Further Observations on the Structure of the AES Algorithm

Beomsik Song and Jennifer Seberry

Centre for Computer Security Research
School of Information Technology and Computer Science
University of Wollongong
Wollongong 2522, Australia
{bs81,jennifer_seberry}@uow.edu.au

Abstract. We present our further observations on the structure of the *AES* algorithm relating to the cyclic properties of the functions used in this cipher. We note that the maximal period of the linear layer of the *AES* algorithm is short, as previously observed by S. Murphy and M.J.B. Robshaw. However, we also note that when the non-linear and the linear layer are combined, the maximal period is dramatically increased not to allow algebraic clues for its cryptanalysis. At the end of this paper we describe the impact of our observations on the security of the *AES* algorithm. We conclude that although the *AES* algorithm consists of simple functions, this cipher is much more complicated than might have been expected.

Keywords: Cyclic Properties, SubBytes transformation, ShiftRows transformation, MixColumns transformation, Maximal period.

1 Introduction

A well-designed *SPN (Substitution Permutation Network)* structure block cipher, *Rijndael* [4] was recently (26. Nov. 2001) selected as the *AES (Advanced Encryption Standard)* algorithm [11]. This cipher has been reputed to be secure against conventional cryptanalytic methods [4, 8], such as *DC (Differential Cryptanalysis)* [1] and *LC (Linear Cryptanalysis)* [7], and throughout the *AES* process the security of the *AES* algorithm was examined with considerable cryptanalytic methods [2–4, 13, 14]. But despite the novelty of the *AES* algorithm [5], the fact that the *AES* algorithm uses mathematically simple functions [6, 12, 15, 16] has led to some commentators' concern about the security of this cipher. In particular, S. Murphy and M.J.B. Robshaw [15, 16] have modified the original structure of the *AES* algorithm so that the affine transformation used for generating the *S*-box (non-linear layer) is included in the linear layer, and have shown that any input to the modified linear layer of the *AES* algorithm is mapped to itself after 16 iterations of the linear transformation (the maximal period of the modified linear layer is 16 [15, 16]. Based on this observation, they have remarked that the linear layer of the *AES* algorithm may not be so effective at mixing data. At this stage, to make the concept of "mixing data" clear, we briefly define the

T. Johansson (Ed.): FSE 2003, LNCS 2887, pp. 223–234, 2003.

effect of mixing data, which Murphy and Robshaw considered. We define that in a set K consisting of n elements, if an input of a function F is mapped to itself after p iterations of the function, then the effect of mixing data is $e = \frac{p}{n}$.

In this paper, we present our further observations on the *AES* algorithm in terms of the cyclic properties of the *AES* algorithm. We examine the cyclic properties of the *AES* algorithm via each function in the original structure. We note that the maximal period of each function used in the *AES* algorithm is short, and that the maximal period of the composition of the functions used in the linear layer is also short. We however note that the composition of the non-linear layer and the linear layer dramatically increases the maximal period of the basic structure to highly guarantee the effect of mixing data. Specifically, we have found that:

- any input data block of the SubBytes transformation (non-linear layer) returns to the initial state after 277182 ($\approx 2^{18}$) repeated applications (the maximal period of the SubBytes transformation is 277182).
- any input data block of the ShiftRows transformation (in the linear layer) returns to the initial state after 4 repeated applications (the maximal period of the ShiftRows transformation is 4).
- any input data block of the MixColumns transformation (in the linear layer) returns to the initial state after 4 repeated applications as well (the maximal period of the MixColumns transformation is 4).
- when the ShiftRows transformation and the MixColumns transformation in the linear layer are considered together, the maximal period is 8.
- when the SubBytes transformation (non-linear layer) and the ShiftRows transformation (in the linear layer) are considered together, the maximal period is 554364 ($\approx 2^{19}$).

More importantly, we have found that the maximal period of the composition of the SubBytes transformation (non-linear layer) and the MixColumns transformation (in the linear layer) is 1,440,607,416,177,321,097,705,832,170,004,940 ($\approx 2^{110}$). Our observations indicate that the structure of the *AES* algorithm is good enough to bring magnificent synergy effects in mixing data when the linear and the non-linear layers are combined. In the last part of this paper we discuss the relevance of our observations to the security of the *AES* algorithm.

This paper is organised as follows: the description of the *AES* algorithm is presented in Section 2; the cyclic properties of the functions are described in Section 3; the impact of our observations on the security of the *AES* algorithm are discussed in Section 4; and the conclusion is given in Section 5.

2 Description of the AES Algorithm

The *AES* algorithm is an *SPN* structure block cipher, which processes variable-length blocks with variable-length keys (128, 192, and 256). In the standard case, it processes data blocks of 128 bits with a 128-bit Cipher Key [4, 11]. In this paper we discuss the standard case because the results of our observations will be similar in the other cases.

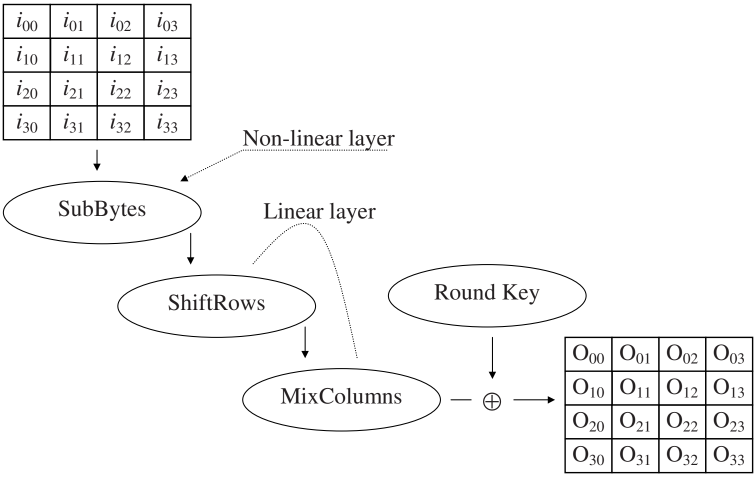
i_{00} i_{01} i_{02} i_{03}
i_{10} i_{11} i_{12} i_{13}
i_{20} i_{21} i_{22} i_{23}
i_{30} i_{31} i_{32} i_{33}
Non-linear layer
SubBytes
Linear layer
ShiftRows
Round Key
MixColumns
$\oplus$
O_{00} O_{01} O_{02} O_{03}
O_{10} O_{11} O_{12} O_{13}
O_{20} O_{21} O_{22} O_{23}
O_{30} O_{31} O_{32} O_{33}

Fig. 1. Basic structure of the *AES* algorithm.

As Figure 1 shows, the *AES* algorithm consists of a non-linear layer (SubBytes transformation) and linear layer (ShiftRows transformation and MixColumns transformation). Each byte in the block is bytewise substituted by the SubBytes transformation using a 256-byte S-box, and then every byte in each row is cyclicly shifted by a certain value (row #0: 0, row #1: 1, row #2: 2, row #3: 3) by the ShiftRows transformation. After this, all four bytes in each column are mixed through the MixColumns transformation by the matrix formula in Figure 2. Here, each column is considered as a polynomial over $GF(2^8)$, and multiplied with a fixed polynomial $03 \cdot x^3 + 01 \cdot x^2 + 01 \cdot x + 02$ (modulo $x^4 + 1$). After these operations, a 128-bit round key extended from the Cipher Key is XORed in the last part of the round. The MixColumns transformation is omitted in the last round (10th round), but before the first round a 128-bit initial round key is XORed through the initial round key addition routine. The round keys are derived from the Cipher Key by the following manner: Let us denote the columns in the Cipher Key by CK_0,CK_1,CK_2,CK_3, the columns in the round keys by K_0,K_1,K_2,...,K_{43}, and the round constants by *Rcon*. Then the columns in the round keys are

$$\begin{cases} K_0 = CK_0, K_1 = CK_1, K_2 = CK_2, K_3 = CK_3, & \\ K_n = K_{n-4} \oplus SubBytes(RotBytes(K_{n-1})) \oplus Rcon & \text{if } 4 \mid n \\ K_n = K_{n-4} \oplus K_{n-1} & \text{otherwise.} \end{cases}$$

3 Cyclic Properties of the Functions

In this section, we refer to cyclic properties of the functions used in the *AES* algorithm. The cyclic property of each function is examined first, and then the cyclic properties of the combined functions are obtained. For future reference, we define $f^n(I) = f \circ f \circ f \circ \cdots \circ f(I)$.

$$\begin{pmatrix} O_{0c} \\ O_{1c} \\ O_{2c} \\ O_{3c} \end{pmatrix} = \begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} \begin{pmatrix} i_{0c} \\ i_{1c} \\ i_{2c} \\ i_{3c} \end{pmatrix}$$

Fig. 2. Mixing of four bytes in a column.

3.1 Cyclic Property of Each Function

Cyclic Property of the SubBytes Transformation. From the analysis of 256 substitution values in the S-box, we have found the maximal period of the SubBytes transformation (non-linear layer).

Property 1 *Every input byte of the S-box returns to the initial value after some t repeated applications of the substitution. In other words, for any input i of the S-box=S,*

$$S^t(i) = i.$$

The 256 values of the input byte can be classified into five small groups as in Table 1 according to the values of t. The number of values in each group (the period of each group) is 87, 81, 59, 27, and 2 respectively.

In Table 1, each value in each group is mapped to the value next to it. For example 'f2' → '89' → 'a7' → $\cdots$ → '04' → 'f2', and '73' → '8f' → '73'. From Property 1, we can see that although the S-box is a non-linear function, every input block of the SubBytes transformation is mapped to itself after some repeated applications of the SubBytes transformation. Indeed, we see that if each byte in an input block (16 bytes) is '8f' or '73' (in group 5), then this block returns to the initial state after just two applications of the SubBytes transformation. From Property 1, if we consider the *L.C.M (Least Common Multiple)* of 87, 81, 59, 27, and 2, then we find the following cyclic property of the SubBytes transformation.

Property 2 *For any input block I of the SubBytes transformation,*

$$SubBytes^{277182}(I) = I.$$

That is, the maximal period of the SubBytes transformation is 277182. The minimal period of the SubBytes transformation is 2 when each byte in the input block I is '8f' or '73'.

Cyclic Property of the ShiftRows Transformation. The cyclic property of the ShiftRows transformation is immediately found from the shift values (row #0: 0, row #1: 1, row #2: 2, row #3: 3) in each row.

Table 1. Classifying the substitution values in the S-box.

Group #1 (maximal period: 87)

f2, 89, a7, 5c, 4a, d6, f6, 42, 2c, 71, a3, 0a, 67, 85, 97, 88, c4, 1c, 9c, de, 1d, a4, 49, 3b, e2, 98, 46, 5a, be, ae, e4, 69, f9, 99, ee, 28, 34, 18, ad, 95, 2a, e5, d9, 35, 96, 90, 60, d0, 70, 51, d1, 3e, b2, 37, 9a, b8, 6c, 50, 53, ed, 55, fc, b0, e7, 94, 22, 93, dc, 86, 44, 1b, af, 79, b6, 4e, 2f, 15, 59, cb, 1f, c0, ba, f4, bf, 08, 30, 04

Group #2 (maximal period: 81)

7c, 10, ca, 74, 92, 4f, 84, 5f, cf, 8a, 7e, f3, 0d, d7, 0e, ab, 62, aa, ac, 91, 81, 0c, fe, bb, ea, 87, 17, f0, 8c, 64, 43, 1a, a2, 3a, 80, cd, bd, 7a, da, 57, 5b, 39, 12, c9, dd, c1, 78, bc, 65, 4d, e3, 11, 82, 13, 7d, ff, 16, 47, a0, e0, e1, f8, 41, 83, ec, ce, 8b, 3d, 27, cc, 4b, b3, 6d, 3c, eb, e9, 1e, 72, 40, 09, 01

Group #3 (maximal period: 59)

00, 63, fb, 0f, 76, 38, 07, c5, a6, 24, 36, 05, 6b, 7f, d2, b5, d5, 03, 7b, 21, fd, 54, 20, b7, a9, d3, 66, 33, c3, 2e, 31, c7, c6, b4, 8d, 5d, 4c, 29, a5, 06, 6f, a8, c2, 25, 3f, 75, 9d, 5e, 58, 6a, 02, 77, f5, e6, 8e, 19, d4, 48, 52

Group #4 (maximal period: 27)

ef, df, 9e, 0b, 2b, f1, a1, 32, 23, 26, f7, 68, 45, 6e, 9f, db, b9, 56, b1, c8, e8, 9b, 14, fa, 2d, d8, 61

Group #5 (maximal period: 2)

73, 8f

* Each value in each table is followed by its substitution value

Property 3 *For any input block I of the ShiftRows transformation,*

$$ShiftRows(ShiftRows(ShiftRows(ShiftRows(I)))) = I.$$

In other words, the maximal period of the ShiftRows transformation is 4. The minimal period of the ShiftRows transformation is 1 when all bytes in the input block I are the same.

Cyclic Property of the MixColumns Transformation. In terms of the MixColumns transformation, we have found that the maximal period of this function is 4. Let us look carefully once again at the algebraic structure of the MixColumns transformation described in Section 2. As realised, each input column (four bytes) is considered as a polynomial over $GF(2^8)$ and multiplied modulo $x^4 + 1$ with a fixed polynomial $b(x) = 03 \cdot x^3 + 01 \cdot x^2 + 01 \cdot x + 02$. This can be written as a matrix multiplication, as in Figure 2, and from this matrix formula we can obtain the relation between an input column (I_c) and the corresponding output column (O_c). Hence, we can find that for any input column I_c (four bytes),

$$M(M(M(M(I_c)))) = I_c$$

where M is the matrix multiplication described in Figure 2. When all four bytes of I_c are the same,

$$M(I_c) = I_c.$$

If we now consider one input block (four columns) of the MixColumns transformation described in Figure 1, then we find the following property.

Property 4 *For any input block I (16 bytes) of the MixColumns transformation,*

$$MixColumns(MixColumns(MixColumns(MixColumns(I)))) = I.$$

In other words, the maximal period of the MixColumns transformation is 4. The minimal period of the MixColumns transformation is 1 when the bytes are the same in each column.

3.2 Cyclic Properties of Combined Functions

We now refer to the cyclic properties of cases when the above functions are combined. We first refer to the maximal period of the linear layer (the composition of the ShiftRows transformation and the MixColumns transformation). In the case when the ShiftRows transformation and the MixColumns transformation are considered together, we obtain the maximal period of the linear layer.

Property 5 *Any input block I of the linear layer is mapped to itself after 8 repeated applications of the linear layer. In other words, the maximal period of the linear layer is 8.*

From the two minimal periods referred to in Property 3 and Property 4 we obtain the following property.

Property 6 *Any input block I of the linear layer, in which all bytes are the same, is mapped to itself after one application of the linear layer. That is, the minimal period of the linear layer is 1.*

When the SubBytes transformation (non-linear layer) and the ShiftRows transformation (in the linear layer) are combined, we obtain the following cyclic property from the *L.C.M* of the two maximal periods referred to in Property 2 and Property 3.

Property 7 *Any input block I of the composition of the SubBytes transformation and the ShiftRows transformation is mapped to itself after 554364 repeated applications of the composition. In other words, the maximal period of the composition of the SubBytes transformation and the ShiftRows transformation is 554364.*

Property 8 *In Property 7, if all bytes in the input block I are the same and are either '73' or '8f', then this block is mapped to itself after two repeated applications of the composition. That is, the minimal period of the composition of the SubBytes transformation and the ShiftRows transformation is 2.*

More importantly, we show that although the maximal periods of both the non-linear layer and the linear layer are short, the maximal period is surprisingly increased in the composition of the non-linear layer and the MixColumns transformation. We first change the order of the SubBytes transformation and the ShiftRows transformation with each other as shown in Figure 3 (b) (the order of these two functions is changeable).

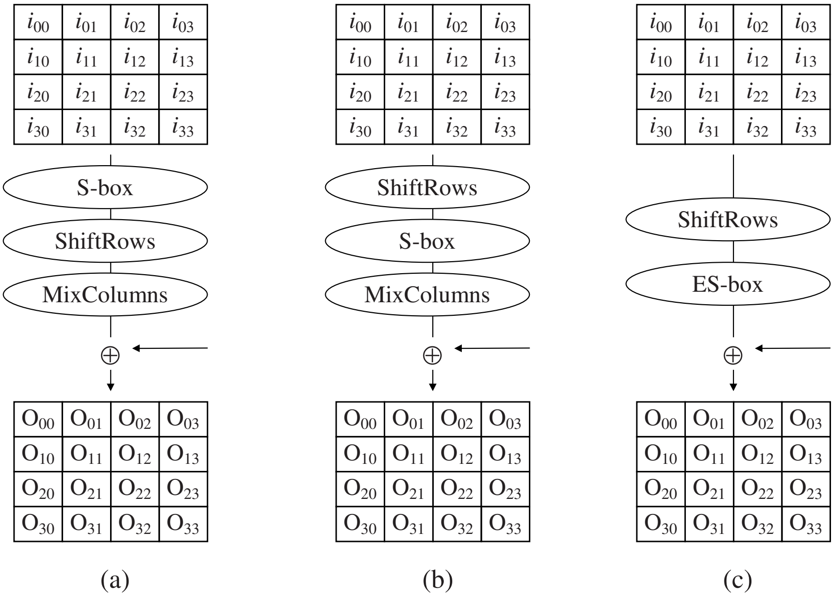
i_{00} i_{01} i_{02} i_{03}
i_{10} i_{11} i_{12} i_{13}
i_{20} i_{21} i_{22} i_{23}
i_{30} i_{31} i_{32} i_{33}
S-box
ShiftRows
MixColumns
$\oplus$
O_{00} O_{01} O_{02} O_{03}
O_{10} O_{11} O_{12} O_{13}
O_{20} O_{21} O_{22} O_{23}
O_{30} O_{31} O_{32} O_{33}
(a)
ShiftRows
S-box
MixColumns
(b)
ShiftRows
ES-box
(c)

Fig. 3. Re-ordering of SubBytes and ShiftRows.

We then consider the S-box and the MixColumns transformation together. As a result, we obtain an extended S-box, ES-box, which consists of 2^{32} non-linear substitution paths, as shown in Figure 3 (c) and Table 2. Now, using the same idea used to obtain Property 1, we classify the 2^{32} four-byte input values of the ES-box into 52 small groups according to their periods.

The number of values in each group (the period of each group) is 1,088,297,796 ($\approx 2^{30}$), 637,481,159 ($\approx 2^{29}$), 129,021,490 ($\approx 2^{27}$), 64,376,666 ($\approx 2^{26}$), and so on. Table 3 shows the classification of all substitution values in the ES-box, which has been obtained from our analysis (see the appendix for more details).

From these values of the periods we finally find that the maximal period of the composition of the SubBytes transformation (non-linear layer) and

Table 2. *ES*-box.

I	0x00000000	0x00000001	• • • •	0xabcdef12	• • •	0xffffffff
	↓	↓	↓ ↓	↓	↓ ↓	↓
ES(I)	0x63636363	0x7c7c425d	• • • •	0x0eb03a4d	• • •	0x16161616

Table 3. Classifying the substitution values in the *ES*-box.

1088297796, **637481159**, 637481159, 637481159, 637481159, **129021490**, 129021490, 129021490, 129021490, 64376666, 64376666, **11782972**, 39488, 16934, 13548, 13548, 10756, 7582, 5640, 5640, 3560, 1902, 1902, 548, 548, 136, 90, 90, 87, 81, 59, 47, 47, 47, 47, 40, 36, 36, 27, 24, 21, 21, 15, 15, 12, 8, 4, 4, 4, 2, 2, 2

e.g. Period of group #1 : 1088297796, Period of group #2 : 637481159,
Period of group #6 : 129021490, Period of group #12 : 11782972.

the MixColumns transformation (in the linear layer) is 1,440,607,416,177,321, 097,705,832,170,004,940 ($\approx 2^{110}$). Here, we note that the maximal period of this composition is the largest L.C.M of any four values above. This is because one input block consists of four columns.

We now discuss shorter periods of the composition of the the SubBytes transformation and the MixColumns transformation which cryptanalysts may be concerned about. We first refer to the minimal period. In very rare cases where each column in an input block I is '73737373', '8f8f8f8f', '5da35da3', 'c086c086', 'a35da35d' or '86c086c0' (each of these values is mapped to itself after 2 iterations of *ES*-box: see the appendix), for example,

$$\mathrm{I} = \text{8f8f8f8f c086c086 73737373 5da35da3},$$

the period of the composition of the SubBytes transformation and the MixColumns transformation is 2 (this is the minimal period of the composition of the SubBytes transformation and the MixColumns transformation). We next refer to the periods of the composition of the SubBytes transformation and the MixColumns transformation for input blocks in which all bytes are the same. If all bytes in an input block I of the composition of the SubBytes transformation and the MixColumns transformation are the same, then this block leads to an output block in which all bytes are the same. In this case, the period of the composition of the SubBytes transformation and the MixColumns transformation is the same as the period of the *S*-box referred to in Table 1. For example, if the bytes in an input block I of the combined function of the SubBytes transformation and the MixColumns transformation are all ‘f2’, then this block is mapped to itself after 87 iterations of this combined function (see Group #1 in Table 1 and Period 87 in the appendix).

In the next section, we discuss that input blocks having short periods could provide some algebraic clues for cryptanalysis, as some previous works have expected [15, 16]. We show that input blocks having short periods, when compared with others, could have relatively simple hidden algebraic relations with the corresponding output blocks. However, we also note that although in some cases the composition of the non-linear layer and the linear layer has short periods which could provide some algebraic clues for cryptanalysis, the key schedule of the *AES* algorithm does not allow the short periods to go on.

4 Impact on the Security of the AES Algorithm

In this section, we discuss the impact of our observations on the security of the *AES* algorithm. We show that input blocks having short periods (the effect of mixing data $e = \frac{p}{n}$ is very small) are apt to give hidden algebraic clues for cryptanalysis when compared with others. To do this, we first find some input blocks having shortest periods in the composition of the non-linear layer and the linear layer (the SubBytes transformation+the ShiftRows transformation+the MixColumns transformation).

Property 9 *For any input block I of the composition of the non-linear layer and the linear layer (the SubBytes transformation, the ShiftRows transformation, and the MixColumns transformation), if all bytes in I are the same, then all bytes in the output block are also the same. In this case, the composition of the non-linear layer and the linear layer is equivalent to the S-box because the ShiftRows transformation and the MixColumns transformation do not affect the data transformation.*

Property 10 *For any input block I of the composition of the non-linear layer and the linear layer, if all bytes in I are equal to i (any value), then the period of the composition of the non-linear layer and the linear layer for this input block is the same as the period of the S-box for i.*

For example, if the bytes in an input block I of the composition of the non-linear layer and the linear layer are all ‘ef’, then this input block is mapped to itself after 27 iterations (the period of the S-box for ‘ef’ is 27 as given in Table 1). This means that the effect of mixing data of the composition of the non-linear layer and the linear layer is $e = \frac{27}{2^{128}}$ for this input block (2^{128} is the number of all possible blocks presented by 128 bits).

Property 11 *In Property 10, if all bytes in I are the same and are either ‘73’ or ‘8f’, then I is mapped to itself after 2 iterations of the composition of the non-linear layer and the linear layer. In other words, the minimal period of the composition of the non-linear layer and the linear layer is 2 (the minimal effect of mixing data of the non-linear layer and the linear layer is $e = \frac{2}{2^{128}}$).*

We now show that input blocks having short periods could provide some algebraic clues for cryptanalysis if the key schedule of the *AES* algorithm were not well-designed. Let us assume that contrary to the original key schedule of the *AES* algorithm, for any Cipher Key in which all bytes are the same, a certain key schedule generates round keys in which each round key has all its bytes the same. (This does not actually appear in the original key schedule.) For example, suppose that the initial round key consists of all '78', that the first round key consists of all '6f', ..., and that the tenth round key consists of all '63'. Then, if we consider the encryption procedure, we see, from Property 9, that any plaintext in which all bytes are the same leads to a ciphertext in which all bytes are the same. This means that if anyone uses, for encryption, a Cipher Key in which all bytes are the same, then attackers will easily become aware of this fact with a chosen plaintext in which all bytes are the same. As long as the attackers realise this fact, it will be easy to find the Cipher Key. They will find the Cipher Key from 256 key searches. However, we note that this scenario does not occur with the original key schedule of the *AES* algorithm because plaintexts having short periods are not able to keep up the short periods in the original key schedule. For example, we consider the most simple case where a plaintext, in which all bytes are '73', is encrypted with a Cipher Key in which all bytes are '00'. In this case, by Property 11, the period of the composition of the non-linear layer and the linear layer is 2 for the intermediate text

$$I_0 = 73737373\ 73737373\ 73737373\ 73737373$$

after the initial round key addition. However, we have found that the period of the composition of the SubBytes transformation (non-linear layer) and the MixColumns transformation (in the linear layer) becomes 1,088,297,796 ($\approx 2^{30}$) for the intermediate text

$$I_1 = \text{edececec edececec edececec edececec}$$

after the first round key addition. We here emphasise once again that although the combined function of the non-linear layer and the linear layer of the *AES* algorithm has some short periods in rare cases, the key schedule does not allow these short periods to go on, thus denying algebraic clues for its cryptanalysis.

5 Conclusions

We have summarised our further observations on the *AES* algorithm relating to the cyclic properties of this cipher. Specifically, we have shown that the maximal period of each function used in the *AES* algorithm is short, and that the maximal period of the composition of the functions used in the linear layer is short as well. However, more importantly, we have also shown that the well-designed structure brings remarkable synergy effects in the cyclic property of this cipher when the linear layer and the non-linear layer are combined. We note that the structure of the *AES* algorithm is good enough to guarantee high data mixing effects. We also note that although the composition of the non-linear layer and the linear layer of the *AES* algorithm has, in some cases, short periods which could provide some algebraic clues for its cryptanalysis, the well-designed key schedule does not allow these short periods to go on. We believe that the combination of the simple functions in the well-designed structure is one of the advantages of the *AES* algorithm although some research studies have been recently making considerable progress [9, 10] in the cryptanalysis of the *AES*-like block ciphers.

References

1. E. Biham and A. Shamir, "Differential cryptanalysis of DES-like cryptosystems", *J. Cryptology*, Vol.4, pp.3-72, 1991.
2. E. Biham and N. Keller, "Cryptanalysis of Reduced Variants of Rijndael", http://csrc.nist.gov/encryption/aes/round2/conf3/aes3papers.html, 2000.
3. H. Gilbert and M. Minier, "A Collision Attack on 7 Rounds of Rijndael", *Proceeding of the Third Advanced Encryption Standard Candidate Conference*, NIST, pp.230-241, 2000.
4. J. Daemen and V. Rijmen, "*AES* Proposal: Rijndael", http://csrc.nist.gov/encryption/aes/rijndael/Rijndael.pdf, 1999.
5. J. Daemen and V. Rijmen, "Answer to New Observations on Rijndael", *AES Forum comment*, August 2000, http://www.esat.kuleuven.ac.be/∼rijmen/rijndael/.
6. L. Knudsen and H. Raddum, "Recommendation to NIST for the AES", *Second round comments to NIST*, May 2000, http://csrc.nist.gov/encryption/aes/round2/comments/.
7. M. Matsui, "Linear cryptanalysis method for DES cipher", *Advances in Cryptology-Eurocrypt'93*, Lecture Notes in Computer Science, Springer-Verlag, pp.386-397, 1993.
8. M. Sugita, K. Kobara, K. Uehara, S. Kubota, and H. Imai, "Relationships among Differential, Truncated Differential, Impossible Differential Cryptanalyses against Word-oriented Block Ciphers like Rijndael, E2", *Proceeding of the Third AES Candidate Conference*, 2000.
9. N. Courtois and J. Pieprzyk, "Cryptanalysis of Block Ciphers with Overdefined Systems of Equations", *IACR eprint*, April 2002, http://www.iacr.org/complete/.
10. N. Courtois and J. Pieprzyk, "Cryptanalysis of Block Ciphers with Overdefined Systems of Equations", *Proceeding of ASIACRYPT'2002*, Lecture Notes In Computer Science Vol.2501, pp.267-287, 2002.
11. National Institute of Standard and Technology, "Advanced Encryption Standard(*AES*)", FIPS 197, 2001.
12. N. Ferguson, R. Schroeppel, and D. Whiting, "A simple algebraic representation of Rijndael", *Proceeding of SAC'2001*, Lecture Notes In Computer Science Vol.2259, pp.103-111, 2001.
13. N. Ferguson, J. Kelsey, S. Lucks, B. Schneier, M. Stay, D. Wagner, and D. Whiting, "Improved Cryptanalysis of Rijndael", *Fast Software Encryption Workshop '2000*, Preproceeding, 2000.
14. S. Lucks, "Attacking Seven Rounds of Rijndael under 192-Bit and 256-Bit Keys", *Proceeding of the Third Advanced Encryption Standard Candidate Conference*, NIST, pp.215-229, 2000.
15. S. Murphy and M.J.B Robshaw, "New Observations on Rijndael", *AES Forum comment*, August 2000, http://www.isg.rhul.ac.uk/∼sean/.
16. S. Murphy and M.J.B Robshaw, "Further Comments on the Structure of Rijndael", *AES Forum comment*, August 2000, http://www.isg.rhul.ac.uk/∼sean/.

Appendix: Grouping in the ES-Box

Periods	Elements in each group
1088297796	00000003, 7b7b4b53, ●●●●●●●●●●, 4487de39
637481159	00000002, 77775f4b, ●●●●●●●●●●, 3943ffc4
637481159	00000004, f2f2cb5a, ●●●●●●●●●●, a6284276
637481159	00000006, 6f6f777b, ●●●●●●●●●●, 24c3a2a6
637481159	00000008, 303096c5, ●●●●●●●●●●, d4f75ed0
129021490	00000001, 7c7c425d, ●●●●●●●●●●, 40f39ed7
129021490	00000007, c5c59234, ●●●●●●●●●●, 25322e95
129021490	00000009, 0101c5a7, ●●●●●●●●●●, f8bc508a
129021490	00000010, caca832a, ●●●●●●●●●●, 9660fca0
64376666	00000016, 47470f2b, ●●●●●●●●●●, c50ccf88
64376666	00000142, 330d8ce2, ●●●●●●●●●●, e401999a
11782972	000000ea, 878754b0, ●●●●●●●●●●, 638a2857
39488	00020002, 4b5f4b5f, ●●●●●●●●●●, 30a530a5
16934	00010001, 5d425d42, ●●●●●●●●●●, 6ad56ad5
13548	00023af9, 468fbf7b, ●●●●●●●●●●, 6b5493f6
13548	0005fde6, a1c7299d, ●●●●●●●●●●, 8bf1558a
10756	001004ad, e474f2ac, ●●●●●●●●●●, 245557ee
7582	00070007, 34923492, ●●●●●●●●●●, d740d740
5640	00022db0, 60198ddf, ●●●●●●●●●●, feb74bd1
5640	0015e186, 91861d8c, ●●●●●●●●●●, 5d50a4a6
3560	00094090, ac1ad06d, ●●●●●●●●●●, f6110e3e
1902	0000c22b, b73b421a, ●●●●●●●●●●, 07a9ec2e
1902	0021e4f9, 2aa0fc18, ●●●●●●●●●●, 76a21d37
548	00b800b8, 7d727d72, ●●●●●●●●●●, 05a905a9
548	00c600c6, d601d601, ●●●●●●●●●●, 85708570
136	01d266c5, a9fe5e55, ●●●●●●●●●●, f554d80d
90	02338d7f, 3fdf63b8, ●●●●●●●●●●, 3c0c694e
90	0304c1ca, f778e5ef, ●●●●●●●●●●, 8683dfa2
87	f2f2f2f2, 89898989, ●●●●●●●●●●, 04040404
81	7c7c7c7c, 10101010, ●●●●●●●●●●, 01010101
59	00000000, 63636363, ●●●●●●●●●●, 52525252
47	0112dc34, 267c8afb, ●●●●●●●●●●, c406421d
47	018b9ded, b4b1024d, ●●●●●●●●●●, 32926cc7
47	024db4b1, 95eed67c, ●●●●●●●●●●, 9ded018b
47	03c975a2, 2d5cc9b9, ●●●●●●●●●●, c0c8d6db
40	0aff4adf, bcb47f4e, ●●●●●●●●●●, 1864fa71
36	03d603d6, 7af77af7, ●●●●●●●●●●, 3e0a3e0a
36	07f107f1, 0d690d69, ●●●●●●●●●●, 17a517a5
27	efefefef, dfdfdfdf, ●●●●●●●●●●, 61616161
24	03d503d5, 8bf38bf3, ●●●●●●●●●●, c6abc6ab
21	050f050f, 514c514c, ●●●●●●●●●●, e344e344
21	0f050f05, 4c514c51, ●●●●●●●●●●, 44e344e3
15	0e6e0e6e, c3f7c3f7, ●●●●●●●●●●, ecbeecbe
15	6e0e6e0e, f7c3f7c3, ●●●●●●●●●●, beecbeec
12	0327266c, 1eaab216, ●●●●●●●●●●, 837b2f79
8	cac4cac4, a4cca4cc, ●●●●●●●●●●, 4a2d4a2d
4	01828fc8, 5627aa2f, 8fc80182, aa2f5627
4	27aa2f56, c801828f, 2f5627aa, 828fc801
4	a37dadf5, 7dadf5a3, adf5a37d, f5a37dad
2	73737373, 8f8f8f8f
2	5da35da3, c086c086
2	a35da35d, 86c086c0