Mini-ciphers: a reliable testbed for cryptanalysis?

Jorge Nakahara Jr[1] and Daniel Santana de Freitas[2]

[1] EPFL, Lausanne, Switzerland
jorge.nakahara@epfl.ch
[2] UFSC, Santa Catarina, Brazil
santana@inf.ufsc.br

Abstract. This paper reports on higher-order square analysis of the AES cipher. We present experimental results of attack simulations on mini-AES versions with word sizes of 3, 4, 5, 6 and 7 bits and describe the propagation of higher-order Λ-sets inside some of these distinguishers. A possible explanation of the length of the square distinguishers uses the concept of higher-order derivatives of discrete mappings.

1 Introduction

This paper describes experimental results of higher-order square attacks [6, 11, 14] applied to scaled-down or mini-versions of the AES cipher. We detail the internal structures (patterns) of square distinguishers but do not apply them to key-recovery attacks. Although this process is straightforward, our aim is to determine how many rounds these distinguishers can cover without exhausting the codebook.

Scaling can be done either upwards or downwards. In our context, scaling means shrinking or enlarging each internal cipher component accordingly, resulting in a mini-version or a super-version of the original cipher, operating on smaller or larger blocks of data, respectively. Thus, scaling does not mean changing the number of rounds, and therefore, it should not be misunderstood as reduced-round versions of a cipher.

In this paper, our experiments are focused on scaled-down versions only. Nonetheless, scaling-up is also possible, and is a flexibility of many cryptographic algorithm usually ignored and neglected by designers. This fact may be due to increased storage requirements (e.g. for S-boxes) or because of operations involving words whose size is not a power of 2, which do not fit exactly into the word size of popular microprocessors, thus leading to inneficient implementations.

The attractiveness of mini-ciphers comes from the fact that they provide a convenient testbed for new cryptanalytic ideas, and also to obtain experimental data such as success rate, and to corroborate (or not) theoretical predictions, because their reduced size lead to smaller attack complexities. Examples of algorithms which have mini-versions analysed include RC4 [12], RadioGatún [9], IDEA [3] and the AES [18, 19, 4, 23].

Dagstuhl Seminar Proceedings 09031
Symmetric Cryptography
http://drops.dagstuhl.de/opus/volltexte/2009/1961

Some algorithms such as DES [7], CAST and Serpent are harder to scale (either up or down) because they contain operations not amenable to scaling such as bit permutations, bitwise rotation and non-surjective mappings. Thus, cryptanalysts are left with too many degrees of freedom for choosing the parameters and components of mini-versions. In the end, the contrived mini-ciphers are somehow just *ad hoc* designs.

On the other hand, there are ciphers with implicit scaled-down versions, such as IDEA [15] and AES [1], since their design is word-oriented. Thus, scaling is straightforward, based on a single parameter, the word size. It is notable that designers rarely provide scaled versions in full detail. One often missing part is the key schedule algorithm. For example, in IDEA, the bit-rotation operation in its key schedule is problematic to scale. The choice of the rotation amount may depend on the design criteria (which is not clear or definitive).

In the most favorable cases, the ciphers are fully parameterized for block, key and word sizes, such as RC5 and RC6 [21, 22]. It is important to notice that scaling should be done carefully in order not to affect the overall strength of the algorithm. Why? Because it is implicitly expected that attack results (and conclusions about the behaviour of) mini-versions can be extrapolated to the original algorithm. This assumption is not always justified.

There are limits to the amount of shrinking that can be performed in mini-cipher designs. For instance, 2-bit S-boxes are linear mappings, and such extreme cases destroy the main (if not the only) nonlinear component of many ciphers. Thus, one cannot expect to extend results on such mini-versions to the original cipher, because the inherent strength of the cipher has been jeopardized.

This paper is organized as follows: Sect.2 describes tentatives of defining mini-versions of the AES, and its purposes; Sect.3 describes our tentative mini-versions of AES, and the rationale behind those designs; Sect.4 provides some background on square attacks and terminology; Sect. 5 discuss the issue of higher-order derivatives of discrete mappings; Sect. 6 describes distinguishers obtained experimentally for mini-versions of the AES; Sect. 7 summarizes the paper.

2 Mini-AESs

Several mini-versions of the AES cipher have already been reported. For instance:

- M. Musa, E. Schaefer, S. Wedig [18]: 4-bit words, GF(2)[x]/($x^4 + x + 1$), 16-bit block, 2×2 state, purpose: explain AES operations
- R.C.-W. Phan [19]: 4-bit words, 2×2 state, 16-bit key, GF(2)[x]/(x^4+x+1), purpose: explain AES operations
- C. Cid *et al.* [4]: mini-versions are denoted SR(n,r,c,e) and SR*(n,r,c,e), with n= #rounds, r= #rows, c= #columns, 'e' is word size, purpose: algebraic cryptanalysis
- N. Courtois [5]: CTC2, 3-bit words, GF(2)[x]/(x^3+x+1), purpose: algebraic cryptanalysis

– R.-P. Weinmann [23]: 3-bit words, GF(2)[x]/(x^3+x+1), 2×2 state, purpose: algebraic cryptanalysis

3 Our Own Trial Designs

Mini-ciphers described in the literature, such as the ones listed in Sect. 2, usually do not provide enough details to allow full implementation. Often, something is missing such as the key scheduling algorithm. The source code of these mini-versions, which could clarify these gaps, is almost always absent, too. A reason might be the purpose of each mini-version. Some of them were designed for didactic purposes only and never meant to be used in practice. Our aim, though, is practical experimentation of cryptanalytic attacks.

We have designed mini-versions of AES based only on the word size t in bits, for $3 \leq t \leq 7$. Source code in ANSI C of all of these mini-versions of AES is freely available via the authors or at

http://www.dagstuhl.de/en/program/calendar/semhp/?semnr=09031

Table 1 lists the main parameters of our mini-versions of AES for different word sizes (denoted t). The $t \times t$ S-boxes we used are listed in the appendix. The finite fields used are GF(2^t) = GF(2)[x]/(m(x)), where the irreducible polynomials $m(x)$ are listed in Table 1. These polynomials were chosen at random. The round operations AK_i, SB, SR and MC for the mini-versions are the same as for the original AES, but on a reduced scale. In particular, for MC, the coefficients of the MDS matrix are the t least significant bits of the corresponding coefficients of the original AES MDS matrix. The same approach holds for constants used in the key schedule. The number of rounds is the same as those used in AES, even for different key sizes.

Table 1. Parameters of mini-AES ciphers and AES.

t (bits)	$m(x)$	State Size ($16t$ bits)	Key Sizes (bits)
3	x^3+x+1	48	48; 72; 96
4	x^4+x+1	64	64; 96; 128
5	$x^5+x^4+x^3+x^2+1$	80	80; 120; 160
6	$x^6+x^5+x^3+x^2+1$	96	96; 144; 192
7	$x^7+x^6+x^5+x^4+x^3+x^2+1$	112	112; 168; 224
8	$x^8+x^4+x^3+x+1$	128	128; 192; 256

4 Square Attacks

The square attack originated as a dedicated attack on the Square block cipher [6]. This technique has similarities with the multiset attack [2], the saturation attack

[16] and with integral cryptanalysis [10, 13]. All of these techniques operate in a chosen-plaintext (CP) setting.

A fundamental concept in a square attack is the Λ-set [6], which is a *multiset*, that is, a set with multiplicities, containing b full n-bit text blocks, where n is the block size and b is typically a power of 2. These n-bit text blocks are analysed by tracing (not necessarily contiguous) t-bit words, $t < n$, across the different round operations. For example,

$$\{(0||1||2||3), (1||2||2||1), (3||1||2||2), (2||2||2||1),\\ (7||5||2||0), (4||5||2||7), (5||5||2||4), (6||5||2||4)\} \quad (1)$$

is a Λ-set with $b = 2^t = 8$ text blocks, each of which is 12 bits wide ($n = 12$). Double vertical bars, $||$, mean concatenation. We further consider each of these 2^t text blocks as a concatenation of four t-bit words ($t = 3$), and thus, keep track of particular *patterns* in these t-bit words across the 2^t text blocks. These t bits are often composed of contiguous bits that respect word boundaries, hence the use of the term *word*. The patterns of interest are the following:

- if the t-bit word in a Λ-set assumes each of the values 0 to $2^t - 1$ exactly once, then the word contains a permutation and is called an *active* word, denoted 'A'. This is the case of the word formed by the first 3-bit word of each element in the multiset (1), which is $\{0, 1, 3, 2, 7, 4, 5, 6\}$;
- if the t-bit word assumes an arbitrary *constant* value, it is called *passive* and is denoted 'P'. This is the case of the third set of three bits in (1), $\{2, 2, 2, 2, 2, 2, 2, 2\}$;
- if the t-bit word contains an *even* number of repetitions of some arbitrary values (each element has even multiplicity), the word is called *even*, and is denoted 'E'. This is the case of the second set of three bits in (1), $\{1, 2, 1, 2, 5, 5, 5, 5\}$;
- if the sum of all t-bit values in a given word under some operator $\boxdot$, results in a *predictable* amount, then this word is called *balanced*, and denoted 'B';
- otherwise, if the $\boxdot$-sum results in an unpredictable value, the word is denoted '?'. This is the case of the fourth set of three bits in (1), which is $\{3, 1, 2, 1, 0, 7, 4, 4\}$, with $\boxdot = \oplus$ that is exclusive-or.

Multiset attacks are aimed at the bijective cipher components. In a sense, this technique uses permutations (e.g. 'A' words) to attack other permutations (e.g. S-boxes). Moreover, ciphers that operate on neatly partitioned words are the main targets. A typical square attack starts with Λ-sets with a single active word. After crossing multiple rounds, this 'A' word progressively looses its internal patterned structure, degrading into 'E' words (not permutations anymore), then 'B' words (not even anymore), and finally '?' words (no detectable pattern). Only the initial (plaintext) can be controlled by an adversary. The propagation of words across multiple rounds is unconstrained by the adversary. On one hand, this fact leads to square distinguishers holding with certainty. On the other hand, it is not yet known how to construct probabilistic nor iterative square

distinguishers, like differential characteristics (which are typically constructed by concatenating smaller characteristics).

Square distinguishers can be viewed as exploiting partial functions of the encryption mapping, since some of the plaintext bits are fixed (to an arbitrary constant value: 'P') while other bits are assigned all possible values ('A').

Since the exclusive-or ($\oplus$) operator is used to combine round subkeys with intermediate cipher data, it is natural to use $\oplus = \boxdot$ as the operator for computing the sum, because this value becomes independent of the round subkeys (an even number of plaintexts are added). The expected exclusive-or sum for balanced words is *zero*. All multiset attacks reported in the literature have used either the $\oplus$ or the modular additive sum [10, 20] as the operator in the distinguishers.

The original square attack in [6] has order 1 because it uses one active t-bit word only (while the remaining t-bit words are passive). Square attacks of nth-order use n t-bit words (multi-words) at once or a single tn-bit active word, while the remaining words are passive. The data requirement is proportional to initial active word size: 2^{tn} chosen plaintexts (CP).

Reasoning about the propagation of different kinds of patterns across each round transformation is relatively straightforward for 1st-order square distinguishers. Higher-order analysis, though, requires us to change focus from a single word to several words at once (these multi-words are denoted with '*' in the distinguishers). But, explaining how far such patterns propagate until they degenerate into '?' words requires another mechanism. To explain the length of higher-order distinguishers we use the concept of higher-order derivatives.

5 Differences and Higher-order Diferences

In [14], Lai related the notion of higher-order differential with that of derivatives of discrete mappings:

Definition 1. *Let $(S, +)$ and $(T, +)$ be Abelian groups. For a function $f : S \to T$, the derivative of f at the point $a \in S$ is defined as*

$$\Delta_a f(x) = f(x + a) - f(x),$$

where $-a$ is inverse of a in T.

The ith-derivative of f at the points $a_1, \ldots, a_i$ is defined as

$$\Delta^{(i)}_{a_1,\ldots,a_i} f(x) = \Delta_{a_i}(\Delta^{(i-1)}_{a_1,\ldots,a_{i-1}} f(x)).$$

Proposition 1. *For a function $f : S \to T$ with degree d, the dth-derivative of f is a constant.*

Lemma 1. *For a function $f : S \to T$ with degree d, the $(d + 1)$th-derivative of f is zero.*

For AES and its mini-versions, $(S, +)$ and $(T, +)$ are $(GF(2), \oplus)$, and the mappings of interest are the S-boxes and in particular the Boolean multivariate mappings induced by each S-box output bit. All $t \times t$ S-boxes for the mini-versions under analysis are bijective and consist of the composition of inversion in $\mathrm{GF}(2^t)$ followed by an affine transformation, just like the AES S-box. More especifically, each output bit can be expressed as a multivariate polynomial of degree exactly $t-1$, which is the maximum possible. This choice (and the overall design of the mini-versions) was intended to preserve the cryptographic strength of the mini-versions as close as possible to what is expected of block ciphers operating on $16 \cdot t$-bit blocks.

6 Experimental Results

A 1st-order square distinguisher is depicted in (2). It covers 3.25 rounds of AES, and has the same length for all relevant word sizes, $t \geq 3$. The position of the initial 'A' word does not change the length of the distinguisher. We count every round transformation AK_i, SB, SR and MC as a 0.25 fraction of a round. Symbols on top of arrows indicate the round transformations the cipher state undergoes during encryption. Composition of mappings operate in right-to-left order.

$$\begin{pmatrix} A & P & P & P \\ P & P & P & P \\ P & P & P & P \\ P & P & P & P \end{pmatrix} \xrightarrow{MC \circ SR \circ SB \circ AK_0} \begin{pmatrix} A & P & P & P \\ A & P & P & P \\ A & P & P & P \\ A & P & P & P \end{pmatrix} \xrightarrow{SR \circ SB \circ AK_1} \begin{pmatrix} A & P & P & P \\ P & P & P & A \\ P & P & A & P \\ P & A & P & P \end{pmatrix}$$

$$\xrightarrow{SR \circ SB \circ AK_2 \circ MC} \begin{pmatrix} A & A & A & A \\ A & A & A & A \\ A & A & A & A \\ A & A & A & A \end{pmatrix} \xrightarrow{AK_3 \circ MC} \begin{pmatrix} B & B & B & B \\ B & B & B & B \\ B & B & B & B \\ B & B & B & B \end{pmatrix} \xrightarrow{SB} \begin{pmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{pmatrix} \quad (2)$$

Distinguisher (2) has originally been described in [6], and the reasoning for the propagation of 'A', 'P', 'B' and '?' patterns is based on the structure of each pattern and how they change according to each round transformation.

Another reasoning to explain why (2) reaches only 3.25 rounds and not further uses higher-order derivatives. If we consider that a Λ-set contains $2^{t \cdot o}$ CP, where o is the order of the square attack then, one can view this Λ-set as a $t \cdot o$-th derivative applied to the encryption mapping of a t-bit oriented cipher, with distinct points of derivation. The degree of each output bit after every round increases as a power of $t-1$, because of the SB layer. Thus, after the first round, all output bits can be represented as multivariate functions of degree $t-1$. After two rounds, the output bits can be viewed as multivariate functions of degree $(t-1)^2$, because of the second (nested) S-box layer, SB. And so on, up to a maximum degree of $16 \cdot t$ because there are at most $16 \cdot t$ plaintext bits[3].

[3] The ANF (algebraic normal form) of any monomial can involve at most $16 \cdot t$ distinct variables.

According to Lemma 1, a Λ-set corresponds to a $t \cdot o$-th order derivative. If the derivative has order higher than the degree of the output mappings of the mini-version (after a certain number of rounds), then the exclusive-or sum, or the derivative of the mapping, vanishes.

But, there is a twist: if one starts counting the degree of the multivariate polynomials from the first SB layer, then there is no agreement with the theory above. For instance, (2) covers 3.25 rounds, which means three SB layers, that is, the degree of each output bit is $3^3 = 27$, but (2) uses only 2^4 CP or a 4th-order derivative, which is not enough for a multivariate polynomial of degree 27 to vanish. But, if we discard the first two rounds of the distinguisher, then there is agreement between theory and practice. Why? Notice that the initial 'A' pattern is 'preserved' across the first two SB layers. If we discard these two S-box layers and start counting the degree from the third SB layer, then the output bits will have degree $t - 1 = 3$ which vanish for a 4th-order derivative. For the next SB layer, the polynomials will have degree $(t - 1)^2 = 9$ which do not vanish for a 4th-order derivative. Thus, the distinguisher cannot cross the fourth SB layer.

The reason to discard the first two S-box layers follows similarly reasoning to why it makes no sense to use 16th-order Λ-sets: since the encryption function is a $16 \cdot t$-bit permutation mapping, its output will always be a huge ($16 \cdot t$-bit) active word, independent of t and for any $16 \cdot t$-bit block cipher.

Taking this *ad hoc* reasoning into account, the theoretical predictions are corroborated by attack experiments on mini-AES versions and even AES and Rijndael. Without the concept of higher-order derivatives, the usual explanation of pattern propagation across higher-order square distinguishers is much harder to justify. Thus, the derivatives (ignoring the first two rounds) provides a more consistent explanation for higher-order distinguishers, not only for mini-versions but also for the original AES.

In general, for an $t \cdot o$-th order Λ-set, containing $2^{t \cdot o}$ CP, a o-th order square distinguisher is expected to reach r rounds (or to cross r SB layers) as long as the following relation is satisfied

$$\min(16 \cdot t, (t - 1)^{r-2}) < t \cdot o. \tag{3}$$

It means that, ignoring the first two rounds $(r-2)$, the degree of the multivariate polynomial of each output bit of every round increases as a power of $t - 1$ per S-box layer, until a maximum of $16 \cdot t$, because there are at most $16 \cdot t$ plaintext bits. A $t \cdot o$-th derivative can vanish polynomials with degree at most $t \cdot o - 1$, according to Lemma1, setting a threshold to the length of the distinguisher.

As a test, the next relevant higher-order distinguisher has order four because of MC round transformation that combines four words at a time through an MDS matrix. The corresponding distinguisher is depicted in (4), which starts with a $4 \cdot t$-bit active word split into four pieces denoted A^*. Each of the four t-bit words of this active multi-word can be seen as 'E' words (denoted E_1, ..., E_4). The position of the initial four A^* is not random. The four words are positioned so as to account for the forthcoming SR transformation and the subsequent MC layer. Thus, after $SR \circ SB \circ AK_1 \circ MC \circ SR \circ SB \circ AK_0$ or 1.75 rounds, the state

still contains one $4 \cdot t$-bit active word, while the remaining words are passive. Therefore, the careful choice of the positioning of A^* helped extend this pattern across almost two rounds.

Again, computing the degree of Boolean expressions of S-box output bits after the second SB layer, explains why (4) reaches 4.25 rounds and not further. Recal that we use a 16th-order derivative, because it is a 4th-order attack and $t = 4$. After the third S-box layer, the degree of the polynomial is 3, which vanishes for a 16-th order derivative. The same happens after the fourth SB layer (the degree becomes 9), but not after the fifth SB layer (the degree becomes 27). That is why this distinguisher cannot reach beyond 4.25 rounds.

$$
\begin{pmatrix} A^* & P & P & P \\ P & A^* & P & P \\ P & P & A^* & P \\ P & P & P & A^* \end{pmatrix} = \begin{pmatrix} E_1 & P & P & P \\ P & E_2 & P & P \\ P & P & E_3 & P \\ P & P & P & E_4 \end{pmatrix} \overset{SR \circ SB \circ AK_0}{\rightarrow} \begin{pmatrix} A^* & P & P & P \\ A^* & P & P & P \\ A^* & P & P & P \\ A^* & P & P & P \end{pmatrix}
$$

$$
\overset{SR \circ SB \circ AK_1 \circ MC}{\rightarrow} \begin{pmatrix} A^* & P & P & P \\ P & P & P & A^* \\ P & P & A^* & P \\ P & A^* & P & P \end{pmatrix} \overset{MC}{\rightarrow} \begin{pmatrix} E_1 & E_2 & E_3 & E_4 \\ E_1 & E_2 & E_3 & E_4 \\ E_1 & E_2 & E_3 & E_4 \\ E_1 & E_2 & E_3 & E_4 \end{pmatrix} \overset{SR \circ SB \circ AK_2}{\rightarrow}
$$

$$
\begin{pmatrix} E_1 & E_2 & E_3 & E_4 \\ E_2 & E_3 & E_4 & E_1 \\ E_3 & E_4 & E_1 & E_2 \\ E_4 & E_1 & E_2 & E_3 \end{pmatrix} = \begin{pmatrix} A_1^* & A_2^* & A_3^* & A_4^* \\ A_1^* & A_2^* & A_3^* & A_4^* \\ A_1^* & A_2^* & A_3^* & A_4^* \\ A_1^* & A_2^* & A_3^* & A_4^* \end{pmatrix} \overset{SR \circ SB \circ AK_3 \circ MC}{\rightarrow} \begin{pmatrix} A_1^* & A_2^* & A_3^* & A_4^* \\ A_2^* & A_3^* & A_4^* & A_1^* \\ A_3^* & A_4^* & A_1^* & A_2^* \\ A_4^* & A_1^* & A_2^* & A_3^* \end{pmatrix}
$$

$$
\overset{AK_4 \circ MC}{\rightarrow} \begin{pmatrix} B & B & B & B \\ B & B & B & B \\ B & B & B & B \\ B & B & B & B \end{pmatrix} \overset{SB}{\rightarrow} \begin{pmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{pmatrix} \tag{4}
$$

Unlike (2), the distinguisher (4) is one round longer. An exception is the case $t = 3$ which is depicted in (5).

$$\begin{pmatrix} A^* & P & P & P \\ P & A^* & P & P \\ P & P & A^* & P \\ P & P & P & A^* \end{pmatrix} = \begin{pmatrix} E_1 & P & P & P \\ P & E_2 & P & P \\ P & P & E_3 & P \\ P & P & P & E_4 \end{pmatrix} \overset{SR\circ SB\circ AK_0}{\rightarrow} \begin{pmatrix} A^* & P & P & P \\ A^* & P & P & P \\ A^* & P & P & P \\ A^* & P & P & P \end{pmatrix}$$

$$\overset{SR\circ SB\circ AK_1\circ MC}{\rightarrow} \begin{pmatrix} A^* & P & P & P \\ P & P & P & A^* \\ P & P & A^* & P \\ P & A^* & P & P \end{pmatrix} \overset{MC}{\rightarrow} \begin{pmatrix} A_1^* & A_2^* & A_3^* & A_4^* \\ A_1^* & A_2^* & A_3^* & A_4^* \\ A_1^* & A_2^* & A_3^* & A_4^* \\ A_1^* & A_2^* & A_3^* & A_4^* \end{pmatrix} \overset{SR\circ SB\circ AK_2}{\rightarrow}$$

$$\begin{pmatrix} A_1^* & A_2^* & A_3^* & A_4^* \\ A_2^* & A_3^* & A_4^* & A_1^* \\ A_3^* & A_4^* & A_1^* & A_2^* \\ A_4^* & A_1^* & A_2^* & A_3^* \end{pmatrix} \overset{MC}{\rightarrow} \begin{pmatrix} A^* & A^* & A^* & A^* \\ A^* & A^* & A^* & A^* \\ A^* & A^* & A^* & A^* \\ A^* & A^* & A^* & A^* \end{pmatrix} \overset{SR\circ SB\circ AK_3}{\rightarrow} \begin{pmatrix} A_1 & A_2 & A_3 & A_4 \\ A_2 & A_3 & A_4 & A_1 \\ A_3 & A_4 & A_1 & A_2 \\ A_4 & A_1 & A_2 & A_3 \end{pmatrix} \overset{SB\circ AK_4\circ MC}{\rightarrow}$$

$$\begin{pmatrix} E_1 & E_2 & E_3 & E_4 \\ E_1 & E_2 & E_3 & E_4 \\ E_1 & E_2 & E_3 & E_4 \\ E_1 & E_2 & E_3 & E_4 \end{pmatrix} \overset{SR}{\rightarrow} \begin{pmatrix} E_1 & E_2 & E_3 & E_4 \\ E_2 & E_3 & E_4 & E_1 \\ E_3 & E_4 & E_1 & E_2 \\ E_4 & E_1 & E_2 & E_3 \end{pmatrix} \overset{AK_5\circ MC}{\rightarrow} \begin{pmatrix} B & B & B & B \\ B & B & B & B \\ B & B & B & B \\ B & B & B & B \end{pmatrix} \overset{SB}{\rightarrow} \begin{pmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{pmatrix} \quad (5)$$

For (5), the argument of higher-order derivatives matches exactly the lengths obtained experimentally. At the third round, the Boolean expressions of the S-boxes have degree 2, which vanish for an 12th-order derivative (4th-order order Λ-set). After the fourth S-box layer, the degree reaches 4; after the fifth SB layer, the degree becomes 8. In all and these cases the polynomials vanish. But, for the sixth SB layer, the degree becomes 16 which larger than 12. Thus, (5) reaches only 5.25 rounds.

An 8th-order square distinguisher for $t = 4$ is depicted in (6).

$$\begin{pmatrix} A^* & A^* & P & P \\ P & A^* & A^* & P \\ P & P & A^* & A^* \\ A^* & P & P & A^* \end{pmatrix} = \begin{pmatrix} E_1 & E_2 & P & P \\ P & E_3 & E_4 & P \\ P & P & E_5 & E_6 \\ E_8 & P & P & E_7 \end{pmatrix} \overset{SR\circ SB\circ AK_0}{\rightarrow} \begin{pmatrix} A^* & A^* & P & P \\ A^* & A^* & P & P \\ A^* & A^* & P & P \\ A^* & A^* & P & P \end{pmatrix}$$

$$\overset{SR\circ SB\circ AK_1\circ MC}{\rightarrow} \begin{pmatrix} A^* & A^* & P & P \\ A^* & P & P & A^* \\ P & P & A^* & A^* \\ P & A^* & A^* & P \end{pmatrix} \overset{SB\circ AK_2\circ MC}{\rightarrow} \begin{pmatrix} E_1^* & E_2^* & E_3^* & E_4^* \\ E_1^* & E_2^* & E_3^* & E_4^* \\ E_1^* & E_2^* & E_3^* & E_4^* \\ E_1^* & E_2^* & E_3^* & E_4^* \end{pmatrix} \overset{SR}{\rightarrow}$$

$$\begin{pmatrix} E_1^* & E_2^* & E_3^* & E_4^* \\ E_2^* & E_3^* & E_4^* & E_1^* \\ E_3^* & E_4^* & E_1^* & E_2^* \\ E_4^* & E_1^* & E_2^* & E_3^* \end{pmatrix} \overset{SR\circ SB\circ AK_3\circ MC}{\rightarrow} \begin{pmatrix} E & E & E & E \\ E & E & E & E \\ E & E & E & E \\ E & E & E & E \end{pmatrix} \overset{AK_4\circ MC}{\rightarrow}$$

$$\begin{pmatrix} B & B & B & B \\ B & B & B & B \\ B & B & B & B \\ B & B & B & B \end{pmatrix} \overset{SB}{\rightarrow} \begin{pmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{pmatrix} \quad (6)$$

An 8th-order distinguisher for $t = 3$ is depicted in (7) and covers 6.25 rounds.

$$\begin{pmatrix} A^* & A^* & P & P \\ P & A^* & A^* & P \\ P & P & A^* & A^* \\ A^* & P & P & A^* \end{pmatrix} \overset{SR\circ SB\circ AK_0}{\rightarrow} \begin{pmatrix} A^* & A^* & P & P \\ A^* & A^* & P & P \\ A^* & A^* & P & P \\ A^* & A^* & P & P \end{pmatrix} \overset{SR\circ SB\circ AK_1\circ MC}{\rightarrow} \begin{pmatrix} A^* & A^* & P & P \\ A^* & P & P & A^* \\ P & P & A^* & A^* \\ P & A^* & A^* & P \end{pmatrix}$$
$$\overset{SB\circ AK_2\circ MC}{\rightarrow} \begin{pmatrix} E_1^* & E_2^* & E_3^* & E_4^* \\ E_1^* & E_2^* & E_3^* & E_4^* \\ E_1^* & E_2^* & E_3^* & E_4^* \\ E_1^* & E_2^* & E_3^* & E_4^* \end{pmatrix} \overset{SR}{\rightarrow} \begin{pmatrix} E_1^* & E_2^* & E_3^* & E_4^* \\ E_2^* & E_3^* & E_4^* & E_1^* \\ E_3^* & E_4^* & E_1^* & E_2^* \\ E_4^* & E_1^* & E_2^* & E_3^* \end{pmatrix} \overset{SR\circ SB\circ AK_3\circ MC}{\rightarrow}$$
$$\begin{pmatrix} A_1 & A_1 & A_2 & A_2 \\ A_1 & A_2 & A_2 & A_1 \\ A_2 & A_2 & A_1 & A_1 \\ A_2 & A_1 & A_1 & A_2 \end{pmatrix} \overset{SR\circ SB\circ AK_5\circ MC\circ SR\circ SB\circ AK_4\circ MC}{\rightarrow} \begin{pmatrix} E & E & E & E \\ E & E & E & E \\ E & E & E & E \\ E & E & E & E \end{pmatrix} \overset{AK_6\circ MC}{\rightarrow}$$
$$\begin{pmatrix} B & B & B & B \\ B & B & B & B \\ B & B & B & B \\ B & B & B & B \end{pmatrix} \overset{SB}{\rightarrow} \begin{pmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{pmatrix} \tag{7}$$

Finally, the longer square distinguisher we ever found is depicted in (8). It is a 12th-order square distinguisher for $t = 3$ and covers 7.25 rounds.

$$\begin{pmatrix} A^* & A^* & A^* & P \\ P & A^* & A^* & A^* \\ A^* & P & A^* & A^* \\ A^* & A^* & P & A^* \end{pmatrix} \overset{SR\circ SB\circ AK_0}{\rightarrow} \begin{pmatrix} A^* & A^* & A^* & P \\ A^* & A^* & A^* & P \\ A^* & A^* & A^* & P \\ A^* & A^* & A^* & P \end{pmatrix} \overset{SR\circ SB\circ AK_1\circ MC}{\rightarrow}$$
$$\begin{pmatrix} A^* & A^* & A^* & P \\ A^* & A^* & P & A^* \\ A^* & P & A^* & A^* \\ P & A^* & A^* & A^* \end{pmatrix} \overset{SB\circ AK_2\circ MC}{\rightarrow} \begin{pmatrix} E_1^* & E_2^* & E_3^* & E_4^* \\ E_1^* & E_2^* & E_3^* & E_4^* \\ E_1^* & E_2^* & E_3^* & E_4^* \\ E_1^* & E_2^* & E_3^* & E_4^* \end{pmatrix} \overset{SR}{\rightarrow} \begin{pmatrix} E_1^* & E_2^* & E_3^* & E_4^* \\ E_2^* & E_3^* & E_4^* & E_1^* \\ E_3^* & E_4^* & E_1^* & E_2^* \\ E_4^* & E_1^* & E_2^* & E_3^* \end{pmatrix}$$
$$\overset{SB\circ AK_3\circ MC}{\rightarrow} \begin{pmatrix} A_1^* & A_2^* & A_3^* & A_4^* \\ A_1^* & A_2^* & A_3^* & A_4^* \\ A_1^* & A_2^* & A_3^* & A_4^* \\ A_1^* & A_2^* & A_3^* & A_4^* \end{pmatrix} \overset{SR\circ SB\circ AK_5\circ MC\circ SR\circ SB\circ AK_4\circ MC\circ SR}{\rightarrow}$$
$$\begin{pmatrix} E & E & E & E \\ E & E & E & E \\ E & E & E & E \\ E & E & E & E \end{pmatrix} \overset{AK_7\circ MC\circ SR\circ SB\circ AK_6\circ MC}{\rightarrow} \begin{pmatrix} B & B & B & B \\ B & B & B & B \\ B & B & B & B \\ B & B & B & B \end{pmatrix} \overset{SB}{\rightarrow} \begin{pmatrix} ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \\ ? & ? & ? & ? \end{pmatrix} \tag{8}$$

7 Summary and Open Problems

Table 2 presents a summary of the square distinguishers for mini-AES versions and AES itself. The figures with '*' indicate the predicted length of distinguishers based on higher-order derivatives.

Table 2. Number of rounds of square distinguishers for t-bit word AES (*: predicted).

t (bits)	Λ-set Order			
	1	4	8	12
3	3.25	5.25	6.25	7.25
4	3.25	4.25	5.25	5.25*
5	3.25	4.25	4.25	4.25*
6	3.25	4.25	4.25*	4.25*
7	3.25	4.25	4.25*	4.25*
8	3.25	4.25	4.25*	4.25*

From Table 2, we conclude that the higher-order square attack (beyond the 4th order) is not effective against AES. Moreover, the square distinguishers seems to behave differently for mini-versions and for the original AES. This asymmetry is due to the decreasing nonlinear degree in the scaled-down S-boxes.

The conclusions for AES cannot be immediately extended to Rijndael with blocks larger than 128 bits because the state matrix of the latter is $4 \times n$, with $n > 4$ and thus, diffusion is not as uniform as in AES. Experiments indicate that the size of distinguishers is larger than predicted by higher-order derivatives due to non-uniform diffusion because of the *ad hoc* ShiftRows offsets. Moreover, some bytes of the distinguisher may still retain some balanced bytes, while in AES, not residual balanced bytes remained.

An interesting open problem is to apply the higher-order derivative approach to general SPN schemes such as SASAS and SASASAS [2], in which the S-boxes (and linear components) are key-dependent, but may have (potentially) small nonlinear degree.

It is an open problem whether other tecniques can be combined with square attacks to exploit further rounds of word-oriented ciphers. For example, in the cases where the degree of the Boolean multivariate expressions do not vanish, there is a residual degree that may still be small enough for cube attacks [8]. For example, the degree of the Boolean expression at the end of (5) in a 4th-order Λ-set is only $16 - 12 = 4$.

Acknowledgement

Many thanks to Henri Gilbert for discussing the effect of active words in the first rounds of higher-order square distinguishers.

References

1. AES, *Advanced Encryption Standard (AES)*, FIPS PUB 197, Federal Information Processing Standard Publication 197, U.S. Department of Commerce, Nov, 2001.
2. A. Biryukov, A. Shamir, *Structural Cryptanalysis of SASAS*, Adv. in Cryptology, Eurocrypt'01, B. Pfitzmann, Ed., Springer-Verlag, LNCS 2045, 2001, 394–405.

3. J. Borst, L.R. Knudsen, V. Rijmen, *Two Attacks on Reduced IDEA*, Adv. in Cryptology, Eurocrypt'97, W. Fumy,Ed., Springer-Verlag, LNCS 1233, 1997, 1–13.
4. C. Cid, S. Murphy, M. Robshaw, *Small Scale Variants of the AES*, Fast Software Encryption, H. Gilbert, H. Handschuh, Eds., Springer-Verlag, LNCS 3557, 2005, 145–162.
5. N. Courtois, http://www.cryptosystem.net/aes/
6. J. Daemen, L.R. Knudsen, V. Rijmen, *The Block Cipher SQUARE*, Fast Software Encryption, E. Biham,Ed., Springer-Verlag , LNCS 1267, 1997, 149–165.
7. DES, *Data Encryption Standard*, FIPS PUB 46-2, Federal Information Processing Standards Publication, 1993.
8. I. Dinur, A. Shamir, *Cube Attacks on Tweakable Black Box Polynomials*, Cryptology ePrint Archive: report 2008/385.
9. T. Fuhr, T. Peyrin, *Cryptanalysis of RadioGatun*, Cryptology ePrint Archive, report 2008/515.
10. Y. Hu, Y. Zhang, G. Xiao, *Integral Cryptanalysis of SAFER+*, Electronic Letters, (35):17, Aug. 1999, 1458–1459.
11. L.R. Knudsen, *Truncated and Higher Order Differentials*, Fast Software Encryption, B. Preneel,Ed., Springer-Verlag, LNCS 1008, 1995, 196–211.
12. L.R. Knudsen, W. Meier, B. Preneel, V. Rijmen, S. Verdoolaege, *Analysis methods for (alleged) RC4*, Adv. in Cryptology - ASIACRYPT'98, K. Ohta and D. Pei, Eds., Springer-Verlag, LNCS 1514, 327–341.
13. L.R. Knudsen, D. Wagner, *Integral Cryptanalysis*, Fast Software Encryption, J. Daemen and V. Rijmen, Eds., Springer-Verlag, LNCS 2365, 2002, 112–127.
14. X. Lai, *Higher order derivatives and differential cryptanalysis*, Communications and Cryptography: Two Sides of One Tapestry, R.E. Blahut *et al.*, Eds., Kluwer Academic Publishers, 1994, 227–233.
15. X. Lai, *On the Design and Security of Block Ciphers*, ETH Series in Information Processing, J.L. Massey,ed., vol. 1, 1995, Hartung-Gorre Verlag, Konstanz.
16. S. Lucks, *The Saturation Attack – a Bait for Twofish*, Fast Software Encryption, M. Matsui, Ed., Springer-Verlag, LNCS 2355, 2001, 1–15.
17. A.J. Menezes, P.C. van Oorschot, S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997.
18. M.A. Mohammad, E. Schaefer, S. Wedig, *A simplified AES algorithm and its linear and differential cryptanalyses*, Cryptologia, Apr. 2003.
19. R.C-.W. Phan, *Mini advanced encryption standard (mini-AES): A testbed for cryptanalysis students*, Cryptologia (26):4, Oct. 2002.
20. G. Piret, J.-J. Quisquater, *Integral Cryptanalysis on Reduced-round Safer++: A way to extend the attack?*, *NESSIE Public Report, NES/DOC/UCL/WP5/002/1*, 2003.
21. R.L. Rivest, *The RC5 Encryption Algorithm*, Fast Software Encryption, B. Preneel, Ed., Springer-Verlag, LNCS 1008, 1995, 86–96.
22. R.L. Rivest, M.J.B. Robshaw, R. Sidney, Y.L. Yin, *The RC6 Block Cipher*, 1st AES Conference, 1998, http://csrc.nist.gov/encryption/aes/
23. R.-P. Weinmann, *Evaluating Algebraic Attacks on the AES*, diploma thesis, Technische Universität Darmstadt.

A Mini-AES S-boxes

The S-boxes used in our experiments are listed below for word sizes $t \in \{3, 4, 5, 6, 7\}$. The S-box entries are listed in sequential order with the leftmost entry

corresponding to $S[0]$, and the rightmost entry to $S[2^t - 1]$. In order to keep the designs as similar as possible to the original AES, the S-boxes are permutations designed as the composition of the inversion mapping in GF(2^t) followed by an affine transformation.

For $\mathbf{t = 3}$: $S = [7, 2, 5, 1, 0, 6, 4, 3]$.

For $\mathbf{t = 4}$: $S = [13, 10, 1, 5, 12, 15, 9, 14, 2, 3, 11, 7, 8, 0, 4, 6]$.

For $\mathbf{t = 5}$: $S = [17, 14, 31, 29, 16, 21, 19, 3, 6, 26, 23, 28, 18, 15,$ 10, 13, 27, 24, 1, 7, 20, 0, 4, 30, 5, 9, 8, 12, 25, 22, 11, 2].

For $\mathbf{t = 6}$: $S = [49, 14, 7, 45, 28, 5, 27, 61, 60, 53, 47, 15, 54, 3,$ 19, 48, 12, 63, 23, 2, 40, 36, 24, 39, 59, 33, 58, 13, 50, 35, 46, 11, 52, 31, 32, 17, 20, 56, 37, 42, 34, 38, 0, 1, 26, 8, 44, 16, 6, 4, 57, 30, 25, 41, 43, 18, 29, 22, 10, 9, 55, 21, 62, 51].

For $\mathbf{t = 7}$: $S = [49, 78, 127, 61, 64, 8, 115, 109, 110, 5, 74, 58, 2, 6,$ 27, 7, 111, 102, 47, 120, 121, 117, 1, 56, 93, 112, 91, 96, 54, 17, 124, 94, 72, 77, 43, 59, 40, 12, 50, 81, 21, 29, 87, 18, 41, 98, 82, 92, 3, 32, 118, 37, 86, 106, 126, 88, 67, 23, 33, 95, 52, 97, 119, 13, 42, 24, 11, 68, 46, 55, 38, 60, 90, 123, 76, 22, 69, 19, 65, 25, 39, 71, 99, 31, 20, 0, 85, 14, 125, 10, 45, 48, 53, 108, 36, 28, 122, 73, 30, 101, 35, 79, 63, 114, 51, 44, 105, 83, 103, 113, 34, 70, 26, 75, 116, 84, 57, 15, 80, 100, 16, 62, 89, 104, 4, 9, 107, 66].