

Related-key Attack on 5-Round Kuznyechik

Vitaly Kiryukhin

JSC «InfoTeCS», Russia
vitaly.kiryukhin@infotecs.ru

Abstract

The first related-key attack on 3-round (of 9) Kuznyechik with 2-round (of 8) key schedule was presented in CTCrypt'18. This article describes a related-key attack on 5-round cipher with the same key schedule. The presented one also has a practical complexity (2^{32} operations, 2^{30} memory, 2^{16} related keys) and verified in practice. We obtained result due to the simultaneous use of the integral properties of the cipher transformations and the key schedule.

Keywords: Kuznyechik, related-key attack, integral cryptanalysis.

1 Introduction

The setting of a related-key attack on cipher was introduced in [6]. Informally this model assumes that adversary has access to several encryptors with different unknown keys, but it knows a certain simple relationship (for example, bitwise xor) between these keys.

In some cases the related-key model is quite consistent with reality. A good example is an iterative hash function using block cipher as part of compression function. In this case, adversary has a possibility of manipulating the encryption keys. Some cryptographic protocols may use related keys by design. One such related-key protocol CTRR was proposed at CTCrypt'18 [2].

In the same publication [2], the first related-key attack on a reduced variant of block cipher Kuznyechik [1] was proposed. This approach exploits the ability of attacker to manipulate keys, and the similarity of the functions in encryption and the key schedule procedures.

In this paper we present a related-key attack on 5-round (of 9) Kuznyechik with 2-round (of 8) key schedule. Main result obtained due to the integral properties [4, 5] of encryption and the key schedule. We also used some approaches from [3]. The simplified versions of Kuznyechik are described in the next section (equations (2) and (3)).

The presented attack was verified in practice with the help of C++ implementation. Source codes can be found at <https://gitlab.com/v.kir/rk-5R-kuznyechik>.

Comparative characteristics of attacks are presented in table 1.

Cipher rounds	Key schedule rounds	Operations	Keys	Memory	Source
3	2	2^{12}	2^{12}	\sim	[2]
5	2	2^{32}	2^{16}	2^{30}	Section 4

Table 1: Related-key attacks on Kuznyechik

2 Definitions

Let \mathbb{F}_{2^8} be a finite field as defined in [1]. Each element of \mathbb{F}_{2^8} can be interpreted as an integer or binary vector. Field elements are indicated by lowercase letters: a, b . Denote vector space of dimension $n \in \mathbb{N}$ over \mathbb{F}_{2^8} by $\mathbb{F}_{2^8}^n$. Elements from $\mathbb{F}_{2^8}^n$ will be denoted by capital letters: A, B . Blocks of plaintext and ciphertext also belong to $\mathbb{F}_{2^8}^n$.

Denote bitwise xor operation by symbol \oplus . Let we have a sequence of blocks

$$B_0, \dots, B_d \in \mathbb{F}_{2^8}^n, d \in \mathbb{N},$$

then we refer to sequence

$$\Delta \mathbf{B} = (B_0 \oplus B_1, B_0 \oplus B_2, \dots, B_0 \oplus B_d) \in (\mathbb{F}_{2^8}^n)^d \quad (1)$$

as a difference. Throughout the article we always use $d = 2^8 - 1$. Differences are indicated by bold: $\boldsymbol{\kappa}, \Delta \mathbf{K}$.

The transformations over $\mathbb{F}_{2^8}^n$ (or sets of elements from $\mathbb{F}_{2^8}^n$) are denoted by Sans Serif font: $\mathbf{f}, \mathbf{S}, \mathbf{L}$. Such characters may mean a bijective transformation of blocks ($\mathbf{f}(A), A \in \mathbb{F}_{2^8}^n$) or non-bijective transformation of differences to the set of differences (for example, $\mathbf{S}(\boldsymbol{\kappa})$ is a set of differences, $\boldsymbol{\kappa} \in (\mathbb{F}_{2^8}^n)^d$). The notation \mathbf{LS} indicates a composition of transformations, where \mathbf{S} applies first.

The difference $\Delta \in (\mathbb{F}_{2^8}^n)^d$ can also be interpreted as n «columns» of d bytes each: $\Delta \in (\mathbb{F}_{2^8}^d)^n$. If i -th «column» ($i = 1, 2, \dots, n$) $\boldsymbol{\alpha} \in \mathbb{F}_{2^8}^d$ contains all different non-zero bytes, we say that i -th position has an integral property *All* (\mathbf{A}). Similarly, if xor of all bytes is equal to zero, then i -th position of the difference has an integral property *Zero* ($\mathbf{0}$). Obviously, the property \mathbf{A} implies the property $\mathbf{0}$. If at least one byte in such «column» is non-zero, we say that i -th position is active, otherwise inactive.

Kuznyechik

Kuznyechik [1] consists of a sequence of 9 rounds and a post-whitening key addition. Each round contains three operations:

- X – modulo 2 addition of an input block with an iterative key;
- S – parallel application of a fixed bijective substitution to each byte of the block;
- L – linear transformation defined as an LFSR over \mathbb{F}_{2^8} .

The block size is 128 bits ($n = 16$ bytes), the size of key K is equal to 256 bits.

Key schedule uses round constants $C_i \in \mathbb{F}_{2^8}^n, i = 1, 2, \dots, 32$.

Round keys $K_i \in \mathbb{F}_{2^8}^n, i = 1, 2, \dots, 10$ are derived from a master key K as follows:

$$\begin{aligned}
K &= K_1 || K_2, \\
(K_{2i+1}, K_{2i+2}) &= \mathbf{F}[C_{8(i-1)+8}] \dots \mathbf{F}[C_{8(i-1)+1}](K_{2i-1}, K_{2i}), \quad i = 1, 2, 3, 4, \\
\mathbf{F}[C](A_1, A_2) &= (\mathbf{LSX}[C](A_1) \oplus A_2, A_1), \quad C, A_1, A_2 \in \mathbb{F}_{2^8}^n.
\end{aligned}$$

We define 3-round Kuznyechik as in [2]. Each round of the key schedule has only 2 rounds of basic cipher's Feistel rounds.

$$\begin{aligned} E_{K_1, K_2}(A) &= \mathbf{X}[K_4]\text{LSX}[K_3]\text{LSX}[K_2]\text{LSX}[K_1](A), \\ (K_3, K_4) &= \text{F}[C_2]\text{F}[C_1](K_1, K_2) \\ K_3 &= K_1 \oplus \text{LSX}[C_2](K_2 \oplus \text{LSX}[C_1](K_1)), \\ K_4 &= K_2 \oplus \text{LSX}[C_1](K_1). \end{aligned} \tag{2}$$

5-round Kuznyechik is defined in a similar way:

$$\begin{aligned} E_{K_1, K_2}(A) &= \mathbf{X}[K_6]\text{LSX}[K_5]\text{LSX}[K_4]\text{LSX}[K_3]\text{LSX}[K_2]\text{LSX}[K_1](A), \\ (K_3, K_4) &= \text{F}[C_2]\text{F}[C_1](K_1, K_2), \\ (K_5, K_6) &= \text{F}[C_4]\text{F}[C_3](K_3, K_4). \end{aligned} \tag{3}$$

Denote also the block before addition of the key K_i by P_i (for example $P_2 = \text{LSX}[K_1](A)$).

3 Technical lemmas and concepts

The polytopic cryptanalysis was first introduced in [3]. We will use some techniques from this concept along with integral cryptanalysis [4].

In particular, we use the difference (1) as « d -difference» in [3]. Let's consider how cipher transformations change this difference.

It's easy to see, that adding a *same* round key does not change the difference. The attack presented in section 4 uses non-equal keys. In this case, the difference between the round keys is added to the difference between the intermediate states. Note that if both such differences $\Delta, \kappa \in (\mathbb{F}_{2^8}^n)^d$ have integral property $\mathbf{0}$, then $\Delta \oplus \kappa$ has the same property.

Suppose that the difference $\Delta \in (\mathbb{F}_{2^8}^n)^d$ has only one active position, then after the S-transformation we have no more than 2^8 possible differences. Indeed, all inactive positions remain inactive. We have one non-zero «column» $\alpha = (c_1, c_2, \dots, c_d) \in \mathbb{F}_{2^8}^d$ and after substitution layer:

$$\mathbf{s}(\alpha) = \{(\mathbf{s}(x \oplus c_1) \oplus \mathbf{s}(x), \mathbf{s}(x \oplus c_2) \oplus \mathbf{s}(x), \dots, \mathbf{s}(x \oplus c_d) \oplus \mathbf{s}(x)), x \in \mathbb{F}_{2^8}\},$$

where $\mathbf{s} : \mathbb{F}_{2^8} \rightarrow \mathbb{F}_{2^8}$ is cipher Sbox. Obviously, the number of differences $\mathbf{s}(\alpha)$ does not exceed the number of x . In most cases, these numbers are equal. If all bytes in α are different, all bytes in $\mathbf{s}(\alpha)$ are also different (the bijective Sbox preserve the integral property \mathbf{A}). If we know α and $\alpha' \in \mathbf{s}(\alpha)$, we can easily find the corresponding x .

The L-transformation bijectively maps one difference to another:

$$\Delta = (\Delta_1, \Delta_2, \dots, \Delta_d), \quad \mathbf{L}(\Delta) = (\mathbf{L}(\Delta_1), \mathbf{L}(\Delta_2), \dots, \mathbf{L}(\Delta_d)).$$

If only one position in input difference is active then all positions in output difference are active (this is true if \mathbf{L} is MDS matrix). Under the same conditions, if one position has the property \mathbf{A} , then all output positions will have this property. The integral property $\mathbf{0}$ is preserved by L-transformation:

$$\bigoplus_{i=1}^d \Delta_i = 0, \quad \bigoplus_{i=1}^d \mathbf{L}(\Delta_i) = \mathbf{L}\left(\bigoplus_{i=1}^d \Delta_i\right) = 0.$$

We will use the so-called integral property [4, 5] of LSXLSX transformation.

Lemma 1. *Let one position in the difference $\Delta \in (\mathbb{F}_{2^8}^n)^d$ has integral property \mathbf{A} and all other positions are inactive (so-called δ -set). Then any difference from $\text{LSXLSX}(\Delta)$ has the integral property $\mathbf{0}$.*

Proof. Adding a round key does not change the difference. Thus, we have $\text{LSLS}(\Delta)$. After the first substitution layer, one position will have the property \mathbf{A} and all others will remain inactive. The first linear transformation will make all bytes active. Each of them will have the property \mathbf{A} . The second \mathbf{S} transformation will preserve \mathbf{A} and consequently the property $\mathbf{0}$. Hence, after the last linear transformation we have the property $\mathbf{0}$ in each position of the difference. \square

Equivalent representation of the last two rounds

The presented attack uses an equivalent representation of the last two rounds.

Let $A, B \in \mathbb{F}_{2^8}^n$ be a plaintext and ciphertext correspondingly. K_1, \dots, K_r, K_{r+1} are round keys, $K_i \in \mathbb{F}_{2^8}^n$, $i = 1, 2, \dots, r + 1$.

The original cipher has the form

$$B = \mathbf{X}[K_{r+1}]\text{LSX}[K_r] \dots \mathbf{X}[K_1](A) = \mathbf{E}_{r+1}(A).$$

Apply the inverse linear transformation to the known ciphertext

$$\begin{aligned} \mathbf{L}^{-1}(B) &= \mathbf{L}^{-1}(\mathbf{X}[K_{r+1}]\text{LSX}[K_r] \dots \mathbf{X}[K_1](A)), \\ \mathbf{L}^{-1}(B) &= \mathbf{L}^{-1}(K_{r+1}) \oplus \mathbf{SX}[K_r] \dots \mathbf{X}[K_1](A). \end{aligned}$$

We denote $B' = \mathbf{L}^{-1}(B)$, $K'_i = \mathbf{L}^{-1}(K_i)$, then the cipher has the form

$$B' = \mathbf{X}[K'_{r+1}]\mathbf{SX}[K_r]\text{LSX}[K_{r-1}] \dots \mathbf{X}[K_1](A).$$

Similarly, for the penultimate round. Let's consider the transformation

$$\mathbf{X}[K_r]\mathbf{L}(A) = K_r \oplus \mathbf{L}(A) = \mathbf{L}(A \oplus \mathbf{L}^{-1}(K_r)) = \mathbf{LX}[K'_r](A).$$

Therefore, the cipher transformation can be represented by the formula

$$B' = \mathbf{X}[K'_{r+1}]\mathbf{SLX}[K'_r]\mathbf{SX}[K_{r-1}] \dots \mathbf{X}[K_1](A).$$

4 Related-key attack

Let's represent 5-round Kuznyechik (3) in equivalent form

$$\begin{aligned} \mathbf{E}_{K_1, K_2}(A) &= \mathbf{X}[K'_6]\mathbf{SLX}[K'_5]\mathbf{SX}[K_4]\text{LSX}[K_3]\text{LSX}[K_2]\text{LSX}[K_1](A), \\ (K_3, K_4) &= \mathbf{F}[C_2]\mathbf{F}[C_1](K_1, K_2), \\ K_4 &= K_2 \oplus \text{LSX}[C_1](K_1), \\ K_3 &= K_1 \oplus \text{LSX}[C_2](K_4), \\ (K_5, K_6) &= \mathbf{F}[C_4]\mathbf{F}[C_3](K_3, K_4), \\ K_6 &= K_4 \oplus \text{LSX}[C_3](K_3), \quad K'_6 = \mathbf{L}^{-1}(K_6), \\ K_5 &= K_3 \oplus \text{LSX}[C_4](K_6), \quad K'_5 = \mathbf{L}^{-1}(K_5). \end{aligned}$$

The attack consists of the following steps:

1. Adversary chooses 2^8 collections of related keys, 2^8 keys in each collection. One plaintext C_1 (first constant in the key schedule) will be used.
2. For one of these collections, the special easy verifiable property (integral distinguisher) is true.
3. The round keys K_6, K_5 are recovered by using integral and polytopic properties.

Let's describe these steps in more detail. We denote

$$\kappa = \begin{pmatrix} 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & \dots & 0 & 2 \\ \dots & \dots & \ddots & \dots & \dots \\ 0 & 0 & \dots & 0 & 255 \end{pmatrix}$$

$\underbrace{\hspace{15em}}_{n=16}$

the difference between keys K_1 . The set $\text{LS}(\kappa)$ contains 2^8 differences. The collection of the related keys looks like

$$(K_1, K_2) \text{ and set } (K_1 \oplus \kappa, K_2 \oplus \kappa''), \text{ where } \kappa'' \in \text{LS}(\kappa).$$

It is easy to see that each collection contains the «main» key and a set of 255 related keys. Adversary does not know the keys, but it know all relations (κ and $\kappa'' \in \text{LS}(\kappa)$) between them. Adversary encrypts only one plaintext C_1 and gets 2^8 ciphertexts for each collection of keys. In total we have $1 + 2^8 \cdot (2^8 - 1)$ different keys and different ciphertexts correspondingly. In the same collection we refer to the difference between i -th round keys K_i as ΔK_i , for example $\kappa = \Delta K_1$ and $\kappa'' = \Delta K_2$.

4.1 Integral property

Figure 1 shows the propagation of differences, which is true for only one collection of keys (for only one $\kappa'' \in \text{LS}(\kappa)$). Active Sboxes have a gray background. Integral properties are indicated in red bold (**A** – all bytes are different, **0** – bitwise xor of all bytes is zero). More detailed pictures are presented in Appendix B.

Let's propagate the difference through S^{-1} . For each of 15 Sboxes we have 2^8 possible differences and for 16'th Sbox we get 2^{16} differences due to $\Delta K'_6 \in S(\kappa)$.

Let's check the integral property $\mathbf{0}$ for each obtained difference. If we correctly guessed κ'' , then there must be at least one such difference for each Sbox. Otherwise, if we do not guess it correctly, then there is at least one Sbox for which there is no such difference. Generally speaking, it is possible that a «false» collection of the related keys will satisfy this property. The probability of the existence of the such «false» collection is approximately 0.23 (for more details see Appendix A). It does not lead to the failure of the attack. We will be able to distinguish this case through the next step.

We also expect that for each of 15 Sboxes about 2 differences have integral property $\mathbf{0}$. For the last Sbox about 2^8 differences have such property. Thus, the set $S^{-1}(\Delta B \oplus S(\kappa))$ will contain about $2^{15} \cdot 2^8 = 2^{23}$ possible differences, each of them has the property $\mathbf{0}$.

4.2 Recovering of the round keys

Let's consider the last linear transformation. We know that $\Delta P'_5 \in SLS(\kappa)$ and $\Delta K'_5 \in SLS(\kappa) \oplus L^{-1}(\kappa)$. The difference before the linear transformation is the sum

$$\Delta P'_5 \oplus \Delta K'_5 \in (SLS(\kappa) \oplus SLS(\kappa) \oplus L^{-1}(\kappa)) = \{\delta_1 \oplus \delta_2 \oplus L^{-1}(\kappa), \delta_1 \in SLS(\kappa), \delta_2 \in SLS(\kappa)\}.$$

On the other hand we have the set of possible differences $S^{-1}(\Delta B \oplus S(\kappa))$ after the linear transformation.

The intersection of sets

$$(SLS(\kappa) \oplus SLS(\kappa) \oplus L^{-1}(\kappa)) \cap L^{-1}S^{-1}(\Delta B \oplus S(\kappa))$$

must contain at least one element. We use only one byte position to determine the inequality of elements from these two sets.

After checking the integral property in the set $L^{-1}S^{-1}(\Delta B \oplus S(\kappa))$ there will be about 2^{23} possible differences.

Recall that the set $S(\kappa)$ contains 2^8 elements. The linear transformation does not change the number of differences ($LS(\kappa)$ contains 2^8 elements). After another substitution layer we have 2^{16} possible differences at each Sbox. The difference κ is known, therefore $L^{-1}(\kappa)$ is also known. Consequently, the set $SLS(\kappa) \oplus SLS(\kappa) \oplus L^{-1}(\kappa)$ contains

$$\frac{2^{16} \cdot (2^{16} - 1)}{2} + 1 < 2^{31}$$

possible differences at each Sbox.

Select the position of one of the block bytes. Recall also that each difference contains $2^8 - 1$ vectors and consequently difference in one position contains $2^8 - 1$ bytes. We store in memory all possible differences from $SLS(\kappa) \oplus SLS(\kappa) \oplus L^{-1}(\kappa)$ for selected position. Let's iterate through all differences γ in $L^{-1}S^{-1}(\Delta B \oplus S(\kappa))$. If γ matches one of the stored differences then we assume that $\gamma = \Delta P'_5 \oplus \Delta K'_5$. We can expect that γ is the only such element even if we compare on eight bytes of a difference rather than $2^8 - 1$. Note that if the collection of the related keys is «false» ($\kappa'' \neq \kappa'$), the match will probably not be found.

At this step we know $\Delta P'_5 \oplus \Delta K'_5$, $L(\Delta P'_5 \oplus \Delta K'_5)$, $\Delta P'_6$, $\Delta K'_6$, ΔB . Block

$$Y : S^{-1}(\Delta P'_6) = L(\Delta P'_5 \oplus \Delta K'_5)$$

can be easily found. Let B_0 be first ciphertext, then $K'_6 = B_0 \oplus Y$. The entire set of related keys K'_6 can also be obtained by adding with $\Delta K'_6$. Therefore, it is possible to decipher all 2^8 ciphertexts through the last round.

We know that $\Delta P_4 \in \text{LS}(\kappa)$, $\Delta K'_5 \in \text{S}(\Delta K'_6) \oplus \text{L}^{-1}(\kappa)$ and also $\Delta P'_5 \oplus \Delta K'_5$. Let's iterate through possible $\tau \in \text{S}(\Delta K'_6) \oplus \text{L}^{-1}(\kappa)$ and propagate $\Delta P'_5 \oplus \Delta K'_5 \oplus \tau$ through S^{-1} . If we guess $\tau = \Delta K'_5$, then $\text{S}^{-1}(\Delta P'_5 \oplus \Delta K'_5 \oplus \tau) = \text{S}^{-1}(\Delta P'_5) \in \text{LS}(\kappa)$. Otherwise, we expect that $\text{S}^{-1}(\Delta P'_5 \oplus \Delta K'_5 \oplus \tau) \notin \text{LS}(\kappa)$. In the matching process, each Sbox can be viewed independently of the others. After that we will know the differences ΔP_4 , $\Delta P'_5$, $\Delta K'_5$. The ciphertexts after 5'th round are also known. Therefore, the keys K'_5 can be found in the same way as K'_6 . Due to the reverse key schedule, the master key $K = K_1 || K_2$ can be easily obtained.

4.3 Complexity

As mentioned before, the attack requires $1 + 2^8 \cdot (2^8 - 1) < 2^{16}$ related keys and one chosen ciphertext.

The integral property for all 2^8 related key collections can be checked in about $2^8 \cdot (15 \cdot 2^8 + 2^{16}) \approx 2^{24}$ operations.

The most time-consuming stage is the construction of the set $\text{SLS}(\kappa) \oplus \text{SLS}(\kappa) \oplus \text{L}^{-1}(\kappa)$. We construct this set for only one Sbox, and store only eight bytes for each difference. It requires about 2^{31} operations and $2^{31} \cdot 8 = 2^{34}$ bytes of memory. These constructed differences are stored in a hash table. The set $\text{L}^{-1}\text{S}^{-1}(\Delta B \oplus \text{S}(\kappa))$ contains much fewer elements. Checking for a single element in a hash table requires constant time. Therefore, the complexity of constructing the hash table will be the most important. The difficulty of recovering the keys K'_5 is also small: $16 \cdot 2^8 \cdot 2^8 + 2^8 \approx 2^{20}$ operations.

The total complexity does not exceed 2^{32} memory access operations and 2^{30} memory (in sixteen-byte blocks). We also note that the attack is deterministic.

We modeled the attack with a non-optimized C++ implementation. The average attack time is about 5 minutes on a common PC. The amount of used memory did not exceed 17 GB.

5 Conclusion

In this paper we present the related-key attack on 5-round Kuznyechik with 2-round key schedule. The attack has a practical complexity (2^{32} operations, 2^{30} memory, 2^{16} related keys) and has been verified with the help of C++ implementation. The experiments confirmed the correctness of the attack.

Source codes can be found at <https://gitlab.com/v.kir/rk-5R-kuznyechik>.

The main result was achieved by using the well-known integral property of LSX-transformations. We were able to use this property both in the cipher itself and in the key schedule.

We did not use any specific properties of the linear transformation and the Sbox. We think that through the use of such properties it is possible to obtain new results. Another possible way is the use of integral distinguishers for a greater number of rounds.

The presented attack also shows a significant security margin of the Kuznyechik's key schedule.

Acknowledgements

The author is grateful to Sergey Svetlov for help with experiments and verification, to Anton Naumenko and Igor Arbekov for support and valuable comments.

References

- [1] *GOST R 34.12-2015. National standard of the Russian Federation. Information technology Cryptographic data security Block ciphers*, 2015, in Russian.
- [2] E. Alekseev, K. Goncharenko, and Grigory Marshalko, “Provably Secure Counter Mode with Related Key-based Internal Re-keying”, *Pre-proceedings*, 7th Workshop on Current Trends in Cryptology (CTCrypt 2018), 2018.
- [3] T. Tiessen, “Polytopic Cryptanalysis”, *LNCS*, EUROCRYPT 2016, **9665**, ed. Fischlin M., Coron J.S., Springer, Berlin, Heidelberg.
- [4] Daemen J., Knudsen L., and Rijmen V., “The block cipher Square”, *LNCS*, FSE 1997, **1267**, ed. Biham E., Springer, Berlin, Heidelberg, 1997.
- [5] Barreto P. and Rijmen V., “The Khazad legacy-level block cipher”, First open NESSIE Workshop, 2000, Submission to NESSIE.
- [6] Biham E., “New types of cryptanalytic attacks using related keys (extended abstract)”, *LNCS*, EUROCRYPT 93, **765**, ed. Hellesteth T., Springer, Berlin, Heidelberg, 1993.

A Probability aspects and experimental verification

«True» and «false» collections of the related keys

We know that there is at least one «true» collection. What is the probability that the integral property (section 4.1) will be correct for the «false» collection?

Assume that all ciphertexts are equally probable and independent of each other. We propagate the difference of each Sbox thorough nonlinear layer. For each of 15 Sboxes we'll have 2^8 possible differences and for 16'th Sbox we get 2^{16} differences. We also assume that the sum of the elements of any difference is uniformly distributed. Hence, the probability of the property **0** is equal to $p = \frac{1}{256}$ for each difference of any Sbox. Denote the probability of the opposite event by $q = 1 - p = \frac{255}{256}$.

Thus, we have:

$q^{2^8} = 0.367\dots$ – there is no difference that has the property **0** for one Sbox;

$1 - q^{2^8} = 0.632\dots$ – there is at least one such difference;

$(1 - q^{2^8})^{15} = 0.001\dots$ – there is at least one such difference for each of the 15 Sboxes.

The probability that one collection of the related keys has the integral property is

$$r = \left(1 - q^{2^8}\right)^{15} \cdot \left(1 - q^{2^{16}}\right) = 0.001\dots$$

We have 2^8 collections of keys and only one «true» collection. The probability that «false» collections do not exist is

$$(1 - r)^{255} = 0.765\dots$$

The opposite probability is

$$1 - (1 - r)^{255} = 0.234\dots$$

We performed $N = 5000$ experiments. The number of cases where collections exist is equal to 1179. The obtained value $\frac{1179}{5000} = 0.236$ is close to theoretical.

Number of possible differences

Let we have «true» collection of the related keys. We estimate the number of possible differences in the set $L^{-1}S^{-1}(\Delta B \oplus S(\kappa))$.

Each Sbox gives at least one possible difference. The probability of the property $\mathbf{0}$ is equal to $p = \frac{1}{256}$ for each difference of any Sbox. We also have 2^8 possible differences for each of 15 Sboxes and 2^{16} for 16'th Sbox.

Thus, average number of elements in the set is equal to

$$\left(1 + \frac{1}{256} \cdot (2^8 - 1)\right)^{15} \cdot \left(1 + \frac{1}{256} \cdot (2^{16} - 1)\right) \approx 2^{23} \ll 2^{31}.$$

The average experimental value is $2^{22.7}$. The maximum value among all N experiments is 2^{29} .

Matching differences

The intersection of sets

$$(SLS(\kappa) \oplus SLS(\kappa) \oplus L^{-1}(\kappa)) \cap L^{-1}S^{-1}(\Delta B \oplus S(\kappa))$$

must contain at least one element. We use only one position to determine the inequality of elements from these two sets.

One position of the first set contains no more than 2^{31} differences. The number of elements of the second set is approximately 2^{23} . We also assume that the elements of these sets are random and equally probable.

Only the first 8 bytes (64 bits) of the difference are stored in memory. Then the average number of «false» matches can be estimated as

$$\frac{2^{31} \cdot 2^{23}}{2^{64}} = 2^{-10}.$$

A «false» match can be easily detected by an additional check. In $N = 5000$ experiments, we got only 7 cases of it.

Eight-byte numbers were chosen for ease of implementation.

B Detailed pictures

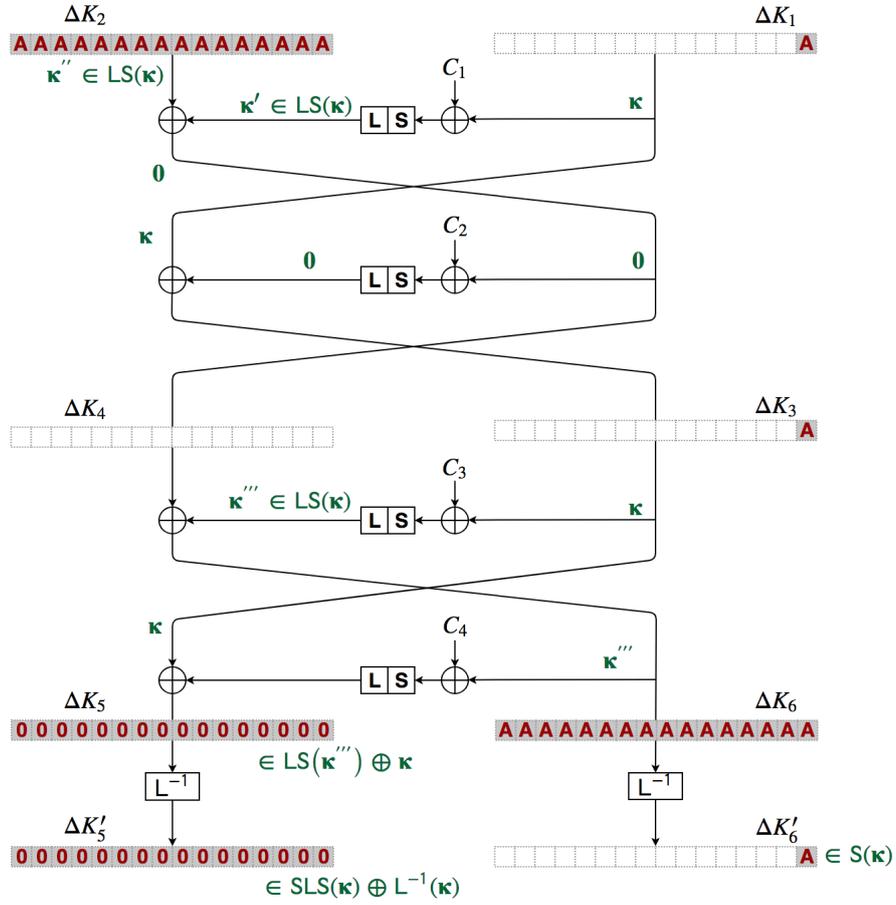


Figure 2: The difference propagation through the key schedule

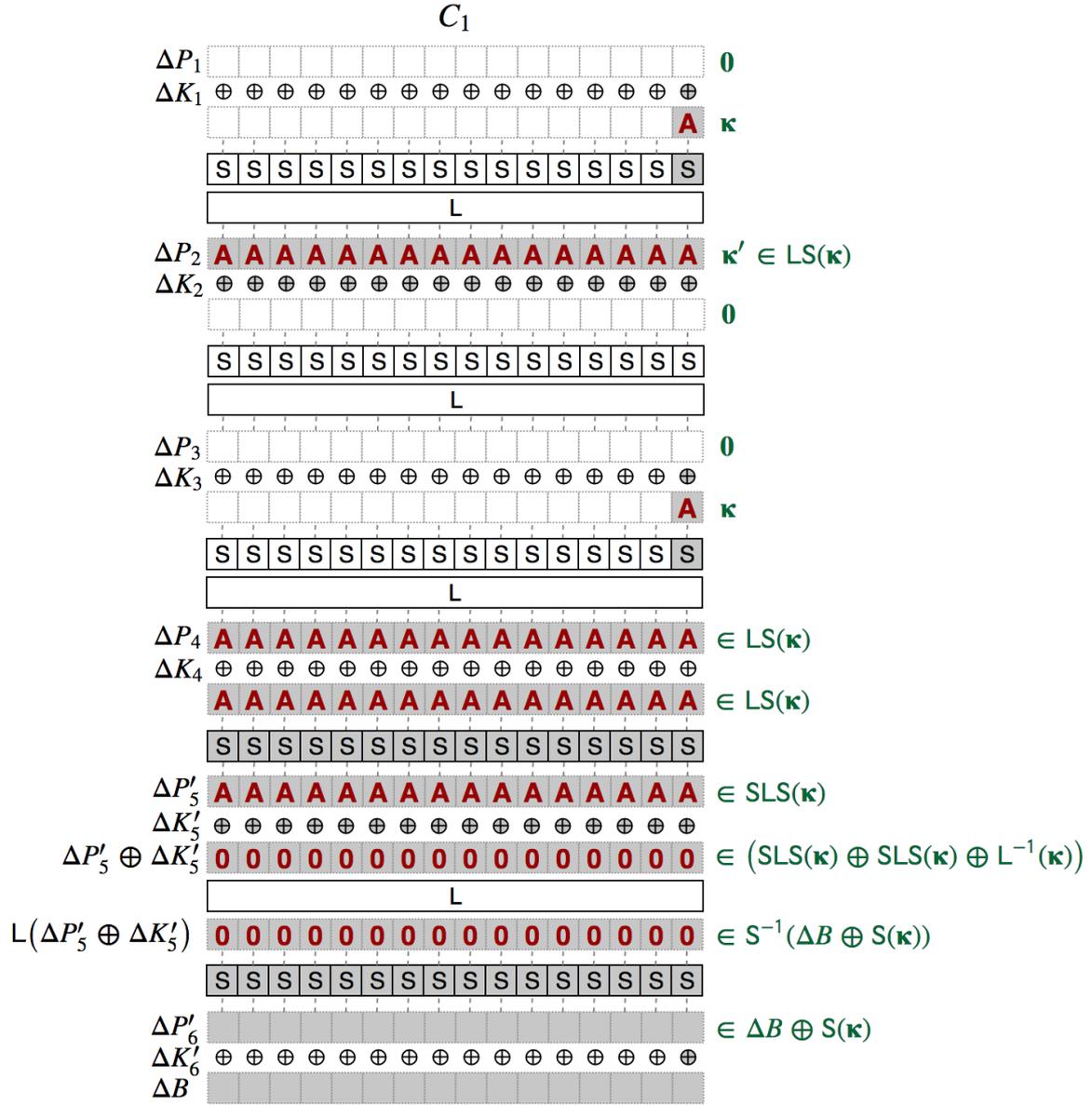


Figure 3: The difference propagation through the cipher