V. A. Kiryukhin, Алгоритм вычисления верхних оценок для дифференциалов не минимального веса в двухраундовых LSX-шифрах,
Матем. вопр. криптогр., 2021, том 12, выпуск 2, 93–109

МАТЕМАТИЧЕСКИЕ ВОПРОСЫ КРИПТОГРАФИИ
2021 Т. 12 № 2 С. 93–109

УДК 519.719.2 DOI https://doi.org/10.4213/mvk358

An algorithm for computing the upper bound for non-minimum weight differentials in 2-round LSX-ciphers

V. A. Kiryukhin

JSC «InfoTeCS», LLC «SFB Lab», Moscow

Получено 18.XI.2020

Abstract. We describe some approaches to upper bounding the non-minimum weight differentials (EDP) and linear hulls (ELP) in 2-round LSX-cipher. We propose a dynamic programming algorithm to solve this problem. For 2-round Kuznyechik the nontrivial upper bounds on all differentials (linear hulls) with 18 and 19 active S-boxes are obtained. These estimates are also holds for other differentials (linear hulls) with a larger number of active S-boxes.

Keywords: Kuznyechik, SPN, LSX, differential cryptanalysis, MEDP, MELP

Алгоритм вычисления верхних оценок для дифференциалов не минимального веса в двухраундовых LSX-шифрах

В. А. Кирюхин

ОАО «ИнфоТеКС», ООО «СФБ Лаб», Москва

Аннотация. Рассматриваются подходы к вычислению верхних оценок для характеристик дифференциалов (EDP) и суммарных линейных соотношений (ELP) не минимального веса в двухраундовых LSX-шифрах. Для решения этой задачи предложен алгоритм динамического программирования. С его помощью для двух раундов шифра Кузнечик получены нетривиальные верхние оценки характеристик дифференциалов (суммарных линейных соотношений), содержащих 18 и 19 активных подстановок. Полученные оценки справедливы также для дифференциалов (суммарных линейных соотношений), содержащих большее число активных подстановок.

Ключевые слова: Кузнечик, SPN, LSX, дифференциальный криптоанализ, MEDP, MELP

Citation: Matematicheskie Voprosy Kriptografii, 2021, v. 12, № 2, pp. 93–109 (Russian)

1. Introduction

Differential [2] and linear [3] cryptanalysis are the two most known statistical attacks applicable to block ciphers. In this paper we will focus on the first method. The analogous results for linear cryptanalysis may be obtained in a similar way due to the existience of well-known duality [4].

There are several approaches to estimate the security of ciphers against differential attacks. In many papers the probabilistic characteristics of differential trails were studied. The maximal value of such characteristics (EDCP) decreases when the number of active S-boxes within R rounds increases. The upper bound on such characteristics may be analytically obtained for many LSX-ciphers (AES [11], Khazad [12], Kuznyechik [1], etc.). In particular, these results are presented in [11, 17].

However, many researchers note that differential cryptanalysis exploits differentials and not differential trails (see for example [5, 14, 16]). The expected probabilistic characteristic (EDP) of a differential $(\Delta x, \Delta y)$ corresponds to the sum of characteristics of all trails with input difference Δx and output difference Δy [8]. So from this point of view the security of a cipher against differential attacks is based on the maximum expected differential probability (MEDP) over $R \geqslant 2$ rounds.

To complete the picture, we note that the cipher transformations were considered in the widely used so-called «Markov model» [8] (all round keys in LSX-cipher are supposed to be independent and uniformly distributed). The reader is referred to [18] for a critical view on this assumption.

Related works. For 2-round LSX-ciphers, some approaches to the computation of upper bounds for the MEDP are known [13–15].

An algorithm for computing the exact MEDP of 2-round AES was proposed in [5]. In [10] the upper bounds for the MEDP for so-called «nested» LSX-ciphers (e.g. 4-round AES) are given.

In [16] it was shown that for some 2-round LSX-ciphers the MEDP is realized by differentials involving a number of active S-boxes which exceeds the branch number of the linear layer (non-minimum weight differentials).

Some results on the differential properties of 2-round Kuznyechik were obtained in [19]. The cited paper contains an algorithm for constructing the best minimum weight differentials and a proof that all other differentials have a lower EDP. Thanks to these two results, the exact value of the 2-round MEDP was computed.

Our contribution. We propose a dynamic programming algorithm designed to bound the non-minimum-weight differentials in 2-round LSX-ciphers. It uses only the difference distribution table and the differential

branch number of the linear layer. The algorithm minimizes the number of high probability differential trails and does not try to minimize the total number of trails. As a result, the algorithm is not effective for ciphers with small block size (for example, 32-bit 2-round AES).

We have applied the developed algorithm to the 2-round Kuznyechik (Section 4): the probability for any 2-round differential (linear hull) with $n+3 = 19$ active S-boxes is upper bounded by $2^{-88.34}$ ($2^{-79.63...}$) correspondingly. These bounds also holds for any differential (linear hull) with $a \geqslant n+3$ active S-boxes. Similar results were obtained for 2-round Khazad in [21], and the exact values of MEDP $= 2^{-45} + 2^{-60}$ and MELP $= 2^{-37.80...}$ for Khazad were also proved.

The set of estimates obtained may be used in further researches to calculate the bounds on the MEDP (MELP) for more rounds. We plan to use our new results together with a modified KMT2-DC (KMT2) algorithm [6, 7]. The approach used in [7] allows to incorporate other upper bounds when those bounds are superior to the values determined directly by the original algorithm [6]. In this way, we hope to improve security estimates of Kuznyechik with respect to the differential and linear cryptanalysis.

2. Notation and definitions

An LSX cipher E consists of sequence of rounds. Each of them contains three operations: X – modulo 2 addition of an input block with an iterative key, S – parallel application of a fixed bijective substitution s, L – linear transformation which may be represented as multiplication by the binary matrix.

To simplify the text and notation we consider only byte-oriented LSX-ciphers.

Let us denote:
n – block size in bytes,
$\oplus$ – bitwise XOR operation,
$v[i]$ – i-th element of vector or sequence v, $1 \leqslant i \leqslant l$, where l is the number of elements of v,
$\mathrm{Supp}(v) = \{i\colon v[i] \neq 0\}$ – the support of a vector v,
$\mathrm{wt}(v) = \#\{i\colon v[i] \neq 0\}$ – the weight of a vector v,
$\mathbf{F}_q$ – finite field of q elements,
$\mathbf{F}_q^*$ – set of all nonzero elements of the field $\mathbf{F}_q$,
$\mathbf{F}_q^l$ – set of l-element vector over $\mathbf{F}_q$.
Depending on the context, we will interpret a value $z \in \overline{0, 2^l - 1}$ as element of $\mathbf{F}_{2^l}$ or $\mathbf{F}_2^l$ or as an integer.

Definition 1. Let $\mathsf{s}\colon \mathbf{F}_2^8 \to \mathbf{F}_2^8$, let $a, b \in \mathbf{F}_2^8$ be fixed, and let x be a random variable having uniform distribution on $\mathbf{F}_2^8$. The differential probability of (a, b) is defined as

$$\mathrm{DP}\,(a, b) = \mathbf{P}_x\left(\mathsf{s}(x) \oplus \mathsf{s}(x \oplus a) = b\right).$$

Definition 2. Let E be a cipher with key-size κ and block-size l. Let x be a random variable having uniform distribution on $\mathbf{F}_2^l$. Then the expected (over keys K) differential probability of $(\Delta x, \Delta y)$ is defined as

$$\mathrm{EDP}\,(\Delta x, \Delta y) = 2^{-\kappa} \sum_{K \in \mathbf{F}_2^\kappa} \mathbf{P}_x\left(E_K(x) \oplus E_K(x \oplus \Delta x) = \Delta y\right),$$

where E_K is a cipher with key K.

Definition 3. The maximum expected differential probability is

$$\mathrm{MEDP} = \max_{\Delta x \neq 0, \Delta y} \mathrm{EDP}\,(\Delta x, \Delta y).$$

Definition 4. Let s be a function $\mathbf{F}_2^8 \to \mathbf{F}_2^8$. The differential distribution table DDT is a $2^8 \times 2^8$ matrix of transition probabilities such that

$$\mathrm{DDT}[a][b] = \frac{\#\left\{x \in \mathbf{F}_2^8,\ \mathsf{s}\,(x) \oplus \mathsf{s}\,(x \oplus a) = b\right\}}{2^8} = \mathrm{DP}\,(a, b)\,,\ a, b \in \mathbf{F}_2^8,$$

and $p_{\max} = \max\limits_{a \neq 0, b} \mathrm{DDT}[a][b]$.

Definition 5. Let L-transformation (from $\mathbf{F}_{2^8}^n$ to $\mathbf{F}_{2^8}^n$) be $\mathbf{F}_{2^8}$-linear. We associate with L the code $\mathcal{C}_\mathsf{L}$ of length $2n$ over $\mathbf{F}_{2^8}$ defined by

$$\mathcal{C}_\mathsf{L} = \left\{(\mathbf{c}, \mathsf{L}(\mathbf{c}))\,,\ \mathbf{c} \in \mathbf{F}_{2^8}^n\right\}.$$

The differential branch number $\mathcal{B}_\mathsf{L}$ of the linear transformation L is the minimum distance of the code $\mathcal{C}_\mathsf{L}$

$$\mathcal{B}_\mathsf{L} = \min_{\mathbf{c} \neq 0} \mathrm{wt}\,(\mathbf{c}, \mathsf{L}(\mathbf{c}))\,.$$

Further, to simplify the text, we assume that $\mathcal{C}_\mathsf{L}$ is an MDS code and $\mathcal{B} = \mathcal{B}_\mathsf{L} = n + 1$.

2-round LSX-cipher may be represented as a sequence of operations

$$y = K_3 \oplus \mathsf{S}\left(K_2 \oplus \mathsf{LS}\left(K_1 \oplus x\right)\right),$$

where $x, y \in \mathbf{F}_{2^8}^n$ are the plaintext and the ciphertext, $K_1, K_2, K_3 \in \mathbf{F}_{2^8}^n$ are round keys derived from the masterkey K. The linear transformation on the last round was omitted without loss of generality.

A differential trail $\Omega = (\Delta x, \Delta_1, \Delta_2, \Delta y)$ in 2-round LSX is a collection of four differences, where $\Delta x = x \oplus x'$, Δ_1 is the difference after the first nonlinear transformation, $\Delta_2 = \mathsf{L}(\Delta_1)$, $\Delta y = y \oplus y'$, x and x' are plaintext blocks, y and y' are the corresponding ciphertext blocks.

Definition 6 ([16])**.** The expected 2-round trail Ω probability is defined as

$$\mathrm{EDCP}(\Omega) = 2^{-\kappa} \sum_{K \in \mathbf{F}_2^\kappa} \mathbf{P}_x \left(\Delta_1 = x_1 \oplus x_1' \text{ and } \Delta_2 = x_2 \oplus x_2' \text{ and } \Delta y = y \oplus y' \right),$$

where x is a random variable with the uniform distribution, $x' = \Delta x \oplus x$, x_1, x_1' are states after the first S-transformation, x_2, x_2' are states before the second S-transformation, κ is a size of the masterkey K.

We futher assume that all round keys are independent and uniformly distributed (so-called Markov assumption [8]). Under this assumption we have

$$\mathrm{EDCP}(\Delta x, \Delta_1, \Delta_2, \Delta y) = \left(\prod_{j=1}^{n} \mathrm{DP}(\Delta x[j], \Delta_1[j]) \right) \left(\prod_{j=1}^{n} \mathrm{DP}(\Delta_2[j], \Delta y[j]) \right).$$

Note that if $\mathrm{EDCP}(\Delta x, \Delta_1, \Delta_2, \Delta y) \neq 0$, then $\mathrm{Supp}(\Delta x) = \mathrm{Supp}(\Delta_1)$, $\mathrm{Supp}(\Delta_2) = \mathrm{Supp}(\Delta y)$ and (Δ_1, Δ_2) is a codeword of the code $\mathcal{C}_{\mathcal{L}}$. Therefore

$$\mathrm{EDP}(\Delta x, \Delta y) = \sum_{\substack{(\Delta_1, \Delta_2) \in \mathcal{C}_{\mathcal{L}}, \\ \mathrm{Supp}(\Delta x) = \mathrm{Supp}(\Delta_1), \\ \mathrm{Supp}(\Delta_2) = \mathrm{Supp}(\Delta y)}} \prod_{j \in \mathrm{Supp}(\Delta x)} \mathrm{DP}(\Delta x[j], \Delta_1[j]) \prod_{j \in \mathrm{Supp}(\Delta y)} \mathrm{DP}(\Delta_2[j], \Delta y[j]).$$

The equivalence of the above formula for $\mathrm{EDP}(\Delta x, \Delta y)$ and the definition 2 was proved in [8].

We define the weight (number of nonzero bytes) of the differential $(\Delta x, \Delta y)$ or the differential trail $(\Delta x, \Delta_1, \Delta_2, \Delta y)$ as $\mathrm{wt}(\Delta x) + \mathrm{wt}(\Delta y)$. Denote

$$\mathrm{MEDP}_w = \max_{\Delta x \neq 0, \Delta y, \mathrm{wt}(\Delta x) + \mathrm{wt}(\Delta y) = w} \mathrm{EDP}(\Delta x, \Delta y),$$

$$\mathrm{MEDP}^{+}_{w} = \max_{\Delta x \neq 0, \Delta y, \mathrm{wt}(\Delta x)+\mathrm{wt}(\Delta y) \geqslant w} \mathrm{EDP}\left(\Delta x, \Delta y\right), \ \mathcal{B} \leqslant w \leqslant 2 \cdot n.$$

Note that all mentioned definitions EDP, EDCP, MEDP are related to 2-round case unless otherwise stated.

Our main goal is to compute the nontrivial upper bound on $\mathrm{MEDP}^{+}_{\mathcal{B}+1}$, $\mathrm{MEDP}^{+}_{\mathcal{B}+2}$, etc.

3. Upper bound for non-minimum weight differentials

The strategy of our approach is as follows. Each differential trail $\Omega = (\Delta x, \Delta_1, \Delta_2, \Delta y)$ in 2-round differential $(\Delta x, \Delta y)$ uniquely corresponds to codeword (Δ_1, Δ_2) in $\mathcal{C}_{\mathcal{L}}$. All possible trails (codewords) in the differential satisfy conditions $\mathrm{Supp}(\Delta x) = \mathrm{Supp}(\Delta_1)$, $\mathrm{Supp}(\Delta_2) = \mathrm{Supp}(\Delta y)$. Derive constraints («maximum cost») for the entire set of such codewords. Divide the set into several subsets. Compute contribution to the constraints («cost») and the corresponding upper bound («value») for each possible subset. Select subsets so that the upper bound («total value») is maximum and the selection satisfies all constraints («total cost» does not exceed «maximum cost»). Thus, we obtain the upper bound on the differential.

3.1. Auxiliary lemmas

Lemma 1 (The rearrangement inequality [9])**.** *Let $l \in \mathbb{N}$, and suppose that $c_1, c_2, \ldots, c_l$ and $d_1, d_2, \ldots, d_l$ are sequences of nonnegative values. Let $\widetilde{c}_1, \widetilde{c}_2, \ldots, \widetilde{c}_l$ and $\widetilde{d}_1, \widetilde{d}_2, \ldots, \widetilde{d}_l$ be the sequences obtained by sorting original sequences in nonincreasing order. Then*

$$\sum_{i=1}^{l} c_i d_i \leqslant \sum_{i=1}^{l} \widetilde{c}_i \widetilde{d}_i.$$

Lemma 2. *Let $l \in \mathbb{N}$ and $c_1, c_2, \ldots, c_l$, $\widetilde{c}_1, \widetilde{c}_2, \ldots, \widetilde{c}_l$, and $d_1, d_2, \ldots, d_l$ be sequences of nonnegative values. All sequences are sorted in nonincreasing order. Suppose there exists l', $1 \leqslant l' \leqslant l$, such that*

1) $\widetilde{c}_i \geqslant c_i$, *for* $1 \leqslant i \leqslant l'$,
2) $\widetilde{c}_i \leqslant c_i$, *for* $l' + 1 \leqslant i \leqslant l$,
3) $\sum_{i=1}^{l} c_i \leqslant \sum_{i=1}^{l} \widetilde{c}_i$.

Then $\sum_{i=1}^{l} c_i d_i \leqslant \sum_{i=1}^{l} \widetilde{c}_i d_i$.

Proof. The proof of the lemma is given, for example, in [6]. □

If conditions 1–3 holds for some sequences $\widetilde{\boldsymbol{c}}$ and $\boldsymbol{c}$, then we say that $\widetilde{\boldsymbol{c}}$ is larger than $\boldsymbol{c}$ in the sense of Lemma 2.

Let D be a $h \times v$ matrix such that

$$D[i][j] \in \{p_1, p_2, ..., p_t, p_{\max}\},\ 1 \leqslant i \leqslant h,\ 1 \leqslant j \leqslant v,\ t \in \mathbb{N},$$
$$0 \leqslant p_1 < p_2 < \ldots < p_t < p_{\max} \leqslant 1,\ p_k, p_{\max} \in \mathbb{R},\ 1 \leqslant k \leqslant t.$$

Denote

$$\begin{gathered}\nu_k(D) = \#\{(i,j) : D[i][j] = p_k,\ 1 \leqslant i \leqslant h,\ 1 \leqslant j \leqslant v\},\ 1 \leqslant k \leqslant t,\\ \nu_{\max}(D) = \#\{(i,j) : D[i][j] = p_{\max},\ 1 \leqslant i \leqslant h,\ 1 \leqslant j \leqslant v\}.\end{gathered} \tag{1}$$

Denote by $\omega_l(D)$ the number of rows containing exactly l elements $p_{\max}$

$$\begin{gathered}\omega_l(D) = \#\{i\colon \#\{j\colon D[i][j] = p_{\max},\ 1 \leqslant j \leqslant v\} = l,\ 1 \leqslant i \leqslant h\},\\ \sum_{l=1}^{v} \omega_l(D) \cdot l = \nu_{\max}(D), \quad l_{\max}(D) = \max_{\omega_l(D) \neq 0} (l)\,.\end{gathered} \tag{2}$$

Let $\widetilde{D}$ be the reordered matrix D (see Fig. 1.a). The reordering procedure consists of three following steps:

1) sort each row of $\widetilde{D}$ in nonicreasing order,
2) sort each column of $\widetilde{D}$ in nonicreasing order,
3) reorder each unequal to $p_{\max}$ element:

$$\begin{gathered}\forall i, j, i', j' :\ \widetilde{D}[i][j] = p_{\max} \text{ or } \widetilde{D}[i'][j'] = p_{\max} \text{ or}\\ \left(\widetilde{D}[i][j] \geqslant \widetilde{D}[i'][j'],\ i' > i \text{ or } i' = i,\ j' > j\right),\\ 1 \leqslant i, i' \leqslant h,\ 1 \leqslant j, j' \leqslant v.\end{gathered}$$

Lemma 3. *Let D and $\widetilde{D}$ be defined as above, then*

$$\nu_k(D) = \nu_k(\widetilde{D}),\ \nu_{\max}(D) = \nu_{\max}(\widetilde{D}),\ 1 \leqslant k \leqslant t,$$
$$\omega_l(D) = \omega_l(\widetilde{D}),\ l_{\max}(D) = l_{\max}(\widetilde{D})\ \forall l \in \mathbb{N},$$
$$\sum_{i=1}^{h} \prod_{j=1}^{v} D[i][j] \leqslant \sum_{i=1}^{h} \prod_{j=1}^{v} \widetilde{D}[i][j].$$

Proof. The proof of the lemma is given in [21]. □

Lemma 4. *Let D and $\widetilde{D}$ be given as in Lemma 3. Suppose that $c_1, c_2, \ldots, c_h$ is a sequence of nonnegative values and $\widetilde{c}_1, \widetilde{c}_2, \ldots, \widetilde{c}_h$ be obtained from $c_1, c_2, \ldots, c_h$ by sorting in the nonincreasing order. Then*

$$\sum_{i=1}^{h} c_i \prod_{j=1}^{v} D[i][j] \leqslant \sum_{i=1}^{h} \widetilde{c}_i \prod_{j=1}^{v} \widetilde{D}[i][j].$$

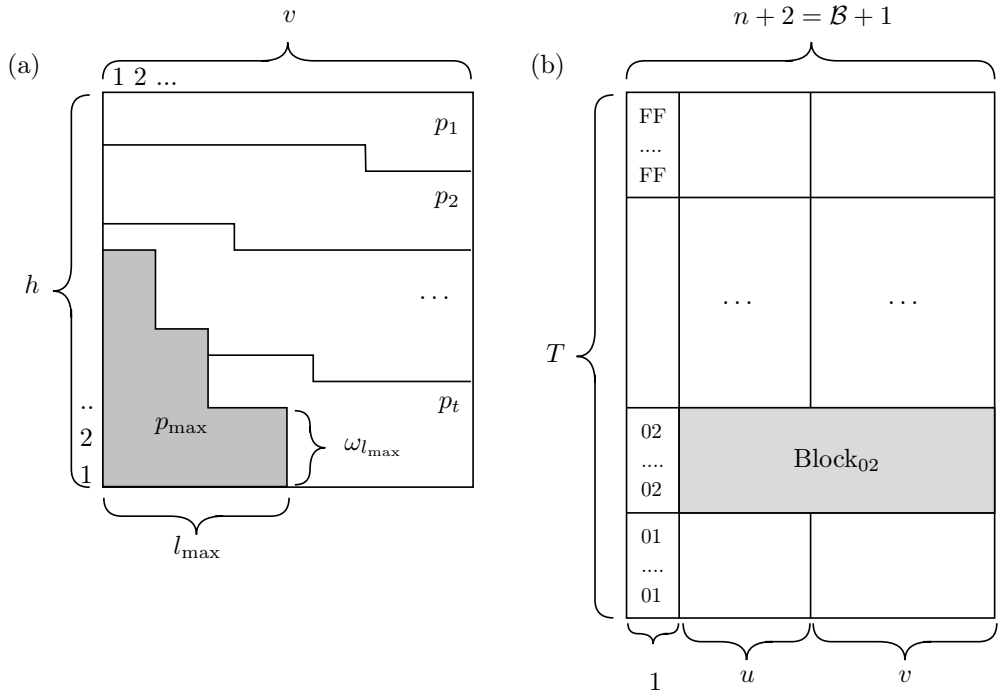

Fig. 1: (a) Example of matrix $\widetilde{D}$ after the reordering procedure. (b) Representation of Trails

Proof. Directly follows from Lemmas 1 and 3. □

3.2. Representation of trails in the differential

Consider an arbitrary differential $(\Delta x, \Delta y)$, $\mathrm{wt}(\Delta x) + \mathrm{wt}(\Delta y) = \mathcal{B} + 1$. We associate the differential with a T-element set of differential trails $(\Delta x, \Delta_1, \Delta_2, \Delta y)$. This set consists only of trails for which conditions $\mathrm{Supp}(\Delta x) = \mathrm{Supp}(\Delta_1) = \{k_1, k_2, \ldots, k_t\}$, $\mathrm{Supp}(\Delta y) = \mathrm{Supp}(\Delta_2) = \{m_1, m_2, \ldots, m_r\}$ are satisfied, $t + r = \mathcal{B} + 1 = n + 2$.

Recall that the linear transformation has the maximal differential branch number $\mathcal{B} = n + 1$. It is easy to show that the number of differential trails does not exceed $T \leqslant (2^8 - 1)^2$. Otherwise, there exists a pair of different codewords (Δ_1, Δ_2) and (Δ_1', Δ_2') such that

$$\mathrm{wt}\left((\Delta_1, \Delta_2) \oplus (\Delta_1', \Delta_2')\right) < \mathcal{B}.$$

Let's imagine a set of differential trails in the form of a table. Such a table, called Trails, has a size $T \times (n + 2)$. Each row consists of non-zero

bytes of the corresponding codeword.

$$\mathrm{Trails}[i] = \left(\Delta_1^{(i)}[k_1], \ldots, \Delta_1^{(i)}[k_t], \Delta_2^{(i)}[m_1], \ldots, \Delta_2^{(i)}[m_r]\right), \ 1 \leqslant i \leqslant T,$$

$$\mathrm{EDP}\left(\Delta x, \Delta y\right) = \sum_{i=1}^{T} \prod_{j=1}^{t} \mathrm{DP}(\Delta x[k_j], \mathrm{Trails}[i][j]) \cdot \prod_{j=t+1}^{t+r} \mathrm{DP}(\mathrm{Trails}[i][j], \Delta y[m_{j-t}]). \quad (3)$$

For definiteness let's sort the table by the byte value in the first column (see Fig.1.b).

Consider an arbitrary differential $(\Delta x, \Delta y)$ and the j-th column of Trails, $1 \leqslant j \leqslant t$. Let $\mathrm{Trails}[i_1][j] = \mathrm{Trails}[i_2][j]$ for some $i_1 \neq i_2$, then $\mathrm{DP}\left(\Delta x[k_j], \mathrm{Trails}[i_1][j]\right) = \mathrm{DP}\left(\Delta x[k_j], \mathrm{Trails}[i_2][j]\right)$. In other words, the same byte values in the column correspond to the same differential characteristics. Similarly for Δy. Let us denote the corresponding table by $\mathrm{DP}^*\left(\mathrm{Trails}\right)$, where

$$\begin{gathered}\mathrm{DP}^*\left(\mathrm{Trails}\right)[i][j] = \mathrm{DP}(\Delta x[k_j], \mathrm{Trails}[i][j]), \ 1 \leqslant i \leqslant T, \ 1 \leqslant j \leqslant t, \\ \mathrm{DP}^*\left(\mathrm{Trails}\right)[i][j] = \mathrm{DP}(\mathrm{Trails}[i][j], \Delta y[m_{j-t}]), \ 1 \leqslant i \leqslant T, \ t < j \leqslant t+r.\end{gathered} \quad (4)$$

We will divide table columns into 3 groups (subtables). The group C contains exactly 1 column. In the group $\mathrm{Trails}_{\mathbb{I}}$ there are u columns. The third group $\mathrm{Trails}_{\mathbb{III}}$ has v columns, $1 + u + v = n + 2$:

$$\begin{gathered}\mathrm{Trails} = \mathrm{C}||\mathrm{Trails}_{\mathbb{I}}||\mathrm{Trails}_{\mathbb{III}}, \\ \mathrm{DP}^*\left(\mathrm{Trails}\right) = \mathrm{DP}^*\left(\mathrm{C}\right)||\mathrm{DP}^*\left(\mathrm{Trails}_{\mathbb{I}}\right)||\mathrm{DP}^*\left(\mathrm{Trails}_{\mathbb{III}}\right),\end{gathered} \quad (5)$$

where $||$ is concatenation. We also denote

$$\mathrm{Block}_j = \left\{\mathrm{Trails}_{\mathbb{I}}[i]||\mathrm{Trails}_{\mathbb{III}}[i] : \mathrm{C}[i] = j, \ 1 \leqslant i \leqslant T\right\}, \ j \in \mathbf{F}_{2^8}^*. \quad (6)$$

3.3. DDT simplification

Let all elements in each row (column) of the DDT be sorted in nonincreasing order. The row and the column with zero indexes are ignored. Let us denote such table by $\mathrm{DDT}_{\mathrm{row}}$ ($\mathrm{DDT}_{\mathrm{col}}$) correspondingly

$$\begin{gathered}\mathrm{DDT}_{\mathrm{row}}[x][1] \geqslant \mathrm{DDT}_{\mathrm{row}}[x][2] \geqslant \ldots \geqslant \mathrm{DDT}_{\mathrm{row}}[x][2^8 - 1], \ x \in \mathbf{F}_{2^8}^*, \\ \mathrm{DDT}_{\mathrm{col}}[1][y] \geqslant \mathrm{DDT}_{\mathrm{col}}[2][y] \geqslant \ldots \geqslant \mathrm{DDT}_{\mathrm{col}}[2^8 - 1][y], \ y \in \mathbf{F}_{2^8}^*.\end{gathered}$$

We define sequences $\boldsymbol{m}_x$, $\boldsymbol{m}_y$ and $\boldsymbol{m}$ as

$$\boldsymbol{m}_x[i] = \max_{a \in \mathbf{F}_{2^8}^*} \mathrm{DDT}_{\mathrm{row}}[a][i], \quad \boldsymbol{m}_y[i] = \max_{a \in \mathbf{F}_{2^8}^*} \mathrm{DDT}_{\mathrm{col}}[i][a], \ i \in \mathbf{F}_{2^8}^*, \tag{7}$$

$$\boldsymbol{m}[i] = \max(\boldsymbol{m}_x[i], \boldsymbol{m}_y[i]), \ 1 \leqslant i \leqslant 2^8 - 1.$$

Let $\boldsymbol{r}$ be any nontrivial sorted row/column of the DDT. The sequence $\boldsymbol{m}$ is «larger» than any $\boldsymbol{r}$: $\boldsymbol{m}[i] \geqslant \boldsymbol{r}[i]$, $1 \leqslant i \leqslant 2^8 - 1$. Denote $\nu_{\max}(\boldsymbol{m}) = \#\{i : \boldsymbol{m}[i] = p_{\max}, \ 1 \leqslant i \leqslant 2^8 - 1\}$. Note that $\sum_{i=1}^{2^8-1} \boldsymbol{m}[i] \geqslant 1$.

For example, $\nu_{\max}(\boldsymbol{m}) = 2$ for Kuznyechik's S-box [1], $\nu_{\max}(\boldsymbol{m}) = 3$ for Khazad's S-box [12], $p_{\max} = 8/256$ in both cases.

We also define the sequences $\boldsymbol{\rho}$, $\boldsymbol{\rho}_x$, $\boldsymbol{\rho}_y$ as follows. Let $\boldsymbol{\rho}_x$ ($\boldsymbol{\rho}_y$) be one of the nontrivial sorted row (column) of the DDT. The sequence $\boldsymbol{\rho}_x$ ($\boldsymbol{\rho}_y$) must be larger than any other sorted row (column) of the DDT in the sense of Lemma 2 , $\sum_{i=1}^{2^8-1} \boldsymbol{\rho}_x[i] = \sum_{i=1}^{2^8-1} \boldsymbol{\rho}_y[i] = 1$. If $\boldsymbol{\rho}_x$ is larger than $\boldsymbol{\rho}_y$ in the sense of Lemma 2, then $\boldsymbol{\rho} = \boldsymbol{\rho}_x$, otherwise $\boldsymbol{\rho} = \boldsymbol{\rho}_y$.

3.4. Constraints

The next Lemma contains some restrictions on the set of codewords.

Lemma 5. *Let table* $\mathrm{Trails}_{\mathbb{III}}$ *and sequence* $\boldsymbol{m}$ *be given as above. The table* $\mathrm{DP}^*(\mathrm{Trails}_{\mathbb{III}})$ *is defined by analogy with* (4). *Let us denote by* $\omega_l(\mathrm{DP}^*(\mathrm{Trails}_{\mathbb{III}}))$ *the number of rows containing exactly* l *elements* $p_{\max}$:

$$\omega_l(\mathrm{DP}^*(\mathrm{Trails}_{\mathbb{III}})) \tag{8}$$
$$= \#\{i \colon \#\{j \colon \mathrm{DP}^*(\mathrm{Trails}_{\mathbb{III}})[i][j] = p_{\max}, 1 \leqslant j \leqslant v\} = l, 1 \leqslant i \leqslant T\}.$$

Then

$$\omega_2 \leqslant \binom{v}{2} \cdot (\nu_{\max}(\boldsymbol{m}))^2, \tag{9}$$

and

$$\sum_{l=2}^{v} \omega_l \cdot \binom{l}{2} \leqslant \binom{v}{2} \cdot (\nu_{\max}(\boldsymbol{m}))^2. \tag{10}$$

Proof. Let's consider two arbitrary columns of $\mathrm{Trails}_{\mathbb{III}}$. These columns do not contain any identical byte pairs (due to the fact that L has the maximum differential branch number). The total number of different byte pairs does not exceed $T \leqslant (2^8 - 1)^2$. In each column no more than $\nu_{\max}(\boldsymbol{m})$ values correspond to $p_{\max}$. Hence, no more than $(\nu_{\max}(\boldsymbol{m}))^2$ byte pairs correspond to $(p_{\max}, p_{\max})$. The number of ways to select 2 columns is $\binom{v}{2}$.

Thus we have (9).

Suppose that a row contains exactly 3 elements $p_{\max}$. Then $\binom{3}{2} = 3$ pairs of columns are generated, each of which contains a pair $(p_{\max}, p_{\max})$. Similarly for rows with l elements $p_{\max}$. Each of them «takes» $\binom{l}{2}$ pairs. We note once again that no more than $\nu_{\max}(m)$ values in each column are associated with the probability $p_{\max}$. Hence, the same upper bound as in (9) is preserved. Thereby we obtain (10). □

3.5. Bounds on $\mathrm{DP}^*(\mathrm{Block})$

Suppose that we are given an arbitrary $\mathrm{Block} \in \{\mathrm{Block}_j,\ j \in \mathbf{F}_{2^8}^*\}$. The block dimensions are $h \cdot (n+1)$, $h \leqslant 2^8 - 1$. We will give an upper bound on the contribution given by Block into the differential $\sum_{i=1}^{h} \prod_{j=1}^{n+1} \mathrm{DP}^*(\mathrm{Block})[i][j]$. We will use Lemmas 2, 3, 4.

Consider $v = 0$ and $u = n + 1$. Then we have

$$\sum_{i=1}^{h} \prod_{j=1}^{u} \mathrm{DP}^*(\mathrm{Block})[i][j] \leqslant \max\left(\max_{x \in \mathbf{F}_{2^8}^*} \sum_{i=1}^{2^8-1} (\mathrm{DDT}[x][i])^u, \max_{y \in \mathbf{F}_{2^8}^*} \sum_{i=1}^{2^8-1} (\mathrm{DDT}[i][y])^u \right). \quad (11)$$

The inequality (11) is so-called «FSE 2003 bound» on MEDP [14]. Lemma 2 allows to select a row (column) that maximizes expression (11). Then we can rewrite inequality (11) as

$$\sum_{i=1}^{h} \prod_{j=1}^{u} \mathrm{DP}^*(\mathrm{Block})[i][j] \leqslant \sum_{i=1}^{2^8-1} (\boldsymbol{\rho}[i])^u. \quad (12)$$

Let $v > 0$. We will divide Block into two parts:

$$\mathrm{Block} = \mathrm{Block}_{\mathbb{I}} || \mathrm{Block}_{\mathbb{III}},$$

$$\sum_{i=1}^{h} \prod_{j=1}^{n+1} \mathrm{DP}^*(\mathrm{Block})[i][j] = \sum_{i=1}^{h} \prod_{j=1}^{u} \mathrm{DP}^*(\mathrm{Block}_{\mathbb{I}})[i][j] \prod_{j=1}^{v} \mathrm{DP}^*(\mathrm{Block}_{\mathbb{III}})[i][j],$$

where $\mathrm{Block}_{\mathbb{I}}$ contains u columns, and $\mathrm{Block}_{\mathbb{III}}$ contains v columns, $u + v = n + 1$. We will evaluate the contribution of $\mathrm{Block}_{\mathbb{I}}$ by using the sequence

$$(\boldsymbol{\rho}[1])^u, (\boldsymbol{\rho}[2])^u, \ldots, \left(\boldsymbol{\rho}[2^8 - 1]\right)^u. \quad (13)$$

We will also get a bound on the contribution of $\text{Block}_{\text{III}}$ by using Lemma 3. Suppose that each column of $\text{DP}^*(\text{Block}_{\text{III}})$ contains elements from the sequence $\boldsymbol{m}$. Assume also that for all l, $0 \leqslant l \leqslant v$, we know

$$\begin{gathered}\omega_l\left(\text{DP}^*\left(\text{Block}_{\text{III}}\right)\right) \\ = \#\{i\colon \#\{j\colon \text{DP}^*\left(\text{Block}_{\text{III}}\right)[i][j] = p_{\max}, 1 \leqslant j \leqslant v\} = l, 1 \leqslant i \leqslant h\}, \\ \sum_{l=1}^{v} \omega_l \cdot l \leqslant \nu_{\max}(\boldsymbol{m}) \cdot v.\end{gathered} \tag{14}$$

In other words, ω_l is the number of rows containing exactly l elements $p_{\max}$. Let $\widetilde{\text{Block}_{\text{III}}}$ be a table obtained by the reordering procedure from Lemma 3. Then we get

$$\sum_{i=1}^{h}\prod_{j=1}^{v}\text{DP}^*\left(\text{Block}_{\text{III}}\right)[i][j] \leqslant \sum_{i=1}^{h}\prod_{j=1}^{v}\text{DP}^*\left(\widetilde{\text{Block}_{\text{III}}}\right)[i][j].$$

By means of Lemma 4 we finally obtain

$$\sum_{i=1}^{h}\prod_{j=1}^{n+1}\text{DP}^*\left(\text{Block}\right)[i][j] \leqslant \sum_{i=1}^{h}\left(\boldsymbol{\rho}[i]\right)^u\prod_{j=1}^{v}\text{DP}^*\left(\widetilde{\text{Block}_{\text{III}}}\right)[i][j]. \tag{15}$$

Thus, if we know the distribution ω_l, $0 \leqslant l \leqslant v$, then we can calculate the upper bound on $\sum_{i=1}^{h}\prod_{j=1}^{n+1}\text{DP}^*\left(\text{Block}\right)[i][j]$.

3.6. Optimization problem

Let's will form all possible sets

$$s_i = \left\{\left(l, \omega_l^{(i)}\right), 0 \leqslant l \leqslant v\right\},\ 1 \leqslant i \leqslant N. \tag{16}$$

For each set the equality $\sum_{l=1}^{v}\omega_l^{(i)} \cdot l = \nu_{\max}(\boldsymbol{m}) \cdot v$ is true. In fact, we construct all possible partitions of the number $\nu_{\max}(\boldsymbol{m}) \cdot v$. The maximum term in the partition does not exceed v.

For each set s_i we calculate the estimate (15) (denote this estimate by π_i) and «contribution» ζ_i for constraints (10): $\zeta_i = \sum_{l=2}^{v}\omega_l \cdot \binom{l}{2}$. We can choose such u and v in (15), which would *minimize* the estimation on the differential $(\Delta x, \Delta y)$. For most practical cases we use $u = 1$ and $v = n$. We get a set of pairs

$$(\pi_1, \zeta_1), (\pi_2, \zeta_2), \ldots, (\pi_N, \zeta_{N'}). \tag{17}$$

Pairs with the same ζ_i value may be removed. The pair with the largest π_i should not be removed. Hence $N' \leqslant \binom{v}{2} \cdot (\nu_{\max}(\boldsymbol{m}))^2$.

We can estimate the first column of DP^* (Trails) using the sequence $\boldsymbol{\rho}_x$ (or $\boldsymbol{\rho}_y$). Due to the fact that $\mathrm{wt}(\Delta x) \geqslant 1$ and $\mathrm{wt}(\Delta y) \geqslant 1$, we can choose $\boldsymbol{\rho}_x$ or $\boldsymbol{\rho}_y$. The aim of the choice is to *minimize* the final value. For definiteness we assume that $\boldsymbol{\rho}_x$ has been chosen.

Denote $I = i_1, i_2, \ldots, i_{2^8-1}$, $1 \leqslant i_j \leqslant N'$, $1 \leqslant j \leqslant 2^8 - 1$. Then

$$\mathrm{MEDP}_{\mathcal{B}+1} \leqslant \overline{\mathrm{MEDP}_{\mathcal{B}+1}} = \max_I \sum_{j=1}^{2^8-1} \boldsymbol{\rho}_x[j] \cdot \pi_{i_j} \text{ and } \sum_{i \in I} \zeta_i \leqslant \binom{v}{2} \cdot (\nu_{\max}(\boldsymbol{m}))^2 . \tag{18}$$

The optimal I is chosen by us using dynamic programming (see non-optimized version of the pseudocode in [21]).

There exists a trivial estimate $\mathrm{MEDP}_{\mathcal{B}+2} \leqslant \sum_{i=1}^{2^8-1} \boldsymbol{\rho}[i] \cdot \overline{\mathrm{MEDP}_{\mathcal{B}+1}} = \overline{\mathrm{MEDP}_{\mathcal{B}+1}}$. Similar estimate may be obtained for $\mathrm{MEDP}_{\mathcal{B}+3}$, etc. Thus, we have proved that $\mathrm{MEDP}^+_{\mathcal{B}+1} \leqslant \overline{\mathrm{MEDP}_{\mathcal{B}+1}}$.

3.7. Another constraints

We can compute the estimate for $\mathrm{MEDP}^+_{\mathcal{B}+1}$ with more higher precision.

Consider the table $\mathrm{DP}^*(\mathrm{Trails}_{\mathbb{III}})$. The number of rows that contain many elements $p_{\max}$ is quite small.

Recall that $\mathrm{wt}(\mathrm{Trails}_{\mathbb{III}}[i] \oplus \mathrm{Trails}_{\mathbb{III}}[j]) \geqslant v - 1$, $i \neq j$. Otherwise, there is a codeword $c \in \mathcal{C}_{\mathsf{L}}$, $\mathrm{wt}(c) < \mathcal{B}$. Thus, any two rows of $\mathrm{Trails}_{\mathbb{III}}$ either differ only by one byte, or do not have any matches.

In each column of $\mathrm{Trails}_{\mathbb{III}}$, no more than $\nu_{\max}(\boldsymbol{m})$ bytes are equal to $p_{\max}$, and $\mathrm{Trails}_{\mathbb{III}}$ has v columns. Denote $W = \nu_{\max}(\boldsymbol{m}) \cdot v$.

Suppose that some row of $\mathrm{DP}^*(\mathrm{Trails}_{\mathbb{III}})$ contains w_1 elements $p_{\max}$. We decrease W by w_1. Let the other row contain w_2 elements $p_{\max}$. These two rows may intersect at most by one byte. Therefore, W has decreased by at least $w_2 - 1$. The third row may intersect with the first and the second rows. Hence we subtract $w_3 - 2$ from W. Continue until $W \geqslant 0$.

Let a series $w_1, w_2, w_3, \ldots, w_T$ be sorted in noninreasing order, where T is the number of rows in $\mathrm{Trails}_{\mathbb{III}}$. Then

$$\left(\nu_{\max}(\boldsymbol{m}) \cdot v - \sum_{i=1}^{l} (w_i - (i-1)) \right) \geqslant 0 \tag{19}$$

must be true for all $l \leqslant T$.

Let's form all series $\psi = w_1, w_2, \ldots, w_l$ for which the inequality (19) is true. Denote the set of such series by Ψ. We will use a relatively small value of l (about 5, 6).

We can modify the algorithm from Subsection 3.6 as follows. For each set s_i from (16) we form a series $\psi = w_1, w_2, \ldots, w_l$. We obtain a sequence similar to (17): $(\pi_1, \zeta_1, \psi_1), (\pi_2, \zeta_2, \psi_2), \ldots, (\pi_N, \zeta_N, \psi_N)$.

Hence, additional constraint is inluded into the optimization problem (18):

$$\mathrm{sort}_l \left(\psi_{i_1} || \psi_{i_2} || \ldots || \psi_{i_{2^8-1}} \right) \in \Psi, \ 1 \leqslant i_j \leqslant N, 1 \leqslant j \leqslant 2^8 - 1,$$

where sort_l denotes the set of l largest elements of the sequence. Note that we may store not the entire sequence $\psi_{i_1} || \psi_{i_2} || \ldots || \psi_{i_{2^8-1}}$ in memory, but only its first l values. Accounting the limitations described in this subsection requires anormous computing resources. Therefore, this modification is only used in the calculation of bound on $\mathrm{MEDP}^+_{\mathcal{B}+1}$. The case $\mathrm{MEDP}^+_{\mathcal{B}+2}$ and others are computationally infeasible with the constraints considered.

3.8. Computing $\mathrm{MEDP}^+_{\mathcal{B}+2}$ and other values

Let $(\Delta x, \Delta y)$ be such that $\mathrm{wt}(\Delta x) + \mathrm{wt}(\Delta y) = \mathcal{B} + 2 = n + 3$. Then Lemma 5 may be reformulated as follows.

Lemma 6. *Let the conditions of Lemma 5 be satisfied and weight of the differential be equal to $n + 3$. Then*

$$\sum_{l=3}^{v} \omega_l \cdot \binom{l}{3} \leqslant \binom{v}{3} \cdot \left(\nu_{\max}(\boldsymbol{m})\right)^3. \tag{20}$$

The algorithm is similar to one in the Subsection 3.6, but the optimization problem is solved in two steps. As in Subsection 3.6:

– form all possible sets $s_i = \{(l, \omega_l), 0 \leqslant l \leqslant v\}$, $1 \leqslant i \leqslant N$, $\sum_{l=1}^{v} \omega_l \cdot l = \nu_{\max}(\boldsymbol{m}) \cdot v$,

– for each set s_i, calculate the estimate π_i by (15), $\zeta_i = \sum_{l=2}^{v} \omega_l \cdot \binom{l}{2}$; $\eta_i = \sum_{l=3}^{v} \omega_l \cdot \binom{l}{3}$.

We obtain the sequence $(\pi_1, \zeta_1, \eta_1), (\pi_2, \zeta_2, \eta_2), \ldots, (\pi_N, \zeta_N, \eta_N)$.

Let's solve the first optimization problem for all values $\eta' \leqslant \binom{v}{3} \cdot \left(\nu_{\max}(\boldsymbol{m})\right)^3$. Denote $I = i_1, i_2, \ldots, i_{2^8-1}$, $i_j \in \mathbb{N}$, $1 \leqslant j \leqslant 2^8 - 1$,

$$\pi' = \max_I \sum_{j=1}^{2^8-1} \boldsymbol{\rho}_x[j] \cdot \pi_{i_j},$$

under condition $\sum_{i\in I}\zeta_i \leqslant \binom{v}{2}\cdot(\nu_{\max}(\boldsymbol{m}))^2$ and $\sum_{i\in I}\eta_i = \eta'$. We can get all the values η' by solving the optimization problem once.

Thus, the sequence $(\pi'_1,\eta'_1),(\pi'_2,\eta'_2),\ldots,(\pi'_{N'},\eta'_{N'})$ will be obtained, $N' \leqslant \binom{v}{3}\cdot(\nu_{\max}(\boldsymbol{m}))^3$.

Further we solve the second optimization problem

$$\mathrm{MEDP}^{+}_{\mathcal{B}+2} \leqslant \overline{\mathrm{MEDP}_{\mathcal{B}+2}} = \max_{I}\sum_{j=1}^{2^8-1}\boldsymbol{\rho}_x[j]\cdot\pi'_{i_j} \text{ and } \sum_{i\in I}\eta'_i \leqslant \binom{v}{3}\cdot(\nu_{\max}(\boldsymbol{m}))^3.$$

The pseudocode in [21] contains a non-optimized version of the algorithm. Application of the described approach is computationally infeasible for $\mathrm{MEDP}^{+}_{\mathcal{B}+3}$ in most cases. Furthermore, the potential improvement of the estimation is very small (see summary table 1).

4. New bounds on MEDP for 2-round Kuznyechik

Kuznyechik block cipher [1] consists of a sequence of 9 rounds and a post-whitening key addition. The block size is 128 bits ($n = 16$ bytes), the key has a size of 256 bits. The cipher S-box has no explicit analytical form [20], such as in AES. The rows and columns of the DDT have different unbalanced distributions. The sequence $\boldsymbol{m}_y$ is «larger» than $\boldsymbol{m}_x$. The L-transformation is defined as a LFSR over $\mathbf{F}_{2^8}$, the differential branch number $\mathcal{B} = n + 1$.

In [19] it was proved that each 2-round best differential contains only one differential trail
$\mathrm{MEDP} = \mathrm{MEDP}_{\mathcal{B}} = \max\limits_{\Omega\neq 0}\mathrm{EDCP}(\Omega) = \left(\frac{8}{256}\right)^{13}\left(\frac{6}{256}\right)^{4} = 2^{-86.66\ldots}$.

Using the proposed algorithms we find that

$$\mathrm{MEDP}^{+}_{\mathcal{B}+1} \leqslant 2^{-87.54\ldots}, \quad \mathrm{MEDP}^{+}_{\mathcal{B}+2} \leqslant 2^{-88.34\ldots}.$$

The calculation of $\mathrm{MEDP}^{+}_{\mathcal{B}+1}$ and $\mathrm{MEDP}^{+}_{\mathcal{B}+2}$ used the fact that $\mathrm{wt}(\Delta x) \geqslant 2$. We can use $\boldsymbol{\rho}_x$ instead of $\boldsymbol{\rho}$ (the rows of DDT instead the columns) in (15) if $u \leqslant 2$. Bound on $\mathrm{MEDP}^{+}_{\mathcal{B}+3}$ obtained in this way will be not less than $2^{-88.42\ldots}$.

Table 1 shows all computed values. The numbers are rounded to the second decimal place. The second data column presents the bounds we obtain using «FSE 2003 bounds» [14]. The last column (*) shows the limitation on the capabilities of the presented algorithm. For information on the linear method (including definitions of $p_{\mathrm{lin,max}}$, MELP and others) see Appendix in the extended version [21] of the paper.

Table 1: Summary table of results for Kuznyechik ($\log_2$ scale)

$(p_{\max})^{\mathcal{B}}$	$\overset{\text{FSE2003}}{\text{MEDP}_{\mathcal{B}}\leqslant}$	$\text{MEDP}_{\mathcal{B}} =$	$\text{MEDP}^{+}_{\mathcal{B}+1} \leqslant$	$\text{MEDP}^{+}_{\mathcal{B}+2} \leqslant$	$(*)\text{MEDP}^{+}_{\mathcal{B}+3} \leqslant$
-85	-83.97	-86.66	-87.54	-88.34	-88.42
$(p_{\text{lin,max}})^{\mathcal{B}}$	$\overset{\text{FSE2003}}{\text{MELP}_{\mathcal{B}}\leqslant}$	$\text{MELP}_{\mathcal{B}} =$	$\text{MELP}^{+}_{\mathcal{B}+1} \leqslant$	$\text{MELP}^{+}_{\mathcal{B}+2} \leqslant$	$(*)\text{MELP}^{+}_{\mathcal{B}+3} \leqslant$
-74.54	-73.54	-76.73	-77.15	-79.63	-80.50

5. Conclusion

We propose a dynamic programming algorithm for bounding non-minimum weight differentials (linear hulls) in 2-round LSX-ciphers. By means of the described algorithm we derive some new bounds on differentials and linear hulls for 2-round Kuznyechik (Table 1). Similar results were obtained for 2-round Khazad (see the extended version of the article [21]), and as a result, the exact values of $\text{MEDP} = 2^{-45} + 2^{-60}$ and $\text{MELP} = 2^{-37.80\ldots}$ were also proved.

The source codes of our algorithms may be found at
https://gitlab.com/v.kir/diff2rLSX

For any LSX-cipher with independent round keys, the R-round MEDP (MELP) is the upper bound for $(R+1)$-round MEDP (MELP). The presented results form a step towards obtaining new nontrivial bounds on R-round MEDP (MELP), i.e. new proofs of Kuznyechik strength against differential and linear cryptanalysis.

Список литературы

[1] *GOST R* 34.12-2018 – *National standard of the Russian Federation – Information technology – Cryptographic data security – Block ciphers*, 2018.

[2] Biham, E., Shamir, A., "Differential cryptanalysis of DES-like cryptosystems", *J. Cryptology*, 1991, 3–72.

[3] Matsui M., "Linear cryptanalysis method for DES cipher", EUROCRYPT'93, Lect. Notes Comput. Sci., **765**, 1994, 386–397.

[4] Biham E., "On Matsui's linear cryptanalysis", EUROCRYPT'94, Lect. Notes Comput. Sci., **950**, 341–355.

[5] Keliher L., Sui. J., "Exact maximum expected differential and linear probability for 2-round Advanced Encryption Standard (AES)", *IET Inf. Security*, **1**:2 (2007), 53–57.

[6] Keliher L., *Linear Cryptanalysis of Substitution-Permutation Networks*, PhD Thesis, Queen's Univ., Kingston, Canada, 2003.

[7] Keliher L., "Refined analysis of bounds related to linear and differential cryptanalysis for the AES", Lect. Notes Comput. Sci., **3373**, 2005, 42–57.

[8] Lai X., Massey J.L., Murphy S., "Markov ciphers and differential cryptanalysis", EUROCRYPT'91, Lect. Notes Comput. Sci., **547**, 1991, 17–38.

[9] Hardy G.H., Littlewood J.E., Polya G., *Inequalities*, Cambridge: Cambridge Univ. Press, 1952.

[10] Sano F., Ohkuma K., Shimizu H., Kawamura S., "On the security of nested SPN cipher against the differential and linear cryptanalysis", *IEICE Trans. on Fundam. Electronics, Communic. and Comput. Sci.*, **E86-A**:1 (2003), 37–46.

[11] Daemen J., Rijmen V., *The Design of Rijndael: AES – The Advanced Encryption Standard*, Springer, Heidelberg etc., 2002, x+238 pp.

[12] Barreto P., Rijmen V., "The Khazad legacy-level block cipher", First open NESSIE Workshop, November 2000.

[13] Kang J.-S., Hong S., Lee S., Yi O., Park C., Lim J., "Practical and provable security against differential and linear cryptanalysis for substitution-permutation networks", *ETRI J.*, **23**:4 (2001).

[14] Park S., Sung S.H., Lee S., Lim J., "Improving the upper bound on the maximum differential and the maximum linear hull probability for SPN structures and AES", FSE 2003, Lect. Notes Comput. Sci., **2887**, 2003, 247–260.

[15] Canteaut A., Roue J., "On the behaviors of affine equivalent S-boxes regarding differential and linear attacks", EUROCRYPT 2015, Lect. Notes Comput. Sci., **9056**, 2015, 45–74.

[16] Canteaut A., Roue J., "Differential attacks against SPN: A thorough analysis", C2SI 2015, Lect. Notes Comput. Sci., **9084**, 2015, 45–62.

[17] Malyshev F.M., Trifonov D.I., "Diffusion properties of XSLP-ciphers", *Matematicheskie Voprosy Kriptografii*, **7:3** (2016), 47–60.

[18] Malyshev F.M., Trishin A.E., "Linear and differential cryptanalysis: Another viewpoint", *Matematicheskie Voprosy Kriptografii*, **11**:2 (2020), 83–98.

[19] Kiryukhin V.A., "Exact maximum expected differential and linear probability for 2-round Kuznyechik", *Matematicheskie Voprosy Kriptografii*, **10**:2 (2019), 107–116.

[20] Shishkin V., Marshalko G., "A memo on Kuznyechik S-box", ISO/IEC JTC 1/SC 27/WG 2 Officer's Contribution N1804, September 2018, 5 pp.

[21] Kiryukhin V., *An algorithm for bounding non-minimum weight differentials in 2-round LSX-ciphers*, 2020, arXiv: Report 2020/1208.