq b\,, \qquad a + b \geq c\,.$$

By applying this alternative model to all branches in the bitwise model as well as in the linked truncated model, we eliminate 3 600 dummy variables for two rounds of AEGIS-128. This reduced the solving runtime for AEGIS-128 from approximately 3 days to 25 minutes in a multi-core setup.

This speedup is explained by the way the branch and bound algorithm works: It first removes all integrality constraints and solves this so-called LP relaxation. This solution provides a first bound for the problem. Of course there may still be integer variables with non-integer values; the solver now picks one variable v with value x of them and branches on that variable i. e. it creates two subproblems one with the additional constraint that $v \leq \lfloor x \rfloor$ and the other with the constraint $v \geq \lceil x \rceil$. If an integer solution is found, it is optimal for that subtree. This is repeated until all subtrees have either been exhausted or cut (because their bound is worse than the current best solution) and only one (integer optimal) solution remains. By using the alternative model, we can capture the branch constraints in the LP relaxation and thus the algorithm needs to branch much less and finds better bounds faster.

Objective Function For the objective function, we keep the truncated bounds for the S-box transitions, but evaluate the exact bitwise cost of approximating the And operations in the output function based on the output masks λ:

$$\text{minimize} \quad 6 \sum_{\substack{i \in \mathcal{I} \\ j \in \mathcal{J} \\ r \in \mathcal{R} \\ c \in \mathcal{C}}} \sigma_{i,j}[r,c] + 2 \sum_{\substack{i \in \mathcal{I}' \\ r \in \mathcal{R} \\ c \in \mathcal{C} \\ b \in \mathcal{B}}} \lambda_i[r,c,b]\,.$$

3.4.1 Results for the Bitwise Model

Solving the bitwise model is significantly more costly than the truncated model, but also produces significantly tighter bounds: While the truncated and improved models can be solved instantaneously by `Gurobi`, the bitwise model takes about 20 minutes for AEGIS-128, 1 hour and 20 minutes for AEGIS-256, and about 5 minutes for AEGIS-128L to find an optimal solution in a multi-core setup. The resulting upper bound for the squared correlation contribution of 2 rounds of AEGIS-128 is 2^{-132} and thus very close to the best found solution with 2^{-140} (using the LAT details, see Section 4). For 2-round AEGIS-256, the bound of 2^{-152}, obtained after running Gurobi for almost a week on 32 cores (with the original Xor model), is similarly close to the best found solution with 2^{-162}, as is the bound of 2^{-140} for 2-round AEGIS-128L compared to 2^{-152}.

4 Improved Linear Approximations of AEGIS

So far, we focused on deriving bounds on the maximum possible squared correlation contribution of characteristics. In this section, we discuss our search for solutions with close-to-optimal quality and discuss the estimated cost of resulting attacks in more detail. We first introduce a suitable Constraint Programming (CP) model based on the results of Section 3 in Subsection 4.1 and then present our results in Subsection 4.2, including characteristics with squared correlation contribution within factors 2^8 to 2^{12} of the derived bounds. In Subsection 4.3, we describe our efforts in experimentally verifying part of these results, as well as the derived estimates on the attack complexity.

4.1 Constraint Programming Model

We model the constraints for valid linear characteristics in the CP tool Z3, including a bitwise model for the linear operations similar to the one of Subsection 3.4 and a model and cost definition of all (or some) of the valid transitions as described by the Linear Approximation Table (LAT) of the AES S-box.

In addition to classical (hard) constraints we use weighted soft constraints: These may be violated, but if they are, they incur a weight. The solver then tries to minimize the total weight. We include soft constraints with the corresponding weights that try to exclude the more costly S-box transitions, while hard constraints forbid impossible transitions. Additionally, we include soft constraints for the outputs of the And-gates.

We split the linear approximation table of the S-box into multiple tables in such a way that we end up with one table per correlation value. We then add soft constraints for each correlation value of each S-box with corresponding weights.

Since the LAT is large and dense – out of 65536 possible mask pairs, 60946 are valid, with 9 different correlations –, we quickly run into unmanageable runtimes with our model. Therefore, we tweak the parameters of our model in two different ways: In one model we include only the two best (non-zero) transition classes of the S-boxes. In the other model we include the five best (non-zero) transition classes of S-boxes but remove the optimization goal (i.e., the exact correlation) for the S-box class. To compensate for the missing optimization goal, we generate many solutions in a loop and pick the best one.

Additionally, we define starting constraints based on minor variations of the best truncated linear characteristics identified by the MILP solver and only include the LAT constraints for S-boxes that are active in the truncated characteristic. Such reductions of the search space are necessary for manageable runtimes, but of course, they imply that the resulting solutions are not necessarily globally optimal, and slightly better solutions may still exist.

4.2 Results

For AEGIS-128 we allow the 14 535 LAT transitions with squared correlation $c^2 \geq 2^{-6.83}$ and do not distinguish between them. After 13 minutes we obtained the linear characteristic with squared correlation contribution of 2^{-140} depicted in Figure 5. Generating many solutions in a loop did not provide any better characteristics.

For AEGIS-256 we allow the 20 655 LAT-transition with squared correlation $c^2 \geq 2^{-7.36}$ and do not distinguish between them. We generate many solutions in a loop and pick the best one with squared correlation contribution of 2^{-162}, which is depicted in Figure 6.

For AEGIS-128L we allow the 14 535 LAT transitions with squared correlation $c^2 \geq 2^{-6.83}$ and do not distinguish between them. By generating many solutions for about 2 days, we are able to pick the best of these solutions with squared correlation contribution of 2^{-152}. It is depicted in Figure 7.

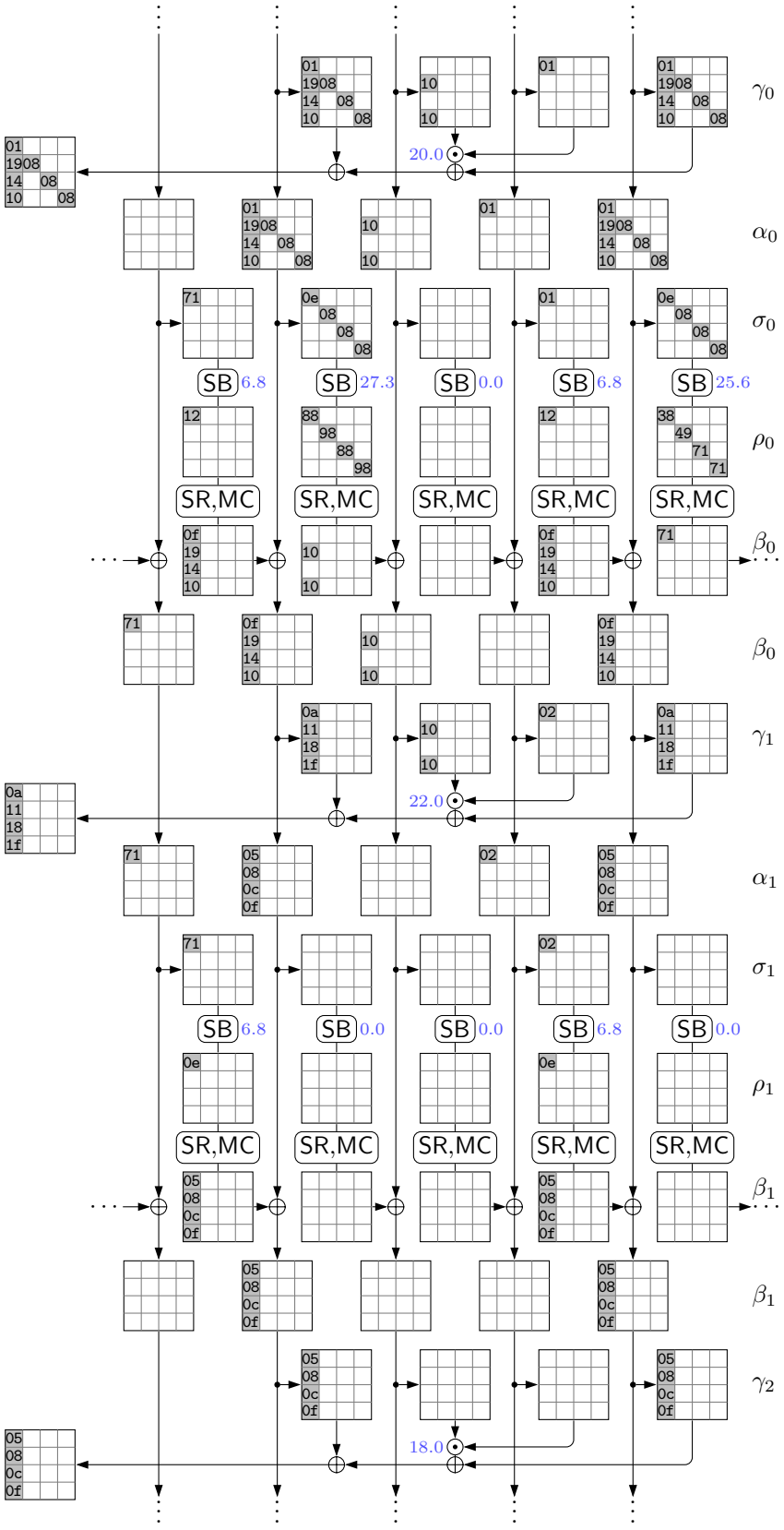


Figure 5: Linear approximation for AEGIS-128 with squared correlation contribution 2^{-140}.

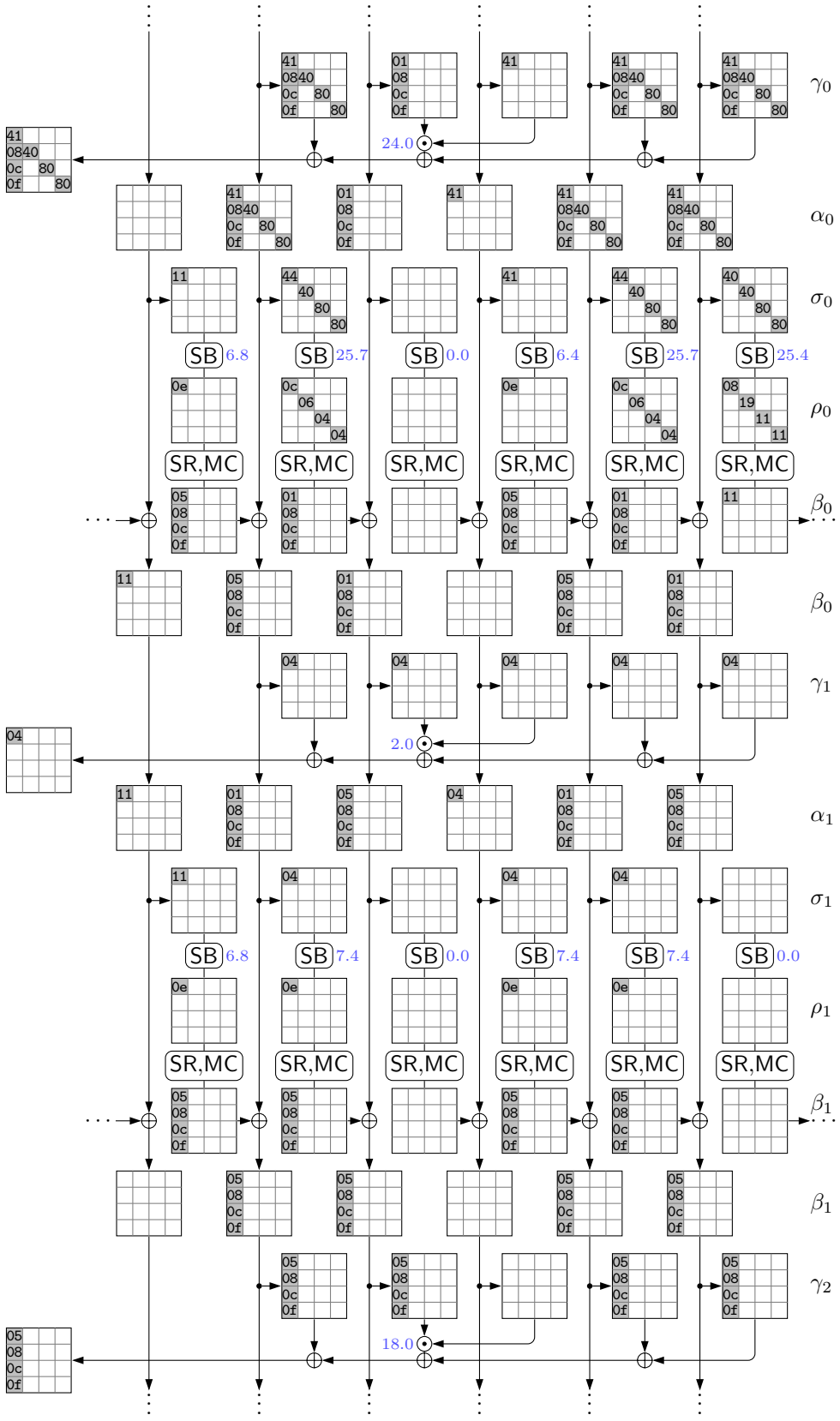


Figure 6: Linear approximation for AEGIS-256 with squared correlation contribution 2^{-162}.

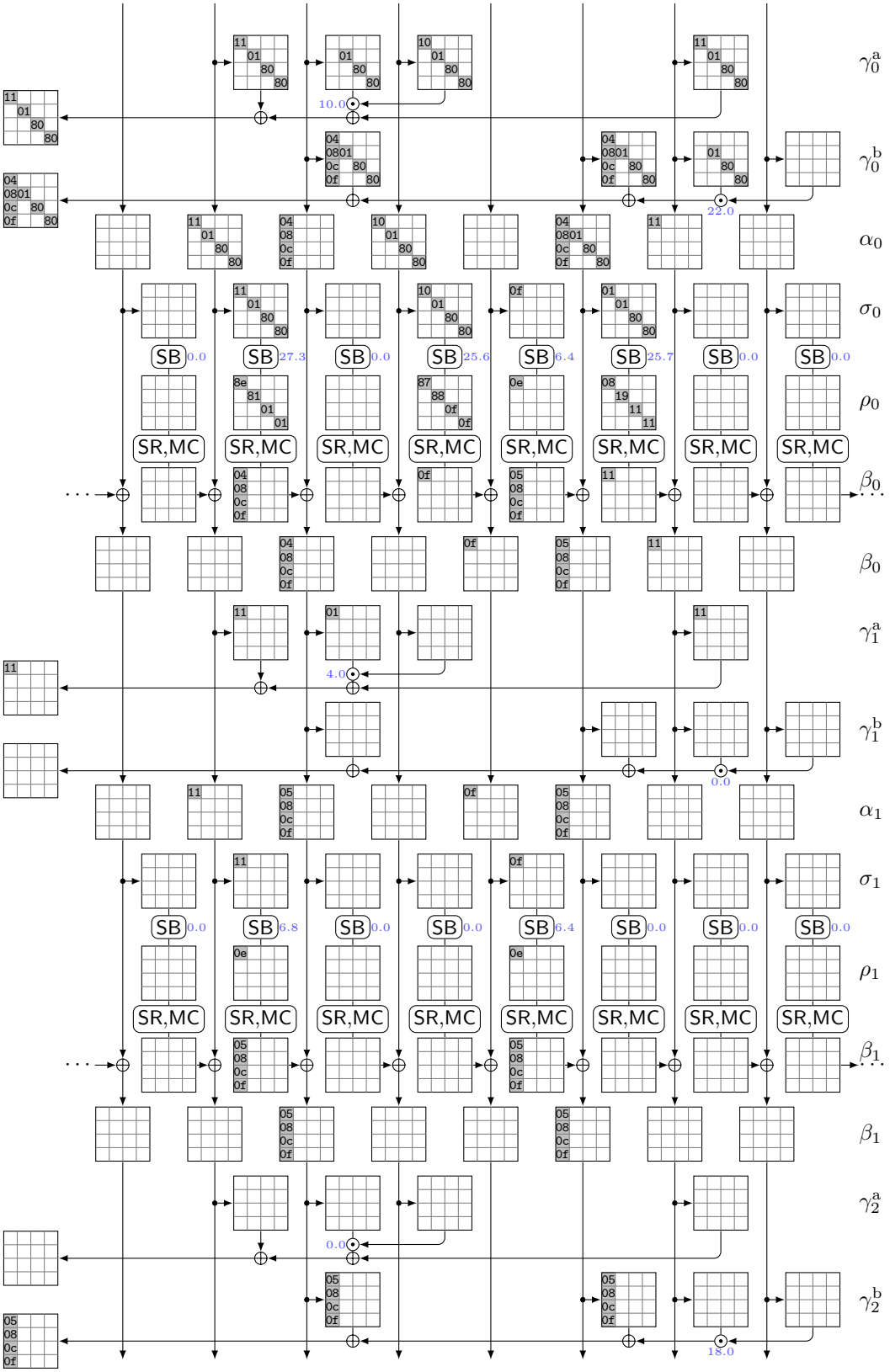


Figure 7: Linear approximation for AEGIS-128L with squared correlation contrib. 2^{-152}.

The model with two possible classes of S-box transitions and corresponding optimization targets takes multiple hours to solve. The characteristics generated by this model were superseded by characteristics identified by running the quicker model in a loop. Runtime numbers use an AMD Ryzen 2700X with a single core each due to limitations of Z3.

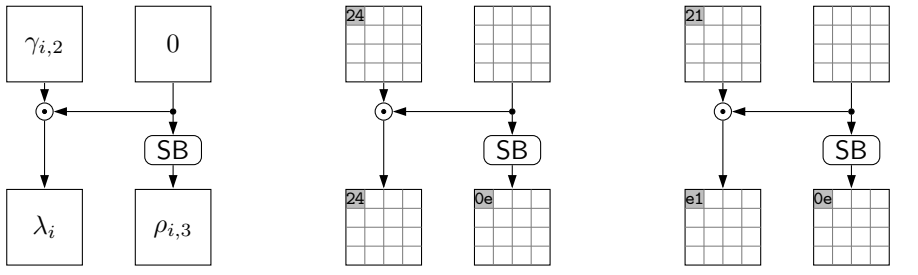
We remark that all three truncated characteristics share some structural patterns; particularly the characteristics for AEGIS-128 and AEGIS-256 are quite similar. They are, however, quite different from the approximations found by Minaud [Min14], which involve only the first and last keystream blocks Z_0 and Z_2, while the mask for Z_1 is zero.

4.3 Experimental Verification

We experimentally verified the squared correlation of parts of these characteristics. For AEGIS-128, we first split the approximation corresponding only to the state update function $\mathcal{F}$ into two "vertical" parts and verified them separately based on the AEGIS reference implementation, within a continuous keystream (without re-initialization): The first part covers the characteristic for substates with indices $j \in \{0, 1, 2\}$, with a predicted squared correlation contribution of $2^{-41.075}$ and a measured result of $2^{-40.490}$ using 2^{44} samples. The second part covers $j \in \{3, 4\}$ with an expected contribution $2^{-39.27}$ and a measured result of $2^{-38.4}$ using 2^{43} samples.

Second, we split the characteristic into three horizontal parts, each corresponding to one output function call $\mathcal{G}$ and (the S-box layer of) one state update call $\mathcal{F}$. In particular, this addresses dependencies related to the consecutive AND-operation and S-box approximations of the same byte, which occurs in the first byte of substate $j = 3$ in each active round of the characteristics for both AEGIS-128 and AEGIS-256, as well as in the first diagonal of substate $j = 3$ in the first round for AEGIS-128L. In all these cases, the S-box input masks must be identical to the masks for one AND-input, so their mask value is irrelevant for the overall correlation, as illustrated in Figure 8. The characteristics in Figures 5, 6, 7 define specific identical input masks for the S-box and one AND input and thus treat the two operations like separate, independent subfunctions of the characteristic, while the combined gadget takes the dependencies of the shared input and the resulting linear hull effect into account with a zero mask on the shared input. An exhaustive evaluation of all possible byte mask values for $\rho_{i,2}$, λ_i, and $\rho_{i,3}$ with all 2^{16} possible input values shows that the best possible combined squared correlation is $2^{-7.36}$ and takes about 20 minutes on a 44-core CPU with a fairly unoptimized search implementation. The best found squared correlation is slightly, but not drastically better than the bound of 2^{-2-6} based on the individual squared correlation contributions of AND and the S-box. Still, there may be more significant deviations from the expected correlation for specific mask values. Furthermore, lower values may be possible in other characteristics where $\beta_{i-1,3} \oplus \beta_{i,3} \neq 0$.

The two examples in Figure 8 are from an alternative characteristic found for the same



(a) $\beta_{i-1,3} \oplus \beta_{i,3} = 0$ (b) $2^{-7.66}$ instead of $2^{-4-6.39}$ (c) 0 instead of $2^{-8-6.39}$

Figure 8: Examples for characteristic parts with different combined squared correlation.

truncated characteristic of AEGIS-256. Their measured correlation deviates significantly: it is much better in the first example (from the second round), where the combined squared correlation of AND and S-box in byte 0 of $S_{1,4}$ is $2^{-7.66}$ instead of the predicted $2^{-10.39}$, and thus even higher than the upper bound of 2^{-2-6} for the product of the individual squared correlation contributions. By choosing different S-box input masks consistent with the characteristic, the estimate for the same gadget could be as bad as 2^{-16}. On the other hand, in the second example (first round), the selected bits in the same byte position are not correlated at all. Since the latter occurred in several example solutions produced by Z3, we added the verification of the gadget as an output condition when searching for good solutions, which only takes a couple of minutes per found solution. It is also possible to constrain the masks involved in gadgets to a list of values corresponding to correlations above a particular bound, similar to the S-box constraints in Z3.

The bounds we provided cannot completely take such linear hull effects into account, as discussed in Subsubsection 3.4.1. The truncated models will be less affected than the bitwise models, since they already approximate each such gadget with 2^{-8}, which is not too different from $2^{-7.36}$. For the bitwise models, larger deviations may occur for characteristics with higher Hamming weights of the output masks, but a potentially high correlation in the gadget. Since the objective function discourages high Hamming weights and the S-box is always bounded by the optimal 2^{-6}, we expect that the deviation from the bound is not too large. Of course, there may also be other linear hull effects that could allow approximations with a better correlation.

For the characteristics in Figures 5, 6, 7, the observed correlation quite closely confirms the expectation based on the correlation contributions: For AEGIS-128 (Figure 5), due to the Hamming weight of 1 in λ_i, the S-box input difference specified by the characteristic is the only possible one, so the evaluation precisely confirmed the squared correlation contribution of 2^{-140}. For AEGIS-256 (Figure 6), the same is true in the second round; in the first round with a Hamming weight of 2, there are 3 possible masks and the observed squared correlation is slightly better at 2^{-10} instead of $2^{-10.39}$. For AEGIS-128L (Figure 7), byte 0 in the first round is also slightly better at $2^{-9.36}$ instead of $2^{-10.83}$ (3 masks).

When using these characteristics for attacks, the data complexity of about c^{-2} can be improved, similar to Minaud's, by combining multiple approximations: for example, the characteristics can be shifted to 16 different anchor positions for the byte at the intersection of the diagonal and the column in Z_0, and can be evaluated in an overlapping way on the keystream (though this may introduce additional dependencies).

5 Conclusion

We proposed improved keystream approximations for the AEGIS family, but also proved upper bounds for the squared correlation contribution of any single suitable linear characteristic. All bounds are below 2^{-128} and thus support AEGIS' security with realistic amounts of data. Still, these bounds should be taken with a grain of salt and do not necessarily prove security against linear cryptanalysis using keystream approximations in general: First, the bounds apply to individual characteristics, and better approximations may exist due to the linear hull effect. However, we expect this effect to be limited, since the fixed λ_i masks in each round limit the number of potentially compatible characteristics. Second, and much more importantly, the inputs to individual nonlinear operations are definitely not independent. There is no key, and in particular, $\mathcal{G}$'s AND-operations share the exact same input values with some of $\mathcal{F}$'s S-boxes. This effect was also observed in Minaud's analysis [Min14], where it improved the attack complexity.

References

[ABP+13] Nadhem J. AlFardan, Daniel J. Bernstein, Kenneth G. Paterson, Bertram Poettering, and Jacob C. N. Schuldt. On the security of RC4 in TLS. In *USENIX Security Symposium 2013*, pages 305–320. USENIX Association, 2013.

[AEL+18] Tomer Ashur, Maria Eichlseder, Martin M. Lauridsen, Gaëtan Leurent, Brice Minaud, Yann Rotella, Yu Sasaki, and Benoît Viguier. Cryptanalysis of MORUS. In Thomas Peyrin and Steven D. Galbraith, editors, *Advances in Cryptology – ASIACRYPT 2018*, volume 11273 of *LNCS*, pages 35–64. Springer, 2018.

[AST+17] Ahmed Abdelkhalek, Yu Sasaki, Yosuke Todo, Mohamed Tolba, and Amr M. Youssef. MILP modeling for (large) s-boxes to optimize probability of differential characteristics. *IACR Transactions on Symmetric Cryptology*, 2017(4):99–129, 2017.

[Ava17] Roberto Avanzi. The QARMA block cipher family. almost MDS matrices over rings with zero divisors, nearly symmetric even-mansour constructions with non-involutory central rounds, and search heuristics for low-latency S-boxes. *IACR Transactions on Symmetric Cryptology*, 2017(1):4–44, 2017.

[Ber18] Daniel J. Bernstein. Announcement of the CAESAR finalists. Talk at FSE 2018 rump session, 2018.

[BJK+16] Christof Beierle, Jérémy Jean, Stefan Kölbl, Gregor Leander, Amir Moradi, Thomas Peyrin, Yu Sasaki, Pascal Sasdrich, and Siang Meng Sim. The SKINNY family of block ciphers and its low-latency variant MANTIS. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology – CRYPTO 2016*, volume 9815 of *LNCS*, pages 123–153. Springer, 2016.

[CAE19] CAESAR Committee. CAESAR: Competition for authenticated encryption: Security, applicability, and robustness, 2013–2019.

[CHP+17] Carlos Cid, Tao Huang, Thomas Peyrin, Yu Sasaki, and Ling Song. A security analysis of Deoxys and its internal tweakable block ciphers. *IACR Transactions on Symmetric Cryptology*, 2017(3):73–107, 2017.

[DGV94] Joan Daemen, René Govaerts, and Joos Vandewalle. Correlation matrices. In Bart Preneel, editor, *Fast Software Encryption – FSE 1994*, volume 1008 of *LNCS*, pages 275–285. Springer, 1994.

[DR98] Joan Daemen and Vincent Rijmen. The block cipher Rijndael. In *CARDIS*, volume 1820 of *Lecture Notes in Computer Science*, pages 277–284. Springer, 1998.

[DR02] Joan Daemen and Vincent Rijmen. *The Design of Rijndael: AES – The Advanced Encryption Standard*. Information Security and Cryptography. Springer, 2002.

[GLMS18] David Gérault, Pascal Lafourcade, Marine Minier, and Christine Solnon. Revisiting AES related-key differential attacks with constraint programming. *Information Processing Letters*, 139:24–29, 2018.

[IOWM13] Takanori Isobe, Toshihiro Ohigashi, Yuhei Watanabe, and Masakatu Morii. Full plaintext recovery attack on broadcast RC4. In Shiho Moriai, editor, *Fast Software Encryption – FSE 2013*, volume 8424 of *LNCS*, pages 179–202. Springer, 2013.

[KEM17] Daniel Kales, Maria Eichlseder, and Florian Mendel. Note on the robustness of CAESAR candidates. IACR Cryptology ePrint Archive, Report 2017/1137, 2017.

[Mat93] Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In Tor Helleseth, editor, *Advances in Cryptology – EUROCRYPT '93*, volume 765 of *LNCS*, pages 386–397. Springer, 1993.

[Min14] Brice Minaud. Linear biases in AEGIS keystream. In Antoine Joux and Amr M. Youssef, editors, *Selected Areas in Cryptography – SAC 2014*, volume 8781 of *LNCS*, pages 290–305. Springer, 2014.

[MWGP11] Nicky Mouha, Qingju Wang, Dawu Gu, and Bart Preneel. Differential and linear cryptanalysis using Mixed-Integer Linear Programming. In Chuankun Wu, Moti Yung, and Dongdai Lin, editors, *Information Security and Cryptology – Inscrypt 2011*, volume 7537 of *LNCS*, pages 57–76. Springer, 2011.

[SHW+14] Siwei Sun, Lei Hu, Peng Wang, Kexin Qiao, Xiaoshuang Ma, and Ling Song. Automatic security evaluation and (related-key) differential characteristic search: Application to SIMON, PRESENT, LBlock, DES(L) and other bit-oriented block ciphers. In Palash Sarkar and Tetsu Iwata, editors, *Advances in Cryptology – ASIACRYPT 2014*, volume 8873 of *LNCS*, pages 158–178. Springer, 2014.

[SSS+19] Danping Shi, Siwei Sun, Yu Sasaki, Chaoyun Li, and Lei Hu. Correlation of quadratic boolean functions: Cryptanalysis of all versions of full MORUS. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology – CRYPTO 2019*, volume 11693 of *LNCS*, pages 180–209. Springer, 2019.

[TIM+18] Yosuke Todo, Takanori Isobe, Willi Meier, Kazumaro Aoki, and Bin Zhang. Fast correlation attack revisited – cryptanalysis on full Grain-128a, Grain-128, and Grain-v1. In Hovav Shacham and Alexandra Boldyreva, editors, *Advances in Cryptology – CRYPTO 2018*, volume 10992 of *LNCS*, pages 129–159. Springer, 2018.

[VV18] Serge Vaudenay and Damian Vizár. Can Caesar beat Galois? – robustness of CAESAR candidates against nonce reusing and high data complexity attacks. In Bart Preneel and Frederik Vercauteren, editors, *Applied Cryptography and Network Security – ACNS 2018*, volume 10892 of *LCNS*, pages 476–494. Springer, 2018.

[WP13] Hongjun Wu and Bart Preneel. AEGIS: A fast authenticated encryption algorithm. In Tanja Lange, Kristin E. Lauter, and Petr Lisonek, editors, *Selected Areas in Cryptography – SAC 2013*, volume 8282 of *LNCS*, pages 185–201. Springer, 2013.

[WP16] Hongjun Wu and Bart Preneel. AEGIS: A fast authenticated encryption algorithm (v1.1). Submission to CAESAR: Competition for Authenticated Encryption. Security, Applicability, and Robustness (Round 3 and Final Portfolio), September 2016. http://competitions.cr.yp.to/round3/aegisv11.pdf.