A New Structural-Differential Property of 5-Round AES

Lorenzo Grassi[1(✉)], Christian Rechberger[1,3], and Sondre Rønjom[2,4]

[1] IAIK, Graz University of Technology, Graz, Austria
{lorenzo.grassi,christian.rechberger}@iaik.tugraz.at
[2] Nasjonal sikkerhetsmyndighet, Oslo, Norway
[3] DTU Compute, DTU, Kongens Lyngby, Denmark
[4] Department of Informatics, University of Bergen, Bergen, Norway
Sondre.Ronjom@ii.uib.no

Abstract. AES is probably the most widely studied and used block cipher. Also versions with a reduced number of rounds are used as a building block in many cryptographic schemes, e.g. several candidates of the SHA-3 and CAESAR competition are based on it.

So far, non-random properties which are independent of the secret key are known for up to 4 rounds of AES. These include differential, impossible differential, and integral properties.

In this paper we describe a *new structural property for up to 5 rounds of AES*, differential in nature and which is independent of the secret key, of the details of the MixColumns matrix (with the exception that the branch number must be maximal) and of the SubBytes operation. It is very simple: By appropriate choices of difference for a number of input pairs it is possible to make sure that the number of times that the difference of the resulting output pairs lie in a particular subspace is *always* a multiple of 8.

We not only observe this property experimentally (using a small-scale version of AES), we also give a detailed proof as to why it has to exist. As a first application of this property, we describe a way to distinguish the 5-round AES permutation (or its inverse) from a random permutation with only 2^{32} chosen texts that has a computational cost of $2^{35.6}$ look-ups into memory of size 2^{36} bytes which has a success probability greater than 99%.

Keywords: Block cipher · Permutation · AES · Secret-key distinguisher

1 Introduction

Block ciphers play an important role in symmetric cryptography providing the basic tool for encryption. They are the oldest and most scrutinized

The extended version of this paper can be found in [13]. It includes a more formal description of the main result of this paper which exploits the subspace trail notation [14] recently introduced at FSE 2017.

J.-S. Coron and J.B. Nielsen (Eds.): EUROCRYPT 2017, Part II, LNCS 10211, pp. 289–317, 2017.
DOI: 10.1007/978-3-319-56614-6_10

cryptographic tools. Consequently, they are the most trusted cryptographic algorithms that are often used as the underlying tool to construct other cryptographic algorithms, whose proofs of security are performed under the assumption that the underlying block cipher is ideal.

While the security of public-key encryption schemes are related to the hardness of well-defined mathematical problems, informally a block cipher is considered secure if an (efficient) adversary, with access to the encryptions of messages of its choice, cannot tell apart those encryptions from the values of a truly random permutation. In other words, this means that an (efficient) adversary, with access to the encryptions of messages of its choice, cannot tell the difference between the block cipher (equipped with a random key) and a truly random permutation. This notion of block cipher security was introduced and formally modeled by Luby and Rackoff [19] in 1988, and it was motivated by the design of DES. To be a bit more precise (but without going into the details), a secret key distinguisher is one of the weakest cryptographic attacks that can be launched against a secret-key cipher. In this attack, there are two oracles: one that simulates the cipher for which the cryptographic key has been chosen at random and the other simulates a truly random permutation. The adversary can query both oracles and his task is to decide which oracle is the cipher and which is the random permutation. The attack is considered to be successful if the number of queries required to make a correct decision is below a well defined level.

The Rijndael block cipher [8] has been designed by Daemen and Rijmen in 1997 and was chosen as the AES (Advanced Encryption Standard) by NIST in 2000. Nowadays, it is probably the most used and studied block cipher. The possibility to set up a *secret key distinguisher for 5-round of AES* that exploits a property which is *independent of the secret key* was already considered in [21] and improved in [14]. However, only partial solutions have been proposed and the problem is still open. As we will argue below, the solutions so far are partial because the distinguishers are derived from a key-recovery attack and they actually exploit as property the existence of a sub-key for which a property on 4 rounds holds.

In this paper, we present (and practical verify) the *first secret-key distinguisher for 5-round AES* which exploits a new structural/differential property which is independent of the secret key, that is a property that can be practically verified without needing to know or to get to know any information of the secret key. As we are going to show, it requires 2^{33} chosen plaintexts/ciphertexts and has a computational cost of $2^{36.6}$ table look-ups.

1.1 Secret-Key Distinguishers for AES-128

In the usual security model, the adversary is given a *black box* (oracle) access to an instance of the encryption function associated with a random secret key and its inverse. The goal is to find the key or more generally to efficiently distinguish the encryption function from a random permutation.

More formally, a block cipher is a family of functions $E : \mathcal{K} \times \mathcal{S} \to \mathcal{S}$, with $\mathcal{K}$ a finite set called the key space and $\mathcal{S}$ a finite set called the domain or message space. For every $k \in \mathcal{K}$, the function $E_k(\cdot) = E(k, \cdot)$ is a permutation. The inverse of the block cipher E is defined as a function $E^{-1} : \mathcal{K} \times \mathcal{S} \to \mathcal{S}$ that satisfies $E_k^{-1}(E_k(s)) = s$ for each $k \in \mathcal{K}$ and for each $s \in \mathcal{S}$. A block cipher $E_k(\cdot)$ with key space $\mathcal{K}$ is a (q, t, ε)-pseudorandom permutation (PRP) if any adversary making at most q oracle queries and running in time at most t can distinguish E_k (for a random key k) from a uniformly random permutation with advantage at most ε.

Definition 1. *Let E be block cipher defined as before, and $Perm(\mathcal{S})$ be the set of all permutations of $\mathcal{S}$. Let D be a distinguisher with oracle access to a permutation and its inverse, and returning a single bit. The (Strong PseudoRandom Permutation) SPRP-advantage of D against E is defined as*

$$\begin{aligned}\mathbf{Adv}_E^{sprp}(D) = |&Prob(\pi \leftarrow Perm(\mathcal{S}) \,:\, D^{\pi(\cdot),\pi^{-1}(\cdot)} = 1) \\ &- Prob(k \leftarrow \mathcal{K} \,:\, D^{E_k(\cdot),E_k^{-1}(\cdot)} = 1)|.\end{aligned}$$

For integers q and t, the SPRP-advantage of E is defined as

$$\mathbf{Adv}_E^{sprp}(q, t) = \max_D \mathbf{Adv}_E^{sprp}(D),$$

where the maximum is taken over all distinguishers making at most q oracle queries and running in time at most t. E is a (q, t, ε)-SPRP if $\mathbf{Adv}_E^{sprp}(q, t) \leq \varepsilon$.

Note that if $Adv_E(D) \simeq 0$, then the $E_k(\cdot)$ behaves (exactly) like a random permutation from the distinguisher point of view.

Before we focus on the 5-round distinguisher, we briefly summarize the properties exploited by the secret key distinguisher on AES-like permutations up to 4 rounds. We stress that, even if a key-recovery attack can also be used as a secret key distinguisher in this paper we focus only on secret-key distinguisher that are independent of the secret key.

The most competitive secret-key distinguishers up to 3-round are based on the differential [5] and on the truncated differential cryptanalysis [17]. These distinguishers exploit the fact that some r-round differential characteristics exist with higher probability for an AES permutation than for a random one. In [7], Daemen *et al.* proposed an attack vector that uses a 3-round distinguisher to attack up to 6 rounds of the cipher and later became known as integral attacks. In an integral distinguisher, given inputs with particular properties, one exploits the fact that the sum of the corresponding ciphertexts is zero with probability 1 for an AES permutation, while this happens with a (much) lower probability for a random permutation. Finally, another possible distinguisher exploits the impossible-differential cryptanalysis, which was independently proposed by Knudsen [18] and by Biham *et al.* [3]. In impossible-differential cryptanalysis, the idea is to exploit the fact that some differential trails hold with probability 0 for an AES permutation (i.e. impossible differential trails), while they have probability greater than 0 for a random permutation.

5-Round "Distinguisher" for AES-128: State of Art. A distinguisher for five rounds of AES-128 has been recently proposed by Sun, Liu, Guo, Qu, and Rijmen at Crypto 2016 [21]. This distinguisher - which requires the *whole* input-output space to work - has been improved in [14], where authors set up a secret key distinguisher in the same setting of the one proposed in [21], but which requires only $2^{98.2}$ chosen plaintexts.

Both these two distinguishers are derived by a key-recovery attack on AES-128 with a secret S-Box. In particular, they are able to distinguish a random permutation from an AES one exploiting the existence of a (secret) key for which a property on 4-round is verified. In more details, the property on 4-round used in [21] is the balance property, while the one used in [14] is the impossible differential one. With respect to a classical key-recovery attack, these distinguishers require the knowledge only of a single byte of the secret subkey to distinguish an AES permutation with a secret S-Box from a random one.

For a complete comparison with the distinguisher presented in this paper, we briefly recall how they are set up, and we refer to [14,21] for a complete discussion. In both cases, authors first assume to know the difference of two bytes (i.e. 1 byte) of one secret subkey. Using this knowledge, they are able to extend a four rounds distinguisher to five rounds. In order to turn these distinguishers into secret-key ones, the idea is simply to iterate these distinguishers on all the 2^8 possible values of the difference of these two bytes of the secret subkey. The idea is that for an AES permutation there exists one difference of these two bytes for which a property (which is independent of the secret key) on 4-round is satisfied, while for a random permutation this property on 4-round is never satisfied (with high probability) for any of the 2^8 possible values.

We stress that both these distinguishers require to find part of the secret key in order to verify a property on 4-round, i.e. they work as key-recovery attacks. Note that the research of a secret-key distinguisher which is independent of the secret key is of particular interest and importance since it (theoretically) allows to set up key recovery attacks, as it already happened for the secret-key distinguishers up to 4 rounds just described. Moreover, we highlight that both these distinguishers are independent of the details of the S-Box, but they depend on the details of the MixColumns matrix (in particular, they exploit the fact that for at least one column of the MixColumns matrix or its inverse two elements are identical).

1.2 Our Result: The First 5-Round Secret-Key Distinguisher for AES-128 Independent of the Secret Key

The results presented in the previous two papers don't solve the problem to set up *a 5-round secret key distinguisher of AES which exploits a property which is independent of the secret key.* In Sect. 3 of this paper, we provide a solution to this problem, that is we propose the *first* secret-key distinguisher on 5-round of AES which exploits a new property which is independent of the secret key and of the details of the S-Box.

The high-level idea is very easily described. By appropriate choices of difference for a number of input pairs it is possible to make sure that the number of times that the difference of the resulting output pairs assumes certain values is *always* a multiple of 8. More concretely, given a set of plaintexts which are equal in certain bytes, consider the corresponding ciphertexts after 5 rounds. The idea is to count the total number of different ciphertext pairs with zero-difference in certain bytes. As we show in detail in the paper, for an AES permutation this number can only be a multiple of 8, while it does not have any particular property for the case of a random permutation. As we will see in the comparison, the resulting distinguisher proposed in this paper is much more efficient than those proposed earlier, it *works both in the encryption and in the decryption mode of AES and it does not depend on the details of the MixColumns matrix (with the exception that the branch number must be five) or/and of the SubBytes operation.* A formal statement of this property used by our distinguisher is given in Theorem 1 in Sect. 3.1, and its detailed proof is given in Sect. 4.

Comparison with 4-Round Secret-Key Distinguishers. These last properties also highlight a difference between our new distinguisher and the others currently used in literature. In most cases, especially in the cryptanalysis of AES, one does not have the necessity to investigate the details of the S-Boxes. Consider for example the 4-round secret-key distinguishers, based on the integral [12] and on the impossible-differential [4] properties. In the first one, given a set of chosen plaintexts of which part is held constant and another part varies through all possibilities, it is possible to prove that their XOR-sum after 4-round is always equal to 0. In the second one, given the same set of chosen plaintexts, it is possible to prove that the difference of each possible pair of ciphertexts after 4-round can not take some values (some differences have prob. 0, i.e. they are impossible). In both cases, the corresponding results are independent of the key and of the non-linear components. That is, if some other S-Boxes with similar differential/linear properties are chosen in a cipher, the corresponding cryptanalytic results remain the same.

Although there are already 4-round impossible differentials and zero-correlation linear hulls for AES, the effort to find new impossible differentials and zero-correlation linear hulls that could cover more rounds has never been stopped. In Eurocrypt 2016, Sun *et al.* [22] proved that, unless the details of the S-Boxes are exploited, one cannot find any impossible differential or zero-correlation linear hull of the AES that covers 5 or more rounds. Moreover, due to the link among impossible differential, integral and zero correlation linear cryptanalysis [23], an analogous result holds also for the integral case. On the other hand, our new property presented in this paper holds up to 5-round of AES independently of the key and of the details of the S-Box (and of the MixColumns operation), and allows to answer an almost 20-year old problem: given a set of chosen plaintexts similar to the one used by the integral and impossible differential distinguishers just recalled, is there any property which is independent of the secret key after 5-round AES?

Table 1. *5-round Secret-Key Distinguishers for AES with a Single Secret S-Box.* In this table, we limit ourselves to consider the distinguishers that exploit a property which is independent of the key, or which are derived by a key-recovery attack but are independent of the S-Box and require the knowledge of only part of the key. The complexity is measured in minimum number of chosen plaintexts CP or/and chosen ciphertexts CC which are needed to distinguish the AES permutation from a random one with probability higher than 99%. Time complexity is measured in memory accesses (M) or XOR operations (XOR). The case in which the final MixColumns operation is omitted is denoted by "r.5 round", i.e. r full rounds and the final one. "Key-Independence" denotes a distinguisher which is able to distinguish 5-round AES from a random permutation without discovering any information of the secret key or of part of it.

Property	Rounds	Data	CP	CC	Cost	Key-Independence	Ref.
Structural Diff.	$\mathbf{4.5-5}$	$\mathbf{2^{33}}$	✓	✓	$\mathbf{2^{36.6}}$ **M**	✓	**Sect.** 3
Impossible Diff.	$4.5-5$	$2^{98.2}$	✓		2^{107} M		[14]
Integral	5	2^{128}		✓	2^{128} XOR		[21]

Comparison of 5-Round Secret-Key Distinguishers. For a better comparison between this new secret-key distinguisher proposed in this paper and earlier ones, we propose to classify the secret-key distinguishers in the following way (from strongest to weakest):

1. a distinguisher which is completely independent of the secret key (e.g., it exploits properties that are not related to the existence of a key) and independent of the details of the S-Box;
2. a distinguisher which depends on the existence of a key and is derived by a key-recovery attack.

A comparison between our new distinguisher and the ones proposed in [14,21] is given in Table 1, where "Key-Independence" denotes a secret-key distinguisher which is derived by a key-recovery attack, i.e. that does not exploit a property which is independent of the secret key. Moreover, with respect to the previous classification, a complete comparison of all the secret key distinguishers and key-recovery attacks (used as distinguishers) for 5-round AES is provided in Table 2 - Appendix C of the full version of the paper [13].

2 Preliminary - Description of AES

The Advanced Encryption Standard [8] is a *Substitution-Permutation network* that supports key size of 128, 192 and 256 bits. The 128-bit plaintext initializes the internal state as a 4×4 matrix of bytes as values in the finite fields $\mathbb{F}_{256}$, defined using the irreducible polynomial $x^8 + x^4 + x^3 + x + 1$. Depending on the version of AES, N_r round are applied to the state: $N_r = 10$ for AES-128, $N_r = 12$ for AES-192 and $N_r = 14$ for AES-256. An AES round applies four operations to the state matrix:

- *SubBytes* (S-Box) - applying the same 8-bit to 8-bit invertible S-Box 16 times in parallel on each byte of the state (it provides non-linearity in the cipher);
- *ShiftRows* (SR) - cyclic shift of each row to the left;
- *MixColumns* (MC) - multiplication of each column by a constant 4×4 invertible matrix M_{MC} (MC and SR provide diffusion in the cipher[1]);
- *AddRoundKey* (ARK) - XORing the state with a 128-bit subkey.

One round of AES can be described as $R(x) = K \oplus MC \circ SR \circ$ S-Box(x). In the first round an additional AddRoundKey operation (using a whitening key) is applied, and in the last round the MixColumns operation can be omitted. For the following, we *assume that the last MixColumns operation is always omitted.* In the case in which the last MixColumns is not omitted, it is sufficient to exchange the order of the last MixColumns operation and of the AddRoundKey operation - they are linear.

Finally, as we don't use the details of the AES key schedule in this paper, we refer to [8] for a complete description.

The Notation Used in the Paper. Let x denote a plaintext, a ciphertext, an intermediate state or a key. Then $x_{i,j}$ with $i, j \in \{0, ..., 3\}$ denotes the byte in the row i and in the column j. We denote by k^r the key of the r-th round, where k^0 is the secret key. If only the key of the final round is used, then we denote it by k to simplify the notation. Finally, we denote by R one round of AES, while we denote r rounds of AES by R^r. We sometimes use the notation R_K instead of R to highlight the round key K. If the MixColumns operation is omitted in the last round, then we denote it by R_f.

2.1 Differential Trail over 2-round AES

For the following, we recall a 2-round truncated differential trail of AES (see [9] or [10] for details), largely used in the paper and illustrated in Fig. 1.

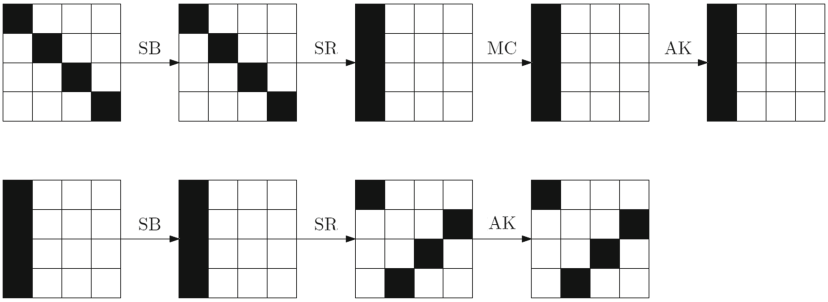

Fig. 1. Differential Trail over 2-round AES.

[1] SR makes sure column values are spread, MC makes sure each column is mixed.

Let $R^2(\cdot)$ denote two AES rounds with fixed random round keys. Consider two plaintexts which are equal in all bytes except for the ones in the i-th diagonal for a certain $i = 0, 1, 2, 3$, i.e. for the bytes in row j and column $i + j$ for each $j = 0, 1, 2, 3$ (the index $i + j$ is taken modulo 4). After one round, the two texts are equal in all bytes except for the ones in the i-th column, i.e. for the bytes in row j and column i for each j. After the second and last round - assuming the final MixColumns is omitted, the two texts are equal in all bytes except for the ones in the i-th anti-diagonal, that is for the bytes in row j and column $i - j$ for each j (the index $i - j$ is taken modulo 4) by definition of anti-diagonal.

For the following, we work with *diagonal sets* of 2^{32} plaintexts, defined as sets of texts which are equal in 3 diagonals, i.e. texts with active bytes in the i-th diagonal for a certain $i = 0, 1, 2, 3$ and with constant bytes in the other three:

$$\begin{bmatrix} A\,C\,C\,C \\ C\,A\,C\,C \\ C\,C\,A\,C \\ C\,C\,C\,A \end{bmatrix} \xrightarrow{R(\cdot)} \begin{bmatrix} A\,C\,C\,C \\ A\,C\,C\,C \\ A\,C\,C\,C \\ A\,C\,C\,C \end{bmatrix} \xrightarrow{R_f(\cdot)} \begin{bmatrix} A\,C\,C\,C \\ C\,C\,C\,A \\ C\,C\,A\,C \\ C\,A\,C\,C \end{bmatrix},$$

where A denotes an active byte (i.e. a byte in which every value in $\mathbb{F}_{2^8}$ appears the same number of times) and C denotes a constant byte (i.e. a byte in which the value is fixed to a constant for all texts). For completeness, we label the last set by *inverse-diagonal set*, i.e. a set of texts where the bytes in one (or more) anti-diagonal(s) are active while the others are constant.

If the final MixColumns is not omitted, certain linear relations - which are given by the definition of the MixColumns matrix - hold between the bytes of the texts that lie in the same column:

$$\begin{bmatrix} A\,C\,C\,C \\ C\,A\,C\,C \\ C\,C\,A\,C \\ C\,C\,C\,A \end{bmatrix} \xrightarrow{R(\cdot)} \begin{bmatrix} A\,C\,C\,C \\ A\,C\,C\,C \\ A\,C\,C\,C \\ A\,C\,C\,C \end{bmatrix} \xrightarrow{R(\cdot)} MC \times \begin{bmatrix} A\,C\,C\,C \\ C\,C\,C\,A \\ C\,C\,A\,C \\ C\,A\,C\,C \end{bmatrix},$$

In this case, we label the last set by *mixed set*. As an example, consider two plaintexts p^1 and p^2 which are equal in all bytes except for the ones in the 0-th diagonal, i.e. except for the bytes in positions (j, j) for each $j = 0, 1, 2, 3$. After 2 (complete) rounds, there exist $x, y, z, w \in F_{2^8}$ such that their difference $R^2(p^1) \oplus R^2(p^2)$ can be re-written as:

$$R^2(p^1) \oplus R^2(p^2) = \begin{bmatrix} 0\times 02 \cdot x & y & z & 0\times 03 \cdot w \\ x & y & 0\times 03 \cdot z & 0\times 02 \cdot w \\ x & 0\times 03 \cdot y & 0\times 02 \cdot z & w \\ 0\times 03 \cdot x & 0\times 02 \cdot y & z & w \end{bmatrix}. \qquad (1)$$

Finally, the same truncated differential analysis of 2-round can be generalized to the cases of an initial diagonal set with more than a single active diagonal, i.e. a set of plaintexts which are equal in all bytes except for the ones that lie in two or three diagonals (instead of only one).

3 New 5-round Secret Key Distinguisher for AES

3.1 Statement of the Property

Consider a diagonal set of plaintexts - i.e. a set of 2^{32} plaintexts which are equal in all bytes except for the ones in i-diagonal for a certain $i = 0, 1, 2, 3$, and the corresponding ciphertexts after 5 rounds. Assume the final MixColumns operation is omitted. In order to set up the distinguisher on 5 rounds of AES, the idea is to count the number of different pairs of ciphertexts which are equal in d anti-diagonals for a certain $1 \leq d \leq 3$ - that is the number of pairs of ciphertexts with zero-difference in the bytes in positions $(i, j - i)$ for all $i = 0, 1, 2, 3$ and $j \in J$ for a certain $J \subseteq \{0, 1, 2, 3\}$ with $|J| = d$ - and to exploit the property that for an AES-like permutation this number is a multiple of 8 with prob. 1.

In more detail, given a set of plaintexts/ciphertexts (p^i, c^i) for $i = 0, ..., 2^{32} - 1$ - where all the plaintexts are in the same diagonal set, the idea is to construct all the possible pairs of ciphertexts (c^i, c^j) for $i \neq j$ and to count the number of different pairs[2] of ciphertexts (c^i, c^j) for which the bytes of the difference $c^i \oplus c^j$ that lie in d anti-diagonals are equal to zero (where $1 \leq d \leq 3$ and the anti-diagonals are fixed in advance). It is possible to prove that for 5-round AES this number has the special property to be a multiple of 8 independently of d - that is on the number of considered anti-diagonals. Instead, for a random permutation the same number does not have any special property (e.g. it has the same probability to be even or odd). This allows to distinguish 5-round AES from a random permutation.

Theorem 1. *Given 2^{32} plaintexts in the same diagonal set defined as before, consider the corresponding ciphertexts after 5 rounds, that is (p^i, c^i) for $i = 0, ..., 2^{32} - 1$ where $c^i = R^5(p^i)$ The number n of different pairs of ciphertexts (c^i, c^j) for $i \neq j$ for which the bytes of the difference $c^i \oplus c^j$ that lie in d anti-diagonals are equal to zero (where $1 \leq d \leq 3$ and the anti-diagonals are fixed in advance) is a multiple of 8, that is $\exists\, n' \in \mathbb{N}$ such that $n = 8 \cdot n'$.*

Idea of the Proof - Lemma 1. As we have seen in the previous section, a diagonal set is always mapped after two rounds into a mixed set. In other words, if two plaintexts have equal bytes expect for the ones in one diagonal, then after two rounds some particular linear relationships (given in (1)) hold among the bytes of the difference of these two texts that lie in the same column with probability 1. In the same way, if two ciphertexts have equal bytes in d anti-diagonals, then these two texts have equal bytes in d diagonals two rounds before (due to the 2-round differential trail described in Sect. 2.1). In other words, a inverse-diagonal set is mapped into a diagonal set two rounds before (assuming the final MixColumns operation is omitted).

Assume for simplicity that the 2^{32} plaintexts are chosen in a diagonal set with the active bytes in the first diagonal (analogous for the other cases).

[2] The two pairs (c^i, c^j) and (c^j, c^i) are considered equivalent. To formalize this concept, one can consider the number of ciphertexts (c^i, c^j) *with* $i < j$ for which the bytes of the difference $c^i \oplus c^j$ that lie in d anti-diagonals are equal to zero.

Due to these two previous considerations, Theorem 1 on 5 rounds of AES (and its proof) is strongly related to the following lemma on 1-round AES.

Lemma 1. *Given* 2^{32} *plaintexts in a mixed set of the form*

$$MC \cdot \begin{bmatrix} A & C & C & C \\ C & C & C & A \\ C & C & A & C \\ C & A & C & C \end{bmatrix}, \tag{2}$$

consider the corresponding ciphertexts after 1 round, that is $(\hat{p}^i, \hat{c}^i)$ *for* $i = 0, ..., 2^{32} - 1$ *where* $\hat{c}^i = R(\hat{p}^i)$*. The number* n *of different pairs of ciphertexts* $(\hat{c}^i, \hat{c}^j)$ *for* $i \neq j$ *for which the bytes of the difference* $c^i \oplus c^j$ *that lie in* d *diagonals are equal to zero (where* $1 \leq d \leq 3$ *and the diagonals are fixed in advance) is a multiple of* 8*, that is* $\exists\, n' \in \mathbb{N}$ *s.t.* $n = 8 \cdot n'$*.*

The complete proof is provided in the next section - Sect. 4. We emphasize that the proof of Theorem 1 follows immediately by the proof of Lemma 1, due to the 2-round truncated differential trail described in Sect. 2.1. In particular, note that considering 2^{32} plaintexts in the same diagonal set (that is 2^{32} plaintexts which are equal in three diagonals and with active bytes in the other one) is equivalent to consider 2^{32} texts in the same mixed set as defined in (2) after two rounds. In other words, all 2^{32} plaintexts of Lemma 1 are definitely reachable in 2 rounds from the initial plaintext (diagonal) structure defined in Theorem 1.

To prove the lemma, the idea is show that given one pair of ciphertexts for which the bytes that lie in d diagonals are equal, then also other pairs of ciphertexts have the same property with probability 1. The complete proof is given in Sect. 4. We highlight that the statement given in Theorem 1 (or Lemma 1) does not depend on the details of the MixColumns matrix (with the exception that the branch number must be five) or/and of the SubBytes operation. In other words, the only property that the proof - given in the next section - exploits is the branch number of the MixColumns matrix.

3.2 Setting up the Distinguisher

Our 5-round distinguisher exploits the property just described that the above defined number n is a multiple of 8 for 5-round AES, while it can take any possible value in the case of a random permutation. In the following we show how to set up the previous distinguisher in an efficient way for the case $d = 1$ (analougos for the other cases).

To implement the distinguisher, one has to count the number of pairs of ciphertexts for which the difference in $d = 1$ anti-diagonal is equal to zero (where this anti-diagonal is fixed in advance). First of all, since the probability that two ciphertexts satisfy this property is 2^{-32} (in general, $2^{-32 \cdot d}$ for d anti-diagonals), we expect that *on average*

$$\binom{2^{32}}{2} \cdot 2^{-32} = 2^{31} \cdot (2^{32} - 1) \cdot 2^{-32} \simeq 2^{31}$$

different pairs of ciphertexts have difference zero in one fixed anti-diagonal both for an AES permutation and for a random one. However, while for an AES permutation this number is a multiple of 8 with probability 1, for a random permutation this happens only with probability $0.125 \equiv 2^{-3}$. In particular, consider s initial arbitrary diagonal sets of plaintexts and for each of them count the number of different pairs of ciphertexts that have difference zero in d anti-diagonals. For an AES permutation, each of these numbers is a multiple of 8, while the probability that this happens for a random permutation is only $2^{-3 \cdot s}$. In order to distinguish the AES permutation from the random one with probability at least pr, it is sufficient that for a random permutation at least one of these numbers is not a multiple of 8, which happens with probability pr:

$$pr = 1 - 2^{-3 \cdot s}.$$

Thus, the probability of success of this distinguisher is greater than 99% (i.e. $pr \geq 0.99$) for $s \geq 3$. Note that for each initial diagonal set, one can count the above defined number n for each one of the four possible anti-diagonals. In other words, there are four different anti-diagonals for which one can count the number n of pairs of ciphertexts with zero difference in that anti-diagonal. It follows that using a single initial diagonal set, it is possible to distinguish 5-round AES from a random permutation with a probability of success of approximately $1 - (2^{-12}) = 99.975\%$.

In conclusion, 2^{32} chosen plaintexts in a single initial arbitrary diagonal set - i.e. a set of 2^{32} plaintexts which are equal in all bytes except for the ones in the i-th diagonal for a certain $i = 0, 1, 2, 3$ - are sufficient to distinguish a random permutation from an AES one. An approximation of the computational cost is given in the following. For completeness, it is also possible to set up a distinguisher for the cases $d = 2$ and $d = 3$ - i.e. the cases in which one count the number n of pairs of ciphertexts for which the bytes in $d = 2, 3$ anti-diagonals are equal. However, it should be noticed that the average number of collisions in these cases are respectively $2^{31} \cdot (2^{32} - 1) \cdot 2^{-64} \simeq 2^{-1}$ and $2^{31} \cdot (2^{32} - 1) \cdot 2^{-96} \simeq 2^{-33}$. As a consequence, the data and computational cost of these cases is not lower than for the case $d = 1$.

3.3 The Computational Cost

We have just seen that 2^{32} chosen plaintexts in a single diagonal set are sufficient to distinguish a random permutation from 5 rounds of AES, simply counting the number of pairs of ciphertexts with equal bytes in d anti-diagonal and checking if it is a multiple of 8 or not. Here we give an estimation of the computational cost of the distinguisher, which is approximately given by the sum of the cost to construct all the pairs and of the cost to count the number of pairs of ciphertexts with the previous property. As a result, the total computational cost can be well approximated by $2^{35.6}$ table look-ups.

Assume the final MixColumns operation is omitted. As we have just said, for each initial diagonal set the two steps of the distinguisher are (1) construct all the possible pairs of ciphertexts and (2) count the number of collisions. First of all, given pair of ciphertexts, note that the cost to check that the bytes in d anti-diagonals are equal corresponds to the cost of a XOR operation[3]. As we are going to show, the major cost of this distinguisher regards the construction of all the possible different pairs, which corresponds to step (1). Since it is possible to construct approximately 2^{63} pairs for each initial diagonal set, the simplest way to do it requires 2^{63} table look-ups. In the following, we present a way to reduce the total cost to approximately $2^{35.6}$ table look-ups, where the used tables are of size 2^{32} texts (or equivalently $2^{32} \cdot 16 = 2^{36}$ byte).

The basic idea is to implement the distinguisher using a *data structure*. The goal is to count the number of pairs of ciphertexts (c^1, c^2) for which the bytes in one of the anti-diagonal are equal, that is such that for a fixed $j \in \{0, 1, 2, 3\}$ the following condition is satisfied:

$$c^1_{i,j-i} = c^2_{i,j-i} \qquad \forall i = 0, 1, 2, 3 \tag{3}$$

where the index is computed modulo 4. To do this, consider an array A of 2^{32} elements completely initialized to zero. The element of A in position x for $0 \leq x \leq 2^{32} - 1$ - denote by $A[x]$ - represents the number of ciphertexts c that satisfy the following equivalence (in the integer field $\mathbb{N}$):

$$x = c_{0,0-j} + 256 \cdot c_{1,1-j} + c_{2,2-j} \cdot 256^2 + c_{3,3-j} \cdot 256^3.$$

It's simple to observe that if two ciphertexts c^1 and c^2 satisfy (3), then they increment the same element x of the array A. It follows that given $r \geq 0$ texts that increment the same element x of the array A, then it is possible to construct

$$\binom{r}{2} = \frac{r \cdot (r-1)}{2}$$

different pairs of texts that satisfy (3). The complete pseudo-code of such an algorithm is given in Algorithm 1.

What is the total computational cost of this procedure? Given a set of 2^{32} (plaintexts, ciphertexts) pairs, one has first to fill the array A using the strategy just described, and then to compute the number of total of pairs of ciphertexts that satisfy the property, for a cost of $3 \cdot 2^{32} = 2^{33.6}$ table look-ups - each one of these three operations require 2^{32} table look-ups. Since one has to repeat this algorithm 4 times - one time for each one of the four anti-diagonal, the total cost is of $4 \cdot 2^{33.6} = 2^{35.6}$ table look-ups, or equivalently 2^{29} five-round encryptions of AES (using the approximation[4] 1 table look-up ≈ 1 round of AES).

[3] As example, let $J \subseteq \{0, 1, 2, 3\}$ with $d = |J|$. Given a pair (c^1, c^2), this operation can be reduced to check that $\tilde{c}_{k,j-k} = 0$ for each $k = 0, ..., 3$ and $j \in J$, where $\tilde{c} \equiv c^1 \oplus c^2$.

[4] We highlight that even if this approximation is not formally correct - the size of the table of an S-Box look-up is lower than the size of the table used for our proposed distinguisher, it allows to give a comparison between our proposed distinguisher and the others currently present in literature. At the same time, we note that the same approximation is largely used in literature.

```
Data: $2^{32}$ (plaintext, ciphertext) pairs $(p^i, c^i)$ for $i = 0, ..., 2^{32} - 1$ in a single
      diagonal set.
Result: 1 for an AES permutation, 0 otherwise (prob. $\geq 99\%$)
Let $(p^i, c^i)$ for $i = 0, ..., 2^{32} - 1$ the (plaintext, ciphertext) pairs;
for all $j \in \{0, 1, 2, 3\}$ do
    Let $A[0, ..., 2^{32} - 1]$ an array initialized to zero;
    for $i$ from 0 to $2^{32} - 1$ do
        $x \leftarrow 0$;
        for $k$ from 0 to 3 do
            $x \leftarrow x + c^i_{k,j-k} \cdot 256^k$; // $c^i_{k,j-k}$ denotes the byte of $c^i$ in row $k$
             and column $j - k \mod 4$
        end
        $A[x] \leftarrow A[x] + 1$;     // $A[x]$ denotes the value stored in the $x$-th
         address of the array $A$
    end
    $n \leftarrow 0$;
    for $i$ from 0 to $2^{32} - 1$ do
        $n \leftarrow n + A[i] \cdot (A[i] - 1)/2$;
    end
    if $(n \mod 8) \neq 0$ then
        return 0;
    end
end
return 1.
```

Algorithm 1. *Secret-Key Distinguisher for 5 rounds of AES* which exploits a property which is independent of the secret key - probability of success: $\geq 99\%$.

Another possible way to implement our distinguisher exploits a re-ordering algorithm. The goal is again to count the number of pairs of ciphertexts for which the bytes that lie in d fixed anti-diagonals are equal. In this case, the idea is to re-order the texts using a particular numerical order which depends - in a "certain way" - on these d anti-diagonals. Then, given a set of ordered texts, the idea is to work only on two consecutive elements in order to count the total number of pairs of ciphertexts with the required property. In other words, given ordered ciphertexts, one can work only on approximately 2^{32} different pairs (composed of consecutive elements with respect to the used order) instead of 2^{63} for each initial diagonal set. All the details of this method are given in Appendix D of [13]. This second implementation could be in some cases more efficient than the one proposed in details in this section when e.g. it is required to do further operations on the pairs of ciphertexts which are equal in the d fixed anti-diagonals.

3.4 Practical Verification

Using a C/C++ implementation[5], we have practically verified the distinguisher on a small scale variant of AES, as presented in [6]. While in "real" AES, each

[5] The source code is available at https://github.com/Krypto-iaik/AES_5round_SKdistinguisher.

word is composed of 8 bits, in this variant each word is composed of 4 bits. We refer to [6] for a complete description of this small-scale AES, and we limit ourselves to describe the results of our 5-round distinguisher in this case.

First of all, note that Theorem 1 holds exactly in the same way also for this small-scale variant of AES (the proof is independent by the fact that each word of AES is of 4 or 8 bits). Thus, our verification on the small-scale variant of AES is strong evidence for it to hold for the real AES.

We have verified the theorem for each possible value of d (i.e. for $1, 2, 3$). For the verification of the secret-key distinguisher, we have chosen plaintexts in the diagonal sets with a single active diagonal and $d = 1$. As result, we have verified that for 5-round AES the number of collisions is a multiple of 2, while this number does not have any particular property for a random permutation. Moreover, we have found that 2 initial diagonal sets are largely sufficient to distinguish a random permutation from an AES permutation also from a practical point of view, as predicted.

The differences between this small-scale AES and the real AES regard the total number of pairs of ciphertexts that satisfy the required property (equal bytes in 1 fixed diagonal), which in this case is well approximated by $2^{15} \cdot (2^{16} - 1) \cdot 2^{-16} \approx 2^{15}$ for each diagonal set, and the lower computational cost, which can be approximated by $2^{17.6} \cdot 4 \approx 2^{19.6}$ memory look-ups for each initial diagonal set, besides the memory costs. The *average* practical results of our experiments are in accordance with these numbers.

3.5 Generalizations of the Central Theorem

Until now we have considered only a particular case in order to set up our distinguisher. However, here we show that it is possible to generalize Theorem 1 as follows.

Firstly, note that the same distinguisher works also in the reverse direction (i.e. in the decryption mode) with the same complexity. Assume that the final MixColumns operation is omitted. In this case the strategy is to choose 2^{32} ciphertexts in the same inverse-diagonal set, i.e. a set of 2^{32} ciphertexts which are equal in all the bytes expect for the ones in the i-th anti-diagonal for a certain $i = 0, 1, 2, 3$ (similar definition of the diagonal set). As before, the idea is to count the number of different pairs of plaintexts for which the bytes that lie in d diagonals are equal, for d fixed diagonals with $1 \leq d \leq 3$. This number has the same properties given in Theorem 1, while for a random permutation it can take any possible value.

Theorem 2. *Assume the final MixColumns operation is omitted. Given 2^{32} ciphertexts in the same inverse-diagonal set - that is, a set of texts with equal bytes expect the ones in the i-th anti-diagonal for a certain $i \in \{0, 1, 2, 3\}$, consider the corresponding plaintexts 5 rounds before, that is (p^i, c^i) for $i = 0, ..., 2^{32} - 1$ where $p^i = R^{-5}(c^i)$ The number n of different pairs of plaintexts (p^i, p^j) for $i \neq j$ for which the bytes of the difference $p^i \oplus p^j$ that lie in d diagonals are equal to zero (where $1 \leq d \leq 3$ and the diagonals are fixed in advance) is a multiple of 8, that is $\exists\, n' \in \mathbb{N}$ such that $n = 8 \cdot n'$.*

A complete proof of this Theorem can be found in Appendix A of the full version of the paper [13].

Secondly, Theorem 1 can be generalized for the cases of diagonal sets in which more than a single diagonal is active. As an example, diagonal sets with 2 or 3 active diagonals can be

$$\begin{bmatrix} A & A & C & C \\ C & A & A & C \\ C & C & A & A \\ A & C & C & A \end{bmatrix} \qquad \text{or} \qquad \begin{bmatrix} A & A & A & C \\ C & A & A & A \\ A & C & A & A \\ A & A & C & A \end{bmatrix}.$$

It is possible to prove that the result given in Theorem 1 is completely independent of the number of active diagonals. In other words, independently of the number of active diagonals of the initial diagonal set of the plaintexts, then the number of pairs of ciphertexts for which the bytes that lie in d anti-diagonals are equal (for d fixed anti-diagonals with $1 \leq d \leq 3$) is a multiple of 8. A formal statement is the following:

Theorem 3. *Given $2^{32 \cdot D}$ plaintexts in the same diagonal set with $1 \leq D \leq 3$ active diagonals defined as before, consider the corresponding ciphertexts after 5 rounds, that is (p^i, c^i) for $i = 0, ..., 2^{32} - 1$ where $c^i = R^5(p^i)$ The number n of different pairs of ciphertexts (c^i, c^j) for $i \neq j$ for which the bytes of the difference $c^i \oplus c^j$ that lie in d anti-diagonals are equal to zero (where $1 \leq d \leq 3$ and the anti-diagonals are fixed in advance) is a multiple of 8, that is $\exists\, n' \in \mathbb{N}$ such that $n = 8 \cdot n'$.*

The proof of this theorem is given in Appendix A - it is simply a generalization of the proof of Theorem 1 given in the next section.

4 A Detailed Proof of Theorem 1 - Lemma 1

In this section we give a detailed and formal proof of Theorem 1. As we have already said, since it is sufficient to prove Lemma 1 in order to prove the Theorem, we focus on this Lemma, which is recalled in the following. Moreover, we assume that for simplicity that the 2^{32} plaintexts are chosen in a diagonal set with the active bytes in the first diagonal (analogous for the other cases).

Lemma 1. *Given 2^{32} plaintexts in the same mixed set of the form (2)*

$$MC \cdot \begin{bmatrix} A & C & C & C \\ C & C & C & A \\ C & C & A & C \\ C & A & C & C \end{bmatrix},$$

consider the corresponding ciphertexts after 1 round, that is $(\hat{p}^i, \hat{c}^i)$ for $i = 0, ..., 2^{32} - 1$ where $\hat{c}^i = R(\hat{p}^i)$. The number n of different pairs of ciphertexts $(\hat{c}^i, \hat{c}^j)$ for which the bytes of the difference $c^i \oplus c^j$ that lie in d diagonals are equal to zero (where $1 \leq d \leq 3$ and the diagonals are fixed in advance) is a multiple of 8, that is $\exists\, n' \in \mathbb{N}$ s.t. $n = 8 \cdot n'$.

Proof. Consider two elements p^1 and p^2 in the set just defined. By definition, there exist $x, y, z, w \in \mathbb{F}_{2^8}$, $x', y', z', w' \in \mathbb{F}_{2^8}$ and a constant $a \in \mathbb{F}_{2^8}^{4\times 4}$ such that:

$$p^1 = a \oplus \begin{bmatrix} 2\cdot x & y & z & 3\cdot w \\ x & y & 3\cdot z & 2\cdot w \\ x & 3\cdot y & 2\cdot z & w \\ 3\cdot x & 2\cdot y & z & w \end{bmatrix}, \qquad p^2 = a \oplus \begin{bmatrix} 2\cdot x' & y' & z' & 3\cdot w' \\ x' & y' & 3\cdot z' & 2\cdot w' \\ x' & 3\cdot y' & 2\cdot z' & w' \\ 3\cdot x' & 2\cdot y' & z' & w' \end{bmatrix}$$

where $2 \equiv 0 \times 02$ and $3 \equiv 0 \times 03$. For the following, we say that p^1 is "generated" by the variables $\langle x, y, z, w\rangle$ and that p^2 is "generated" by the variables $\langle x', y', z', w'\rangle$.

First Case. First, we consider the case in which three variables are equal. W.l.o.g. we assume for example that $y = y'$, $z = z'$, $w = w'$ and $x \neq x'$ (the other cases are analogous). As we are going to show, in this case it is not possible that after one round the bytes of one diagonal (e.g. the j-th diagonal for $j \in \{0, 1, 2, 3\}$) of the two texts are equal). In other words, it is not possible that $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for each $i = 0, ..., 3$ (i.e. the four bytes of the j-th diagonal of $R(p^1) \oplus R(p^2)$ are equal to zero), where the indexes are taken modulo 4. As we are going to show, this is due to the given hypothesis of this case and to the fact that the branch number of the MixColumns operation is equal to five.

By simple computation, the first column (analogues for the other ones) of $SR\circ$ S-Box$(p^1) \oplus SR\circ$ S-Box(p^2) - denoted by $(SR\circ$ S-Box$(p^1) \oplus SR\circ$ S-Box$(p^2))_{\cdot,0}$ - is equal to:

$$(SR \circ \text{ S-Box}(p^1) \oplus SR \circ \text{ S-Box}(p^2))_{\cdot,0} = \begin{bmatrix} \text{S-Box}(2\cdot x \oplus a_{0,0}) \oplus \text{S-Box}(2\cdot x' \oplus a_{0,0}) \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

After the MixColumns operation (note $R(p^1) \oplus R(p^2) = MC(SR\circ$ S-Box$(p^1) \oplus SR\circ$ S-Box$(p^2)) = MC\circ SR\circ$ S-Box$(p^1) \oplus MC\circ SR\circ$ S-Box$(p^2))$, since only one input byte[6] is different from zero, it follows that at least four output bytes must be different from zero, that is all the output bytes are different from zero. This simply implies that it is not possible that the bytes of one or more diagonals of $R(p^1) \oplus R(p^2)$ are equal to zero. As a consequence, this case does not contribute to the number n.

Second Case. Secondly, we consider the case in which two variables are equal, that is w.l.o.g. we assume for example that $z = z'$ and $w = w'$, while $x \neq x'$ and $y \neq y'$ (the other cases are analogous).

Assume there exist two elements p^1 (generated by $\langle x, y\rangle$) and p^2 (generated by $\langle x', y'\rangle$) defined as before such that they have zero-difference in the j-th

[6] Note that S-Box$(2\cdot x \oplus a_{0,0}) \oplus$ S-Box$(2\cdot x' \oplus a_{0,0}) = 0$ if and only if $x = x'$, which can never happen for hypothesis.

diagonal after one round. In other words, let $j \in \{0,1,2,3\}$ and assume that there exist x, y and x', y' such that the generated elements p^1 and p^2 satisfy $(R(p^1) \oplus R(p^2))_{i,i+j} = 0$ for each $i = 0,1,2,3$, where the indexes are taken modulo 4.

This implies that *other* two elements $\hat{p}^1$ (generated by $\langle x, y' \rangle$) and $\hat{p}^2$ (generated by $\langle x, y' \rangle$), that is

$$\hat{p}^1 = a \oplus \begin{bmatrix} 2 \cdot x' & y & 0 & 0 \\ x' & y & 0 & 0 \\ x' & 3 \cdot y & 0 & 0 \\ 3 \cdot x' & 2 \cdot y & 0 & 0 \end{bmatrix} \qquad \text{and} \qquad \hat{p}^2 = a \oplus \begin{bmatrix} 2 \cdot x & y' & 0 & 0 \\ x & y' & 0 & 0 \\ x & 3 \cdot y' & 0 & 0 \\ 3 \cdot x & 2 \cdot y' & 0 & 0 \end{bmatrix},$$

satisfy the condition $(R(\hat{p}^1) \oplus R(\hat{p}^2))_{i,i+j} = 0$ for each $i = 0,1,2,3$ and for a certain j after one round. To prove this fact, it is sufficient to compute $R(p^1) \oplus R(p^2)$ and $R(\hat{p}^1) \oplus R(\hat{p}^2)$, and to prove that they are equal, i.e.

$$R(p^1) \oplus R(p^2) = R(\hat{p}^1) \oplus R(\hat{p}^2).$$

Since $(R(p^1) \oplus R(p^2))_{i,i+j} = 0$ for each $i = 0,1,2,3$, it also follows that $(R(\hat{p}^1) \oplus R(\hat{p}^2))_{i,i+j} = 0$ for each i. In particular, by simple computation the first column of $R(p^1) \oplus R(p^2)$ is given by:

$$\begin{aligned}
(R(p^1) \oplus R(p^2))_{0,0} &= 2 \cdot (\text{S-Box}(2 \cdot x \oplus a_{0,0}) \oplus \text{S-Box}(2 \cdot x' \oplus a_{0,0})) \oplus \\
&\oplus 3 \cdot (\text{S-Box}(y \oplus a_{1,1}) \oplus \text{S-Box}(y' \oplus a_{1,1})), \\
(R(p^1) \oplus R(p^2))_{1,0} &= \text{S-Box}(2 \cdot x \oplus a_{0,0}) \oplus \text{S-Box}(2 \cdot x' \oplus a_{0,0}) \oplus \\
&\oplus 2 \cdot (\text{S-Box}(y \oplus a_{1,1}) \oplus \text{S-Box}(y' \oplus a_{1,1})), \\
(R(p^1) \oplus R(p^2))_{2,0} &= \text{S-Box}(2 \cdot x \oplus a_{0,0}) \oplus \text{S-Box}(2 \cdot x' \oplus a_{0,0}) \oplus \\
&\oplus \text{S-Box}(y \oplus a_{1,1}) \oplus \text{S-Box}(y' \oplus a_{1,1}), \\
(R(p^1) \oplus R(p^2))_{3,0} &= 3 \cdot (\text{S-Box}(2 \cdot x \oplus a_{0,0}) \oplus \text{S-Box}(2 \cdot x' \oplus a_{0,0})) \oplus \\
&\oplus \text{S-Box}(y \oplus a_{1,1}) \oplus \text{S-Box}(y' \oplus a_{1,1}).
\end{aligned}$$

Due to the definition of $\hat{p}^1$ and $\hat{p}^2$, it follows immediately that $(R(p^1) \oplus R(p^2))_{\cdot,0} = (R(\hat{p}^1) \oplus R(\hat{p}^2))_{\cdot,0}$. The same happens for the other columns. Note that the two elements $\hat{p}^1$ and $\hat{p}^2$ exist for sure since we are working with all the 2^{32} plaintexts in the same mixed set (2). This implies that the number of collisions must be even, that is a multiple of 2.

Question: given p^1 and p^2 as before, is it possible that x, y, x', y' exist such that $(R(p^1) \oplus R(p^2))_{i,i+j} = 0$ for each $i = 0,1,2,3$? Yes, again because the branch number of the MixColumns operation is five. Indeed, compute $SR \circ$ S-Box$(p^1) \oplus SR \circ$ S-Box(p^2) and analyze the first column (the others are analogous):

$$(SR \circ \text{ S-Box}(p^1) \oplus SR \circ \text{ S-Box}(p^2))_{\cdot,0} = \begin{bmatrix} \text{S-Box}(2 \cdot x \oplus a_{0,0}) \oplus \text{S-Box}(2 \cdot x' \oplus a_{0,0}) \\ \text{S-Box}(y \oplus a_{1,1}) \oplus \text{S-Box}(y' \oplus a_{1,1}) \\ 0 \\ 0 \end{bmatrix}.$$

After the MixColumns operation (note $R(p^1) \oplus R(p^2) = MC(SR \circ \text{ S-Box}(p^1) \oplus SR \circ \text{ S-Box}(p^2)))$, since two input bytes[7] are different from zero, it follows that at least three output bytes must be different from zero, or at most one output byte could be equal to zero (similar for the other columns). Moreover, this also implies that it is not possible that two or more output bytes in the same column are equal to zero.

Moreover, observe that $(R(p^1) \oplus R(p^2))_{i,i+j} = 0$ for each i if and only if four bytes (one per column) of $R(p^1) \oplus R(p^2)$ are equal to zero. Since there are four "free" variables (i.e. x, y, x', y') and a system of four equations, such a system can have a non-negligible solution.

Finally, since the previous result is independent of the values of $z = z'$ and $w = w'$, it follows that the number of collisions for this case must be a multiple of 2^{17}. Indeed, assume that for certain $\hat{z}$ and $\hat{w}$ there exist x, y, x', y' such that the two elements p^1 and p^2 generated respectively by $\langle x, y\rangle$ by $\langle x', y'\rangle$ satisfy the condition that $R(p^1) \oplus R(p^2)$ has zero-difference in the j-th diagonal. By simple computation, the difference $R(p^1) \oplus R(p^2)$ doesn't depend on $z = z'$ and on $w = w'$, that is for each byte of $(R(p^1) \oplus R(p^2))_{k,l}$ for $k, l = 0, 1, 2, 3$ there exist constant A_i, B_i, C_i for $i = 0, 1, 2, 3$ - that depend only on the coefficients of the MixColumns matrix or/and of the secret-key - such that

$$\begin{aligned}(R(p^1) \oplus R(p^2))_{k,l} =& A_0 \cdot (\text{S-Box}(B_0 \cdot x \oplus C_0) \oplus \text{S-Box}(B_0 \cdot x' \oplus C_0)) \oplus \\ & \oplus A_1 \cdot (\text{S-Box}(B_1 \cdot y \oplus C_1) \oplus \text{S-Box}(B_1 \cdot y' \oplus C_1)) \oplus \\ & \oplus A_2 \cdot (\text{S-Box}(B_2 \cdot z \oplus C_2) \oplus \text{S-Box}(B_2 \cdot z' \oplus C_2)) \oplus \\ & \oplus A_3 \cdot (\text{S-Box}(B_3 \cdot w \oplus C_3) \oplus \text{S-Box}(B_3 \cdot w' \oplus C_3)) = \\ =& A_0 \cdot (\text{S-Box}(B_0 \cdot x \oplus C_0) \oplus \text{S-Box}(B_0 \cdot x' \oplus C_0)) \oplus \\ & \oplus A_1 \cdot (\text{S-Box}(B_1 \cdot y \oplus C_1) \oplus \text{S-Box}(B_1 \cdot y' \oplus C_1)).\end{aligned}$$

It follows that - under the previous hypothesis - each pair of elements p^1 and p^2 respectively generated by (1) $\langle x, y, z, w\rangle$ and by $\langle x', y', z, w\rangle$ or (2) $\langle x, y', z, w\rangle$ and by $\langle x', y, z, w\rangle$ *for each possible value of* z *and* w satisfy the condition that $R(p^1) \oplus R(p^2)$ has zero-difference in the j-th diagonal. Thus, the number of collisions for this case must be a multiple of $2 \cdot (2^8)^2 = 2^{17}$. As before, the existence of all these elements is guaranteed by the fact that we are working with all the 2^{32} plaintexts in the same mixed set (2).

Third Case. Thirdly, we consider the case in which only one variable is equal, that is w.l.o.g. we assume for example $w = w'$, while $x \neq x'$, $y \neq y'$ and $z \neq z'$ (the other cases are analogous).

Assume there exist two elements p^1 (generated by $\langle x, y, z\rangle$) and p^2 (generated by $\langle x', y', z'\rangle$) defined as before and $J \subseteq \{0, 1, 2, 3\}$ with $1 \leq d = |J| \leq 2$ such that the bytes of the two texts are equal after one round in the j-th diagonals

[7] Note that $\text{S-Box}(2 \cdot x \oplus a_{0,0}) \oplus \text{S-Box}(2 \cdot x' \oplus a_{0,0}) = 0$ if and only if $x = x'$, which can never happen for hypothesis. In the same way, $\text{S-Box}(y \oplus a_{1,1}) \oplus \text{S-Box}(y' \oplus a_{1,1}) = 0$ if and only if $y = y'$, which can never happen for hypothesis.

for $j \in J$. In other words, assume there exist x, y, z and x', y', z' such that the generated elements p^1 and p^2 satisfy $(R(p^1) \oplus R(p^2))_{i,j+i}$ for $j \in J$ with $1 \leq |J| \leq 2$. Similar to before, it follows that also the following three pairs of plaintexts generated by:

- $\langle x', y, z \rangle$ and $\langle x, y', z' \rangle$
- $\langle x, y', z \rangle$ and $\langle x', y, z' \rangle$
- $\langle x, y, z' \rangle$ and $\langle x', y', z \rangle$

have the same property (that is, the bytes in the j-th diagonals for $j \in J$ are equal after one round), for a total of four different pairs. As before, in order to prove this fact it is sufficient to show that $R(p^1) \oplus R(p^2) = R(\hat{p}^1) \oplus R(\hat{p}^2)$, where $\hat{p}^1$ and $\hat{p}^2$ are generated by the previous combinations of variables. Note that the two elements $\hat{p}^1$ and $\hat{p}^2$ exist for sure since we are working with all the 2^{32} plaintexts in the same mixed set (2). This implies that the number of collisions must be a multiple of 4.

Finally, we have only to prove that such x, y, z and x', y', z' can exist. As before, we compute $SR \circ$ S-Box$(p^1) \oplus SR \circ$ S-Box(p^2) and analyze the first column (the others are analogous):

$$(SR \circ \text{ S-Box}(p^1) \oplus SR \circ \text{ S-Box}(p^2))_{\cdot,0} = \begin{bmatrix} \text{S-Box}(2 \cdot x \oplus a_{0,0}) \oplus \text{S-Box}(2 \cdot x' \oplus a_{0,0}) \\ \text{S-Box}(y \oplus a_{1,1}) \oplus \text{S-Box}(y' \oplus a_{1,1}) \\ \text{S-Box}(2 \cdot z \oplus a_{2,2}) \oplus \text{S-Box}(2 \cdot z' \oplus a_{2,2}) \\ 0 \end{bmatrix}.$$

After the MixColumns operation, since three input bytes[8] are different from zero, it follows that at least two output bytes must be different from zero, or at most two output bytes could be equal to zero. This implies that the event $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all $i = 0, 1, 2, 3$ and $j \in J$ with $1 \leq |J| \leq 2$ is possible. Moreover, this also implies that it is not possible that three output bytes (of the same column) are equal to zero, or in other words that $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all i and all $j \in J$ with $d = |J| = 3$ is not possible Also in this case, variables x, y, z and x', y', z' can exist since the number of equations is less or equal than the number of variables.

Finally, since the previous result is independent of the values of $w = w'$, it follows that the number of collisions for this case must be a multiple of $4 \cdot 2^8 = 2^{10}$. As before, assume that for a certain $\hat{w}$ there exist x, y, z, x', y', z' such that the two elements p^1 and p^2 generated respectively by $\langle x, y, z \rangle$ and by $\langle x', y', z' \rangle$ satisfy the condition that $R(p^1) \oplus R(p^2)$ has zero-difference in the j-th diagonals for $j \in J$. Also in this case, the idea is to show that the difference $R(p^1) \oplus R(p^2)$ doesn't depend on $w = w'$, that is for each byte of $(R(p^1) \oplus R(p^2))_{i,j}$ there exist constant A_i, B_i, C_i for $i = 0, 1, 2$ - that depend only on the coefficients of the MixColumns matrix or/and of the secret-key - such that

[8] Note that S-Box$(2 \cdot x \oplus a_{0,0}) \oplus$S-Box$(2 \cdot x' \oplus a_{0,0})$ = S-Box$(y \oplus a_{1,1}) \oplus$S-Box$(y' \oplus a_{1,1})$ = S-Box$(2 \cdot z \oplus a_{2,2}) \oplus$ S-Box$(2 \cdot z' \oplus a_{2,2}) = 0$ if and only if $x = x'$, $y = y'$ and $z = z'$, which can never happen for hypothesis.

$$\begin{aligned}(R(p^1) \oplus R(p^2))_{i,j} = & A_0 \cdot (\text{S-Box}(B_0 \cdot x \oplus C_0) \oplus \text{S-Box}(B_0 \cdot x' \oplus C_0)) \oplus \\ & \oplus A_1 \cdot (\text{S-Box}(B_1 \cdot y \oplus C_1) \oplus \text{S-Box}(B_1 \cdot y' \oplus C_1) \oplus \\ & \oplus A_2 \cdot (\text{S-Box}(B_2 \cdot z \oplus C_2) \oplus \text{S-Box}(B_2 \cdot z' \oplus C_2).\end{aligned}$$

It follows that - under the previous hypothesis - each pair of elements p^1 and p^2 respectively generated by one of the four different combinations of the variables $\langle x, y, z, w\rangle$ and $\langle x', y', z', w\rangle$ *for each possible value of* w satisfy the condition that $R(p^1) \oplus R(p^2)$ has zero-difference in the j-th diagonals for $j \in J$. As before, the existence of all these elements is guaranteed by the fact that we are working with all the 2^{32} plaintexts in the same mixed set (2).

Fourth Case. Fourthly, we consider the case in which all the variables are different, that is w.l.o.g. we assume that $x \neq x'$, $y \neq y'$, $z \neq z'$ and $w \neq w'$.

Assume there exist two elements p^1 (generated by $\langle x, y, z, w\rangle$) and p^2 (generated by $\langle x', y', z', w'\rangle$) defined as before and $J \subseteq \{0, 1, 2, 3\}$ with $1 \leq d = |J| \leq 3$ such that the bytes of the two texts are equal after one round in the j-th diagonals for $j \in J$. In other words, assume there exist x, y, z, w and x', y', z', w' such that the generated elements p^1 and p^2 satisfy $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all $i = 0, 1, 2, 3$ and for all $j \in J$ with $1 \leq d = |J| \leq 3$. Similar to before, it follows that also the following seven pairs of plaintexts generated by:

- $\langle x', y, z, w\rangle$ and $\langle x, y', z', w'\rangle$
- $\langle x, y', z, w\rangle$ and $\langle x', y, z', w'\rangle$
- $\langle x, y, z', w\rangle$ and $\langle x', y', z, w'\rangle$
- $\langle x, y, z, w'\rangle$ and $\langle x', y', z', w\rangle$
- $\langle x', y', z, w\rangle$ and $\langle x, y, z', w'\rangle$
- $\langle x', y, z', w\rangle$ and $\langle x, y', z, w'\rangle$
- $\langle x', y, z, w'\rangle$ and $\langle x, y', z', w\rangle$

have the same property (thta is, the bytes in the j-th diagonals for $j \in J$ are equal after one round), for a total of eight different pairs. As before, in order to prove this fact it is sufficient to show that $R(p^1) \oplus R(p^2) = R(\hat{p}^1) \oplus R(\hat{p}^2)$. Moreover, as before note that the two elements $\hat{p}^1$ and $\hat{p}^2$ exist for sure since we are working with all the 2^{32} plaintexts in the same mixed set (2). This implies that the number of collisions must be a multiple of 8.

Finally, we have only to prove that such x, y, z, w and x', y', z', w' can exist. As before, we compute $SR \circ$ S-Box$(p^1) \oplus SR \circ$ S-Box(p^2) and analyze the first column (the others are analogous):

$$(SR \circ \text{ S-Box}(p^1) \oplus SR \circ \text{ S-Box}(p^2))_{\cdot,0} = \begin{bmatrix} \text{S-Box}(2 \cdot x \oplus a_{0,0}) \oplus \text{S-Box}(2 \cdot x' \oplus a_{0,0}) \\ \text{S-Box}(y \oplus a_{1,1}) \oplus \text{S-Box}(y' \oplus a_{1,1}) \\ \text{S-Box}(2 \cdot z \oplus a_{2,2}) \oplus \text{S-Box}(2 \cdot z' \oplus a_{2,2}) \\ \text{S-Box}(w \oplus a_{3,3}) \oplus \text{S-Box}(w' \oplus a_{3,3}) \end{bmatrix}.$$

After the MixColumns operation, since four input bytes[9] are different from zero, it follows that at least one output byte must be different from zero, or at most three output bytes could be equal to zero. This implies that the event $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all $i = 0, 1, 2, 3$ and for all $j \in J$ with $1 \leq |J| \leq 3$ is possible. Also in this case, variables x, y, z, w and x', y', z', w' can exist since the number of equations is less or equal than the number of variables.

Conclusion. We summarize the previous results and we prove the lemma. Given a set (2), we analyze the number of pairs of texts for which the bytes of d diagonals are equal after one round.

If $d = 3$, it is possible to have a collision only in the case in which all the variables that generate the two texts are different, that is $x \neq x'$, $y \neq y'$, and so on. In this case, the number of collisions n must be a multiple of 8, that is there exists $n' \in \mathbb{N}$ such that $n = 8 \cdot n'$.

If $d = 2$, it is possible to have a collision only if at least three variables that generate the two texts are different (i.e. at most one variable can be equal). If all the variables are different, the number of collisions is a multiple of 8, while if one is equal then the number of collisions is a multiple of $1024 \equiv 2^{10}$. In other words, there exist $n', n'_2 \in \mathbb{N}$ such that the total number of collisions n is equal to $n = 8 \cdot n' + 1024 \cdot n'_2 = 8 \cdot (n' + 128 \cdot n'_2)$, i.e. it is a multiple of 8.

If $d = 3$, it is possible to have a collision only if at least two variables that generate the two texts are different (i.e. at most two variables can be equal). If all the variables are different, the number of collisions is a multiple of 8, if one is equal then the number of collisions is a multiple of $1024 \equiv 2^{10}$, while if two are equal then the number of collisions is a multiple of $131072 \equiv 2^{17}$. In other words, there exist $n', n'_2, n'_3 \in \mathbb{N}$ such that the total number of collisions n is equal to $n = 8 \cdot n' + 2^{10} \cdot n'_2 + 2^{17} \cdot n'_3 = 8 \cdot (n' + 2^7 \cdot n'_2 + 2^{14} \cdot n'_3)$, i.e. it is a multiple of 8.

This proves the lemma. □

For completeness, we briefly recall why the proof of Lemma 1 implies Theorem 1. As we have already seen, if two plaintexts are in the same diagonal set, then after two rounds some particular linear relationships (given in (1)) hold among the bytes of the two texts that lie in the same column with probability 1. In the same way, if two ciphertexts have equal bytes in d anti-diagonals, then two rounds before - assuming the final MixColumns operation is omitted - the the two texts have equal bytes in d diagonals (due to the 2-round differential trail described in Sect. 2.1). Thus, it is sufficient to prove that given a mixed set of the form (2), the number of pairs of texts for which the bytes of d diagonals are equal after one round is a multiple of 8, which is the statement of Lemma 1. This finally proves the theorem.

[9] Note that $\text{S-Box}(2 \cdot x \oplus a_{0,0}) \oplus \text{S-Box}(2 \cdot x' \oplus a_{0,0}) = \text{S-Box}(y \oplus a_{1,1}) \oplus \text{S-Box}(y' \oplus a_{1,1}) = \text{S-Box}(2 \cdot z \oplus a_{2,2}) \oplus \text{S-Box}(2 \cdot z' \oplus a_{2,2}) = \text{S-Box}(w \oplus a_{3,3}) \oplus \text{S-Box}(w' \oplus a_{3,3}) = 0$ if and only if $x = x'$, $y = y'$, $z = z'$ and $w = w'$, which can never happen for hypothesis.

5 Conclusion, Applications and Open Problems

In this paper, we have presented a new non-random property for 5 rounds of AES. Additionally, we showed how to set up an efficient 5-round secret-key distinguisher for AES which exploits this property, which is independent of the secret key, improving the very recent results [21] and providing answers to the questions posed in [21]. This distinguisher is structural in the sense that it is independent of the details of the MixColumns matrix (with the exception that the branch number must be five) and also independent of the SubBytes operation. As such it will be straightforward to apply to many other AES-like constructions. Starting from our results, a range of new questions arise for future investigations:

Application to Schemes that directly use round-reduced AES. Round-reduced AES is a popular construction to build different schemes. For example, in the on-going "Competition for Authenticated Encryption: Security, Applicability, and Robustness" (CAESAR) [1], which is currently at its third round, several candidates are designed based on an AES-like SPN structure. Focusing only on the third-round candidates[10], among many others, AEGIS [15] uses four AES round-functions in the state update functions while ELmD [20] recommends to use round-reduced AES including 5-round AES to partially encrypt the data. Although the security of these candidates does not completely depend on the underlying primitives, we believe that a better understanding of the security of round-reduced AES can help get insights to both the design and cryptanalysis of authenticated encryption algorithms.

Further Extensions. Is it possible to set up a secret-key distinguisher for 6-round of AES which exploits a property which is independent of the secret key? Is it possible to set up efficient key recovery attacks for 6- or more rounds of AES that exploits this new 5-round secret-key distinguisher proposed in this paper or a modified version of it?

Permutation and Known-Key Distinguishers. The new 5-round property (or its approach to derive it) might find applications to permutation distinguishers or known-key distinguishers. Permutation distinguisher are usually set up by combining two secret-key distinguishers in an inside-out fashion. It is not immediately clear how the 5-round secret-key distinguisher presented in this paper used in an inside-out approach would be able to maintain the property in both directions simultaneously, but it seems interesting to investigate this direction also.

Acknowledgments. The work in this paper has been partially supported by the Austrian Science Fund (project P26494-N15).

[10] Among previous-round candidates, it is also possible to include PRIMATEs [11] which design is based on an AES-like SPN structure, while 4-round AES is adopted by Marble [16] and used to build the AESQ permutation in PAEQ [2].

A Generalization of Theorem 1

In Theorem 1 given in Sect. 3, we only considered the case of chosen plaintexts in the same diagonal set with a single active diagonal - i.e. $D = 1$. A natural question arises: is it possible to generalize the theorem also for $D = 2$ or/and $D = 3$, that is for chosen plaintexts in the same diagonal set with two or three active diagonals? The answer is yes, and it is given in Theorem 3 recalled in the following. In particular, we prove in this section that the result obtained in Theorem 1 is independent of the number of initial active diagonals D, or, in other words, the property of n to be a multiple of 8 is independent of D.

Theorem 1. *Given $2^{32\cdot D}$ plaintexts in the same diagonal set with $1 \leq D \leq 3$ active diagonals defined as before, consider the corresponding ciphertexts after 5 rounds, that is (p^i, c^i) for $i = 0, ..., 2^{32} - 1$ where $c^i = R^5(p^i)$ The number n of different pairs of ciphertexts (c^i, c^j) for $i \neq j$ for which the bytes of the difference $c^i \oplus c^j$ that lie in d anti-diagonals are equal to zero (where $1 \leq d \leq 3$ and the anti-diagonals are fixed in advance) is a multiple of 8, that is $\exists\, n' \in \mathbb{N}$ such that $n = 8 \cdot n'$.*

Since the proof for the case $D = 1$ is given in Sect. 4, we focus on the cases $D = 2$ and $D = 3$. Also for these cases, the idea is to analyze the middle round and to study each possible case, as done in Sect. 4. Thus, given pair of texts of the form

$$MC \cdot \begin{bmatrix} A & A & C & C \\ A & C & C & A \\ C & C & A & A \\ C & A & A & C \end{bmatrix} \qquad \text{or} \qquad MC \cdot \begin{bmatrix} A & A & A & C \\ A & A & C & A \\ A & C & A & A \\ C & A & A & A \end{bmatrix}, \tag{4}$$

we analyze the property of the number of pairs of texts which are equal in d diagonals after one round.

Since the idea of the proof for $D = 2$ and $D = 3$ is analogous to that given for $D = 1$, we limit ourselves to do some considerations which justify the theorem. A complete proof can be easily obtained exploiting the following considerations and using the same strategy proposed in Sect. 4.

First Consideration. As first consideration, note that we are considering pairs of plaintexts/ciphertexts (p^1, c^1) and (p^2, c^2) such that the plaintexts are in the same diagonal set with at least 2 active diagonals. On the other hand, such a set can be seen as a collection of diagonal set with only 1 active diagonal. Since Theorem 1 holds for each one of these sets, it follows that if n is a multiple of 2^m then m must satisfy $m \leq 3$. This follows immediately by Theorem 1 and the corresponding proof of Sect. 4.

Thus, we have to prove that n is a multiple of 2^m and that $m = 3$ also for the cases $D = 2$ and $D = 3$.

A.1 Case $D = 2$

We start studying the case $D = 2$. As we show in details in the following, the same analysis can be simply modified and adapted for the case $D = 3$.

Consider two texts p^1 and p^2 in the same set (4) (the other cases are analogous). By definition, there exist $x_0, x_1, y_0, y_1, z_0, z_1, w_0, w_1 \in \mathbb{F}_{2^8}$, $x'_0, x'_1, y'_0, y'_1, z'_0, z'_1, w'_0, w'_1 \in \mathbb{F}_{2^8}$ and $a \in \mathbb{F}_{2^8}^{4\times 4}$ such that:

$$p^1 = a \oplus MC \cdot \begin{bmatrix} x_0 & y_0 & 0 & 0 \\ x_1 & 0 & 0 & w_0 \\ 0 & 0 & z_0 & w_1 \\ 0 & y_1 & z_1 & 0 \end{bmatrix}, \qquad p^2 = a \oplus MC \cdot \begin{bmatrix} x'_0 & y'_0 & 0 & 0 \\ x'_1 & 0 & 0 & w'_0 \\ 0 & 0 & z'_0 & w'_1 \\ 0 & y'_1 & z'_1 & 0 \end{bmatrix}.$$

For the following, let $2 \equiv 0 \times 02$ and $3 \equiv 0 \times 03$.

Following the same strategy of Sect. 4, the idea is to consider all the possible cases in which some or no-one variables of p^1 are equal to the ones of p^2. Note that the case $x_1 = x'_1$, $y_1 = y'_1$, $z_1 = z'_1$ and $w_1 = w'_1$ (i.e. two texts that belong into the same set (2)) has already been considered. In particular, by Theorem 1 it follows that in this case the number n is a multiple of 8.

First Case. W.l.o.g. we consider the case $y_1 = y'_1$, $w_i = w'_i$ and $z_i = z'_i$ for $i = 0, 1$, while $y_0 \neq y'_0$ and $x_i \neq x'_i$ for $i = 0, 1$ (the other cases are analogous).

Assume there exist x_0, x_1, y_0 and x'_0, x'_1, y'_0 such that the generated elements p^1 and p^2 satisfy the condition $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all $i = 0, 1, 2, 3$ and for a certain $j \in 0, 1, 2, 3$ - i.e. the bytes of one diagonal of the two texts are equal after one round. First of all, we show that such variables can exist. The condition $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all i and a certain $j \in \{0, 1, 2, 3\}$ implies that four bytes (one per column) of $R(p^1) \oplus R(p^2)$ must be equal to 0. Since there are six independent variables, a solution can exist (note that the number of variables is higher than the number of equations, so two variables are still "free"). Moreover, this is also due to the branch number of the MixColumns operation, which is five. Indeed, by simple computation the first column of $SR(\text{S-Box}(p^1) \oplus \text{ S-Box}(p^2))$ (analogous for the others) is given by:

$$\begin{aligned} SR(\text{S-Box}(p^1) \oplus \text{ S-Box}(p^2))_{0,0} &= \text{S-Box}(2 \cdot x_0 \oplus 3 \cdot x_1 \oplus a_{0,0} \oplus a_{1,0}) \\ &\quad \oplus \text{S-Box}(2 \cdot x'_0 \oplus 3 \cdot x'_1 \oplus a_{0,0} \oplus a_{1,0}), \\ SR(\text{S-Box}(p^1) \oplus \text{ S-Box}(p^2))_{1,0} &= \text{S-Box}(y_0 \oplus a_{1,1}) \oplus \text{S-Box}(y'_0 \oplus a_{1,1}), \\ SR(\text{S-Box}(p^1) \oplus \text{ S-Box}(p^2))_{2,0} &= SR(\text{S-Box}(p^1) \oplus \text{ S-Box}(p^2))_{3,0} = 0. \end{aligned}$$

Thus, if we compute $MC \circ SR(\text{S-Box}(p^1) \oplus \text{ S-Box}(p^2))$ (that is, $R(p^1) \oplus R(p^2)$), since at most two input bytes are different from zero, then it follows that at least three output bytes must be different from zero, or equivalently at most one output byte can be equal to zero. As a consequence, it is possible that $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all i and a certain $j \in \{0, 1, 2, 3\}$. Note that the same can not happen for $d \geq 2$ diagonals. We emphasize that with respect to the case $D = 1$, it is possible that one input byte of the MixColumns operation can be equal to zero. Indeed, it is possible that exist x_0 and x'_0 such that $SR(\text{S-Box}(p^1) \oplus \text{S-Box}(p^2))_{0,0}$ (analogous for the others columns).

As before, the idea is to consider the pairs of texts generated by all the possible combinations of these six variables, as for example $\langle x_0, x_1, y'_0\rangle$ and $\langle x'_0, x'_1, y_0\rangle$, $\langle x_0, x'_1, y_0\rangle$ and $\langle x'_0, x_1, y'_0\rangle$, $\langle x'_0, x_1, y_0\rangle$ and $\langle x_0, x'_1, y'_0\rangle$, $\langle x_1, x_0, y'_0\rangle$ and $\langle x'_0, x'_1, y_0\rangle$ (note that the elements generated by $\langle x_0, x_1, y'_0\rangle$ and by $\langle x_1, x_0, y'_0\rangle$ are different) and so on.

We analyze these cases. It is simple to observe that if p^1 generated by $\langle x_0, x_1, y_0\rangle$ and p^2 generated by $\langle x'_0, x'_1, y'_0\rangle$ satisfy the condition that $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all i and a certain $j \in \{0, 1, 2, 3\}$ - i.e. one diagonal of the two texts are equal after one round, then also the elements generated by $\langle x_0, x_1, y'_0\rangle$ and $\langle x'_0, x'_1, y_0\rangle$ have the same property. To prove this fact, it is sufficient to show that $R(p^1) \oplus R(p^2) = R(\hat{p}^1) \oplus R(\hat{p}^2)$. As an example, by simple computation, it is simple to observe that for the first column:

$$SR(\text{S-Box}(\hat{p}^1) \oplus \text{ S-Box}(\hat{p}^2))_{i,0} = SR(\text{S-Box}(p^1) \oplus \text{ S-Box}(p^2))_{i,0} \qquad \forall i,$$

which implies the statement.

Consider now the elements $\hat{p}^1$ generated by $\langle x_0, x'_1, y_0\rangle$ and $\hat{p}^2$ generated by $\langle x'_0, x_1, y'_0\rangle$ (similar for the elements generated by $\langle x'_0, x_1, y_0\rangle$ and $\langle x_0, x'_1, y'_0\rangle$). By simple computation, the first column of $SR(\text{S-Box}(\hat{p}^1) \oplus \text{ S-Box}(\hat{p}^2))$ (analogous for the others) is given by:

$$\begin{aligned} SR(\text{S-Box}(\hat{p}^1) \oplus \text{ S-Box}(\hat{p}^2))_{0,0} &= \text{S-Box}(2 \cdot x_0 \oplus 3 \cdot x'_1 \oplus a_{0,0} \oplus a_{1,0}) \\ &\quad \oplus \text{S-Box}(2 \cdot x'_0 \oplus 3 \cdot x_1 \oplus a_{0,0} \oplus a_{1,0}) \end{aligned}$$

and for $i = 1, 2, 3$

$$SR(\text{S-Box}(\hat{p}^1) \oplus \text{ S-Box}(\hat{p}^2))_{i,0} = SR(\text{S-Box}(p^1) \oplus \text{ S-Box}(p^2))_{i,0}.$$

Since the S-Box is a non-linear operation, three different cases can happen:

1. $SR(\text{S-Box}(\hat{p}^1) \oplus \text{ S-Box}(\hat{p}^2))_{0,0} = 0$;
2. $SR(\text{S-Box}(\hat{p}^1) \oplus \text{ S-Box}(\hat{p}^2))_{0,0} \neq 0$ and the elements $\hat{p}^1$ and $\hat{p}^2$ satisfy the condition $(R(\hat{p}^1) \oplus R(\hat{p}^2))_{i,j+i} = 0$ for all i and a certain $j \in \{0, 1, 2, 3\}$;
3. $SR(\text{S-Box}(\hat{p}^1) \oplus \text{ S-Box}(\hat{p}^2))_{0,0} \neq 0$ and the elements $\hat{p}^1$ and $\hat{p}^2$ don't satisfy the condition $(R(\hat{p}^1) \oplus R(\hat{p}^2))_{i,j+i} = 0$ for all i and a certain $j \in \{0, 1, 2, 3\}$.

We analyze in details these three cases, starting from the first one. As first thing, note that this case can happen since the condition $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all i and a certain $j \in \{0, 1, 2, 3\}$ imposes a condition only on four out of six variables, that is two variables are still "free". If $SR(\text{S-Box}(\hat{p}^1) \oplus \text{ S-Box}(\hat{p}^2))_{0,0} = 0$, it follows that only one byte (i.e. the second one) of the first column of $SR(\text{S-Box}(\hat{p}^1) \oplus \text{ S-Box}(\hat{p}^2))$ is different from 0 (since $y_0 \neq y'_0$). Thus, since MixColumns operation has branch number 5, all the bytes of the first column of $R(\hat{p}^1) \oplus R(\hat{p}^2)$ must be different from zero, that is no diagonals of $R(\hat{p}^1)$ and $R(\hat{p}^2)$ can be equal. However, note that also in this case it is possible to deduce something. Indeed, by the previous consideration, it follows that the elements generated by $\langle x_0, x'_1, y'_0 \rangle$ and by $\langle x'_0, x_1, y_0 \rangle$ don't satisfy the condition $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all i and a certain $j \in \{0, 1, 2, 3\}$

Consider now the other two cases. Since the S-Box is a non-linear operation, it is not possible to guarantee that

$$SR(\text{S-Box}(\hat{p}^1) \oplus \text{ S-Box}(\hat{p}^2))_{0,0} = SR(\text{S-Box}(p^1) \oplus \text{ S-Box}(p^2))_{0,0}.$$

In other words, they can be equal (which implies that the condition $(R(\hat{p}^1) \oplus R(\hat{p}^2))_{i,j+i} = 0$ for all i and a certain $j \in \{0, 1, 2, 3\}$ - the same j of p^1 and p^2 - holds) or different. In this second case, one can not say anything about the fact that the elements $\hat{p}^1$ and $\hat{p}^2$ satisfy or not the condition $(R(\hat{p}^1) \oplus R(\hat{p}^2))_{i,j+i} = 0$ for all i and a certain $j \in \{0, 1, 2, 3\}$ (the same j of p^1 and p^2). However, suppose that $\hat{p}^1$ and $\hat{p}^2$ satisfy it after one round for the same j of p^1 and p^2 (which is independent by the previous condition). In the same way of before, note that also the elements generated by $\langle x_0, x'_1, y'_0 \rangle$ and $\hat{p}^2$ generated by $\langle x'_0, x_1, y_0 \rangle$ have the same property.

Thus, assume that p^1 generated by $\langle x_0, x_1, y_0 \rangle$ and p^2 generated by $\langle x'_0, x'_1, y'_0 \rangle$ satisfy or not the condition $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all i and a certain $j \in \{0, 1, 2, 3\}$ after one round. By previous considerations, it follows that also the $\hat{p}^1$ generated by $\langle x_0, x'_1, y_0 \rangle$ and $\hat{p}^2$ generated by $\langle x'_0, x_1, y'_0 \rangle$ have the same property. Thus, even if we can not do any claim for the other texts generated by a different combination of these six variables, it is possible to conclude that - *for fixed* $y_1 = y'_1$, $w_i = w'_i$ and $z_i = z'_i$ for $i = 0, 1$ - the number of collisions must be a multiple of 2 for this case.

Finally, since we are working with the entire set of the form (4) - that is, $y_1 = y'_1$, $w_i = w'_i$ and $z_i = z'_i$ for $i = 0, 1$ can take any possible value - and due to the same considerations of Sect. 4, it follows that the number of collisions must be a multiple of $2 \cdot (2^8)^5 = 2^{41}$ for this case.

Second Case. Similar considerations can be done for the case $w_i = w'_i$ and $z_i = z'_i$ for $i = 0, 1$, while $x_i \neq x'_i$ and $y_i \neq y'_i$ for $i = 0, 1$ (the other cases are analogous).

Assume there exist x_0, x_1, y_0, y_1 and x'_0, x'_1, y'_0, y'_1 such that the generated elements p^1 and p^2 satisfy the condition $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all i and a certain $j \in \{0, 1, 2, 3\}$. As before, note that this is possible since this implies that four bytes of $R(p_1) \oplus R(p^2)$ (one per column) must be equal to 0. Since there are eight independent variables, a solution can exist (note that the number of variables is higher than the number of equations, so four variables are still "free"). Due to the branch number of the MixColumns operation, even if four variables are still "free" it is not possible that the condition $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all i holds for two different j. Indeed, the first column of $SR(\text{S-Box}(p^1) \oplus$ S-Box$(p^2))$ (analogous for the others) is given by:

$$\begin{aligned} SR(\text{S-Box}(p^1) \oplus \text{ S-Box}(p^2))_{0,0} &= \text{S-Box}(2 \cdot x_0 \oplus 3 \cdot x_1 \oplus a_{0,0} \oplus a_{1,0}) \\ &\quad \oplus \text{S-Box}(2 \cdot x'_0 \oplus 3 \cdot x'_1 \oplus a_{0,0} \oplus a_{1,0}), \\ SR(\text{S-Box}(p^1) \oplus \text{ S-Box}(p^2))_{1,0} &= \text{S-Box}(y_0 \oplus y_1 \oplus a_{0,1} \oplus a_{3,0}) \end{aligned}$$

$$\oplus \text{ S-Box}(y'_0 \oplus y'_1 \oplus a_{0,1} \oplus a_{3,0}),$$
$$SR(\text{S-Box}(p^1) \oplus \text{ S-Box}(p^2))_{2,0} = SR(\text{S-Box}(p^1) \oplus \text{ S-Box}(p^2))_{3,0} = 0.$$

After the MixColumns operation $MC \circ SR(\text{S-Box}(p^1) \oplus \text{ S-Box}(p^2))$, since at most two input bytes are different from zero, then it follows that at least three output bytes must be different from zero.

Thus, given x_0, x_1, y_0, y_1 and x'_0, x'_1, y'_0, y'_1, the idea is to consider all the possible combinations as before. Also in this case, we can do a claim only on one of them. In particular, if two elements p^1 generated by $\langle x_0, x_1, y_0, y_1\rangle$ and p^2 generated by $\langle x'_0, x'_1, y'_0, y'_1\rangle$ satisfies the condition $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all i and a certain $j \in \{0, 1, 2, 3\}$, we can only claim that also the elements $\hat{p}^1$ generated by $\langle x'_0, x'_1, y_0, y_1\rangle$ and $\hat{p}^2$ generated by $\langle x_0, x_1, y'_0, y'_1\rangle$ have the same property. Considerations for the other combinations are similar to the previous case. Thus, we can claim that - *for fixed* $w_i = w'_i$ and $z_i = z'_i$ for $i = 0, 1$ - also for this case the number of collisions is a multiple of 2.

Finally, since we are working with the entire set of the form (4) - that is, $w_i = w'_i$ and $z_i = z'_i$ for $i = 0, 1$ can take any possible value - and due to the same considerations of Sect. 4, it follows that the number of collisions must be a multiple of $2 \cdot (2^8)^4 = 2^{33}$ for this case.

Second Consideration. What can we deduce by the previous two cases? Suppose to have two texts p^1 generated by $\langle x \equiv (x_0, x_1), y \equiv (y_0, y_1)\rangle$ and p^2 generated by $\langle x' \equiv (x'_0, x'_1), y' \equiv (y'_0, y'_1)\rangle$ that satisfy the condition $(R(\hat{p}^1) \oplus R(\hat{p}^2))_{i,j+i} = 0$ for all i and a certain $j \in \{0, 1, 2, 3\}$ and where $x, y \in \mathbb{F}_{2^8} \times \mathbb{F}_{2^8} \equiv \mathbb{F}^2_{2^8}$. We have seen that given these two elements, one can only claim that also the texts $\hat{p}^1$ generated by $\langle x' \equiv (x'_0, x'_1), y \equiv (y_0, y_1)\rangle$ and $\hat{p}^2$ generated by $\langle x \equiv (x_0, x_1), y' \equiv (y'_0, y'_1)\rangle$ have the same property, that is the condition $(R(\hat{p}^1) \oplus R(\hat{p}^2))_{i,j+i} = 0$ for all i and a certain $j \in \{0, 1, 2, 3\}$ for the same j of p^1 and p^2.

As a consequence, the idea for the case $D = 2$ is not to consider the variables that generate the texts and that are in the same column as independent. In other words, the idea is to work with variables in $\mathbb{F}^2_{2^8}$ and not in $\mathbb{F}_{2^8}$, i.e. to consider only all the possible combinations of $x \equiv (x_0, x_1), y \equiv (y_0, y_1)$ and $x' \equiv (x'_0, x'_1), y' \equiv (y'_0, y'_1)$, and not of x_0, x_1, y_0, y_1 and x'_0, x'_1, y'_0, y'_1. Using this strategy and working in the same way of Sect. 4, it is possible to analyze all the possible cases.

For example, consider the case in which $w_i = w'_i$ for $i = 0, 1$ and $x \equiv (x_0, x_1) \neq x' \equiv (x'_0, x'_1)$, $y \equiv (y_0, y_1) \neq y' \equiv (y'_0, y'_1)$ and $z \equiv (z_0, z_1) \neq z' \equiv (z'_0, z'_1)$. In the same way of before, it is only possible to prove that if there exist p^1 generated by $\langle x, y, z\rangle$ and p^2 generated by $\langle x', y', z'\rangle$ such that $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all i and certain $j \in J$ where $J \subseteq \{0, 1, 2, 3\}$ and $|J| = 2$ - i.e. two diagonals are equal, then a total of four elements generated by

- $\langle x, y, z\rangle$ and $\langle x', y', z'\rangle$
- $\langle x', y, z\rangle$ and $\langle x, y', z'\rangle$

- $\langle x, y', z\rangle$ and $\langle x', y, z'\rangle$
- $\langle x, y, z'\rangle$ and $\langle x', y', z\rangle$

have the same property. No claim can be made about other combinations of variables (as before, this is due to the fact that the S-Box is non-linear). It follows that - *for fixed* $w_i = w'_i$ for $i = 0, 1$- the number of collisions must be a multiple of 4 for this case. As before, since we are working with the entire set of the form (4) it follows that the number of collisions must be a multiple of $4 \cdot (2^8)^2 = 2^{18}$. Moreover, since the branch number of the MixColumns operation is five, note that it is not possible that $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all i and certain $j \in \{0, 1, 2, 3\}$ if $w_l = w'_l$ for $l = 0, 1$ (even if $(R(p^1) \oplus R(p^2))_{i,j+i} = 0$ for all i and certain $j \in J$ where $J \subseteq \{0, 1, 2, 3\}$ imposes only 8 conditions while the number of variables is 12, so 4 variables are still "free").

Similar considerations can be done for the case in which all the variables are different. As a consequence, the theorem is proved for the case $|I| = 2$.

A.2 Case $D = 3$

The case $D = 3$ is analogous to the case $D = 2$ and to the proof given in Sect. 4. For this reason, we limit ourselves to show how to adapt the proof of the case $D = 2$ for this case.

W.l.o.g consider two texts p^1 and p^2 in the same set (4) (the other cases are analogous). By definition, there exist $x_0, x_1, x_2, y_0, y_1, y_2, z_0, z_1, z_2, w_0, w_1, w_2 \in \mathbb{F}_{2^8}$, $x'_0, x'_1, x'_2, y'_0, y'_1, y'_2, z'_0, z'_1, z'_2, w'_0, w'_1, w'_2 \in \mathbb{F}_{2^8}$ and $a \in \mathbb{F}_{2^8}^{4\times 4}$ such that:

$$p^1 = a \oplus MC \cdot \begin{bmatrix} x_0 & y_0 & z_0 & 0 \\ x_1 & y_1 & 0 & w_0 \\ x_2 & 0 & z_1 & w_1 \\ 0 & y_2 & z_2 & w_2 \end{bmatrix}, \qquad p^2 = a \oplus MC \cdot \begin{bmatrix} x'_0 & y'_0 & z'_0 & 0 \\ x'_1 & y'_1 & 0 & w'_0 \\ x'_2 & 0 & z'_1 & w'_1 \\ 0 & y'_2 & z'_2 & w'_2 \end{bmatrix}.$$

Similarly to the case $D = 2$, the idea is to work with variables in $\mathbb{F}_{2^8}^3 \equiv \mathbb{F}_{2^8} \times \mathbb{F}_{2^8} \times \mathbb{F}_{2^8}$, e.g. $x \equiv (x_0, x_1, x_2), y \equiv (y_0, y_1, y_2)$ and so on. In other words, the idea is to consider the variables in the same column as not independent, that is to consider the possible combinations only of variables in $\mathbb{F}_{2^8}^3$ and not in $\mathbb{F}_{2^8}$.

References

1. CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness. http://competitions.cr.yp.to/caesar.html
2. Biryukov, A., Khovratovich, D.: PAEQ v1. http://competitions.cr.yp.to/round1/paeqv1.pdf
3. Biham, E., Biryukov, A., Shamir, A.: Cryptanalysis of skipjack reduced to 31 rounds using impossible differentials. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 12–23. Springer, Heidelberg (1999). doi:10.1007/3-540-48910-X_2
4. Biham, E., Keller, N.: Cryptanalysis of Reduced Variants of Rijndael, unpublished (2001). http://csrc.nist.gov/archive/aes/round2/conf3/papers/35-ebiham.pdf

5. Biham, E., Shamir, A.: Differential Cryptanalysis of the Data Encryption Standard. Springer, New York (1993)
6. Cid, C., Murphy, S., Robshaw, M.J.B.: Small scale variants of the AES. In: Gilbert, H., Handschuh, H. (eds.) FSE 2005. LNCS, vol. 3557, pp. 145–162. Springer, Heidelberg (2005). doi:10.1007/11502760_10
7. Daemen, J., Knudsen, L., Rijmen, V.: The block cipher Square. In: Biham, E. (ed.) FSE 1997. LNCS, vol. 1267, pp. 149–165. Springer, Heidelberg (1997). doi:10.1007/BFb0052343
8. Daemen, J., Rijmen, V.: The Design of Rijndael: AES - The Advanced Encryption Standard. Information Security and Cryptography. Springer, Heidelberg (2002)
9. Daemen, J., Rijmen, V.: Two-round aes differentials. Cryptology ePrint Archive, Report 2006/039 (2006). http://eprint.iacr.org/2006/039
10. Daemen, J., Rijmen, V.: Understanding two-round differentials in AES. In: Prisco, R., Yung, M. (eds.) SCN 2006. LNCS, vol. 4116, pp. 78–94. Springer, Heidelberg (2006). doi:10.1007/11832072_6
11. Andreeva, E., Bilgin, B., Bogdanov, A., Luykx, A., Mendel, F., Mennink, B., Mouha, N., Wang, Q., Yasuda, K.: PRIMATEs v1.02 Submission to the CAESAR Competition. http://competitions.cr.yp.to/round2/primatesv102.pdf
12. Ferguson, N., Kelsey, J., Lucks, S., Schneier, B., Stay, M., Wagner, D., Whiting, D.: Improved cryptanalysis of Rijndael. In: Goos, G., Hartmanis, J., Leeuwen, J., Schneier, B. (eds.) FSE 2000. LNCS, vol. 1978, pp. 213–230. Springer, Heidelberg (2001). doi:10.1007/3-540-44706-7_15
13. Grassi, L., Rechberger, C., Rønjom, S.: A New Structural-Differential Property of 5-Round AES. IACR Cryptology ePrint Archive, vol. 2017 (2017). http://eprint.iacr.org/2017
14. Grassi, L., Rechberger, C., Rønjom, S.: Subspace trail cryptanalysis and its applications to AES. IACR Trans. Symmetric Cryptology **2016**(2), 192–225 (2017). http://ojs.ub.rub.de/index.php/ToSC/article/view/571
15. Wu, H., Preneel, B.: A Fast Authenticated Encryption Algorithm. http://competitions.cr.yp.to/round1/aegisv1.pdf
16. Guo, J.: Marble Version 1.1. https://competitions.cr.yp.to/round1/marblev11.pdf
17. Knudsen, L.R.: Truncated and higher order differentials. In: Preneel, B. (ed.) FSE 1994. LNCS, vol. 1008, pp. 196–211. Springer, Heidelberg (1995). doi:10.1007/3-540-60590-8_16
18. Knudsen, L.R.: DEAL - a 128-bit block cipher. Technical report 151, Department of Informatics, University of Bergen, Norway, February 1998
19. Luby, M., Rackoff, C.: How to construct pseudorandom permutations from pseudorandom functions. SIAM J. Comput. **17**(2), 373–386 (1988)
20. Datta, N., Nandi, M.: ELmD v2.0. http://competitions.cr.yp.to/round2/elmdv20.pdf
21. Sun, B., Liu, M., Guo, J., Qu, L., Rijmen, V.: New insights on AES-Like SPN ciphers. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016. LNCS, vol. 9814, pp. 605–624. Springer, Heidelberg (2016). doi:10.1007/978-3-662-53018-4_22
22. Sun, B., Liu, M., Guo, J., Rijmen, V., Li, R.: Provable security evaluation of structures against impossible differential and zero correlation linear cryptanalysis. In: Fischlin, M., Coron, J.-S. (eds.) EUROCRYPT 2016. LNCS, vol. 9665, pp. 196–213. Springer, Heidelberg (2016). doi:10.1007/978-3-662-49890-3_8
23. Sun, B., Liu, Z., Rijmen, V., Li, R., Cheng, L., Wang, Q., Alkhzaimi, H., Li, C.: Links among impossible differential, integral and zero correlation linear cryptanalysis. In: Gennaro, R., Robshaw, M. (eds.) CRYPTO 2015. LNCS, vol. 9215, pp. 95–115. Springer, Heidelberg (2015). doi:10.1007/978-3-662-47989-6_5