IET Information Security

Research Article

IET Journals
The Institution of Engineering and Technology

Base for algebraic cryptanalysis based on combined representation of S-box

ISSN 1751-8709
Received on 19th February 2018
Revised 20th October 2018
Accepted on 30th November 2018
E-First on 20th February 2019
doi: 10.1049/iet-ifs.2018.5051
www.ietdl.org

Bekir Unlu[1] ✉

[1]*Cryptography Researcher, Belgium*

✉ *E-mail: bekir.unlu@hotmail.be*

Abstract: This study reveals an organised algebraic view on the Advanced Encryption Standard (AES), and in particular the S-box. A clear visualisation of algebraic expressions is granted for both the cipher part and the key schedule part of AES. Additionally, by introducing the combined representation of the S-box, alternative methods can be constructed to serve algebraic cryptanalysis on AES. The process to obtain a combined representation of the S-box, as a combination of four bits, the input value's multiplicative inverse and bytes in hexadecimal notion, may be of high importance for other proposed ciphers too built with layers of S-boxes.

1 Introduction

The non-linearity in Advanced Encryption Standard (AES) is originated by the use of a non-linear S-box and therefor the algebraic expression of the S-box is essential for the complexity of an algebraic or an interpolation attack on AES. The lower degree of the S-box expression, the lower the degree of AES equations, the higher the chances for attacks. Daemen and Rijmen [1] granted the S-box's algebraic expression containing nine terms with extremely high powers. By using this S-box's algebraic expression Ferguson *et al.* [2] succeeded to derive one closed algebraic formula describing the AES cipher as a system of continued fractions, containing 2^{50} terms for a 128-bit AES, but infeasible to be solved. On the other hand, by utilising the relationship between the S-box's outcome and the input's multiplicative inverse, a system of multivariate quadratic equations over GF(2) can be derived. Analysis on solving an overdefined system of multivariate equations through linearisation and relinearisation led to several algorithms to attack AES. Courtois and Pieprzyk [3] proposed the eXtended Spare Linearisation algorithm and made use of the sparsity of the S-box's quadratic equations, consisting of 24 bi-affine or in total 40 full quadratic equations.

It is a fact that AES cryptanalysis can only benefit and improve more from a compact S-box expression with less algebraic complexity, which can produce more equations. A compact expression may even initiate different methods for AES cryptanalysis. This paper is presenting a compact S-box expression, proposing and exploring two methods for cryptanalysis, namely the Byte Multiplication or FD Multiplication and the Field Trace of AES, in which FD is referring to the Galois Field element 'FD'.

Unfortunately, there has been no algebraic enhancements or alternatives produced on S-box's algebraic expression with high degrees. This is also one of the reasons why the AES has not been visualised as ONE and CLOSED system of pure algebraic equations to be solved with algebraic means or pure algebraic cryptanalysis.

A system constructed with GF(2^8) variables and where the S-box is positioned in its context, is getting more interesting and effective if one can provide a more compact algebraic alternative for the S-box. An alternative algebraic expression, named as combined representation, is provided in this paper in which the S-box is represented by a multiplicative inverse and 4 bits.

Illustrated with three rounds, the paper starts with visualising a 128-bit AES encryption as a system of algebraic equations, constructed with S-boxes and newly defined variables, divided into the cipher part and key schedule part variables, without indicating any algebraic expression for the S-box for the time being. Replacing the S-box by the combined representation, enables the AES system of algebraic equations to serve alternative methods for algebraic cryptanalysis.

2 System of algebraic equations

The 128-bit AES is constructed through operations on 128-bits long states with a non-linear S-box, linear transformations and a round key addition. All these operations, except the S-box, can be translated into algebraic expressions with the use of the base unit byte and newly defined round byte-variables in the cipher part as well as in the key schedule part. Hereafter the algebraic equations for three rounds AES are constructed.

2.1 System variables

We define the bytes of AES cipher input 'A_i', the AES key value 'a_i' and the AES cipher output 'Z_i' with $A_i, a_i, Z_i \in \mathrm{GF}(2^8)$ for $i = 0,1,2,3,\ldots,15$. On the other hand, the bytes of cipher round-variables 'p_i', 'q_i' and 'r_i', together with key addition round-variables 'b_i', 'c_i' and 'd_i' with $p_i, q_i, r_i, b_i, c_i, d_i \in \mathrm{GF}(2^8)$ for $i = 0,1,2,3,\ldots,15$ are sufficient to achieve the AES system of algebraic equations for three rounds:

Round 0 – cipher part

$$A_i + a_i = p_i \quad \text{for} i = 0, 1, 2, \ldots, 15$$

Round 1 – cipher part

$$\begin{aligned}
&\texttt{`}02' S(p_0) + \texttt{`}03' S(p_5) + S(p_{10}) + S(p_{15}) + b_0 = q_0\\
&S(p_0) + \texttt{`}02' S(p_5) + \texttt{`}03' S(p_{10}) + S(p_{15}) + b_1 = q_1\\
&S(p_0) + S(p_5) + \texttt{`}02' S(p_{10}) + \texttt{`}03' S(p_{15}) + b_2 = q_2\\
&\texttt{`}03' S(p_0) + S(p_5) + S(p_{10}) + \texttt{`}02' S(p_{15}) + b_3 = q_3
\end{aligned}$$

$$\begin{aligned}
&\texttt{`}02' S(p_4) + \texttt{`}03' S(p_9) + S(p_{14}) + S(p_3) + b_4 = q_4\\
&S(p_4) + \texttt{`}02' S(p_9) + \texttt{`}03' S(p_{14}) + S(p_3) + b_5 = q_5\\
&S(p_4) + S(p_9) + \texttt{`}02' S(p_{14}) + \texttt{`}03' S(p_3) + b_6 = q_6\\
&\texttt{`}03' S(p_4) + S(p_9) + S(p_{14}) + \texttt{`}02' S(p_3) + b_7 = q_7
\end{aligned}$$

$$\begin{aligned}
&\text{`}02'S(p_8) + \text{`}03'S(p_{13}) + S(p_2) + S(p_7) + b_8 = q_8\\
&S(p_8) + \text{`}02'S(p_{13}) + \text{`}03'S(p_2) + S(p_7) + b_9 = q_9\\
&S(p_8) + S(p_{13}) + \text{`}02'S(p_2) + \text{`}03'S(p_7) + b_{10} = q_{10}\\
&\text{`}03'S(p_8) + S(p_{13}) + S(p_2) + \text{`}02'S(p_7) + b_{11} = q_{11}
\end{aligned}$$

$$\begin{aligned}
&\text{`}02'S(p_{12}) + \text{`}03'S(p_1) + S(p_6) + S(p_{11}) + b_{12} = q_{12}\\
&S(p_{12}) + \text{`}02'S(p_1) + \text{`}03'S(p_6) + S(p_{11}) + b_{13} = q_{13}\\
&S(p_{12}) + S(p_1) + \text{`}02'S(p_6) + \text{`}03'S(p_{11}) + b_{14} = q_{14}\\
&\text{`}03'S(p_{12}) + S(p_1) + S(p_6) + \text{`}02'S(p_{11}) + b_{15} = q_{15}
\end{aligned}$$

Round 2 – cipher part

$$\begin{aligned}
&\text{`}02'S(q_0) + \text{`}03'S(q_5) + S(q_{10}) + S(q_{15}) + c_0 = r_0\\
&S(q_0) + \text{`}02'S(q_5) + \text{`}03'S(q_{10}) + S(q_{15}) + c_1 = r_1\\
&S(q_0) + S(q_5) + \text{`}02'S(q_{10}) + \text{`}03'S(q_{15}) + c_2 = r_2\\
&\text{`}03'S(q_0) + S(q_5) + S(q_{10}) + \text{`}02'S(q_{15}) + c_3 = r_3
\end{aligned}$$

$$\begin{aligned}
&\text{`}02'S(q_4) + \text{`}03'S(q_9) + S(q_{14}) + S(q_3) + c_4 = r_4\\
&S(q_4) + \text{`}02'S(q_9) + \text{`}03'S(q_{14}) + S(q_3) + c_5 = r_5\\
&S(q_4) + S(q_9) + \text{`}02'S(q_{14}) + \text{`}03'S(q_3) + c_6 = r_6\\
&\text{`}03'S(q_4) + S(q_9) + S(q_{14}) + \text{`}02'S(q_3) + c_7 = r_7
\end{aligned}$$

$$\begin{aligned}
&\text{`}02'S(q_8) + \text{`}03'S(q_{13}) + S(q_2) + S(q_7) + c_8 = r_8\\
&S(q_8) + \text{`}02'S(q_{13}) + \text{`}03'S(q_2) + S(q_7) + c_9 = r_9\\
&S(q_8) + S(q_{13}) + \text{`}02'S(q_2) + \text{`}03'S(q_7) + c_{10} = r_{10}\\
&\text{`}03'S(q_8) + S(q_{13}) + S(q_2) + \text{`}02'S(q_7) + c_{11} = r_{11}
\end{aligned}$$

$$\begin{aligned}
&\text{`}02'S(q_{12}) + \text{`}03'S(q_1) + S(q_6) + S(q_{11}) + c_{12} = r_{12}\\
&S(q_{12}) + \text{`}02'S(q_1) + \text{`}03'S(q_6) + S(q_{11}) + c_{13} = r_{13}\\
&S(q_{12}) + S(q_1) + \text{`}02'S(q_6) + \text{`}03'S(q_{11}) + c_{14} = r_{14}\\
&\text{`}03'S(q_{12}) + S(q_1) + S(q_6) + \text{`}02'S(q_{11}) + c_{15} = r_{15}
\end{aligned}$$

Round 3 – cipher part

$$\begin{aligned}
&S(r_0) + d_0 = Z_0\\
&S(r_5) + d_1 = Z_1\\
&S(r_{10}) + d_2 = Z_2\\
&S(r_{15}) + d_3 = Z_3\\
&S(r_4) + d_4 = Z_4\\
&S(r_9) + d_5 = Z_5\\
&S(r_{14}) + d_6 = Z_6\\
&S(r_3) + d_7 = Z_7\\
&S(r_8) + d_8 = Z_8\\
&S(r_{13}) + d_9 = Z_9\\
&S(r_2) + d_{10} = Z_{10}\\
&S(r_7) + d_{11} = Z_{11}\\
&S(r_{12}) + d_{12} = Z_{12}\\
&S(r_1) + d_{13} = Z_{13}\\
&S(r_6) + d_{14} = Z_{14}\\
&S(r_{11}) + d_{15} = Z_{15}
\end{aligned}$$

Round 1 – key schedule part

$$\begin{aligned}
&b_0 = S(a_{13}) + a_0 + 01\\
&b_1 = S(a_{14}) + a_1\\
&b_2 = S(a_{15}) + a_2\\
&b_3 = S(a_{12}) + a_3\\
&b_i = b_{i-4} + a_i \quad \text{for} i = 4, 5, 6, \ldots, 15
\end{aligned}$$

Round 2 – key schedule part

$$\begin{aligned}
&c_0 = S(b_{13}) + b_0 + 02\\
&c_1 = S(b_{14}) + b_1\\
&c_2 = S(b_{15}) + b_2\\
&c_3 = S(b_{12}) + b_3\\
&c_i = c_{i-4} + b_i \quad \text{for} i = 4, 5, 6, \ldots, 15
\end{aligned}$$

Round 3 – key schedule part

$$\begin{aligned}
&d_0 = S(c_{13}) + c_0 + 04\\
&d_1 = S(c_{14}) + c_1\\
&d_2 = S(c_{15}) + c_2\\
&d_3 = S(c_{12}) + c_3\\
&d_i = d_{i-4} + c_i \quad \text{for} i = 4, 5, 6, \ldots, 15
\end{aligned}$$

2.2 Solving AES algebraic equations

By creating a system of algebraic equations in GF(2^8) for a 128-bit key AES with the same number of equations and unknowns, surely the system can be solved if the S-box could be written in the function of a multiplicative inverse and that is nearly achieved in the next chapter by revealing the combined representation of the S-box. One should remark that solving an AES system in GF(2^8) with the S-box in function of a multiplicative inverse means a huge challenge too.

The system in GF(2^8) for a 10 round 128-bit key AES contains 336 equations together with 160 cipher round-variables, 10 AES key value variables and 160 key addition round-variables, resulting in a total of 336 variables.

A 128-bit key AES consists of 160 S-boxes in the cipher part and 40 in the key schedule part, a total of 200 S-boxes. Besides the S-box's representation enhancement presented in the next section, the context of the S-box within the algebraic equations is an add-on opportunity for further improvements of the AES system of algebraic equations. In other words, external operations on the equation(s) by means of multiplication in GF(2^8) or taking the Field Trace for example, will offer an add-on flexibility for the new S-box's representation enhancement, which is a nearly function of a multiplicative inverse.

3 S-box combined representation

The S-box can be written in a nearly multiplicative inverse function by rewriting the Affine Transformation (AT) workout (see the Appendix) on a different manner. The name combined representation follows from the combination of the multiplicative inverse, bits and bytes in the formula. Given X the input of the S-box and Y the multiplicative inverse of X,

$$\begin{aligned}
Y = \frac{1}{X} = {} & y_0 + y_1 t + y_2 t^2 + y_3 t^3 + y_4 t^4 + y_5 t^5\\
& + y_6 t^6 + y_7 t^7
\end{aligned}$$

then the S-box combined representation is presented as follows:

$$\begin{aligned}
\boldsymbol{S}(\boldsymbol{X}) = \boldsymbol{Z} = {} & \frac{\text{`}\mathbf{1F}'}{\boldsymbol{X}} + \mathbf{96} \cdot \boldsymbol{y}_7 + \mathbf{46} \cdot \boldsymbol{y}_6 + \mathbf{2E} \cdot \boldsymbol{y}_5\\
& + \mathbf{1A} \cdot \boldsymbol{y}_4 + \text{`}\mathbf{63}'
\end{aligned}$$

with '1***F***', '96', '46', '2*E*', '1*A*', '63' bytes in hexadecimal notion, $y_7, y_6, y_5, y_4 \in GF(2)$.

A combined representation of the *Inverse S-box* can be obtained too with the similar methodology as illustrated in the Appendix for the 'forward' S-box, by rewriting the Inverse AT with Z as input value and X the outcome of the Inverse S-box: (see equation below) with '*C*1', '8*B*', '0*D*', '1*A*', '34', '72', '05' bytes in hexadecimal notion, $z_0, z_1, z_5, z_6, z_7 \in GF(2)$.

3.1 S-box quadratic equations

Given the relationship in GF(2) between S-box's output Z and the input's multiplicative inverse Y

$$y_7 = z_6 + z_4 + z_1$$

$$y_6 = z_5 + z_3 + z_0$$

$$y_5 = z_7 + z_4 + z_2$$

$$y_4 = z_6 + z_3 + z_1$$

$$y_3 = z_5 + z_2 + z_0$$

$$y_2 = z_7 + z_4 + z_1 + 1$$

$$y_1 = z_6 + z_3 + z_0$$

$$y_0 = z_7 + z_5 + z_2 + 1$$

together with

$$X \cdot Y = 1$$

$$X^2 \cdot Y = X$$

$$X^{128} \cdot Y = Y^{128}$$

$$X^4 \cdot Y = X^3$$

and defining

$$D = \text{`}96' \cdot y_7 + \text{`}46' \cdot y_6 + \text{`}2E' \cdot y_5 + \text{`}1A' \cdot y_4 + \text{`}63'$$
$$Z = 1F \cdot Y + D$$

with X, Y, Z and $D \in \mathrm{GF}(2^8)$, we are able to create linearly independent *three sets of eight bi-affine equations* in GF(2) obtained through the combined S-box representation

1. $X \cdot Z = \text{`}1\mathrm{F}' \cdot \boldsymbol{X} \cdot \boldsymbol{Y} + D \cdot X = \text{`}1F' + D \cdot X$
2. $X^2 \cdot Z = \text{`}1F' \cdot \boldsymbol{X}^2 \cdot \boldsymbol{Y} + D \cdot X^2 = \text{`}1F' \cdot \boldsymbol{X} + D \cdot X^2$
3. $X^{128} \cdot Z = \text{`}1F' \cdot \boldsymbol{X}^{128} \cdot \boldsymbol{Y} + D \cdot X^{128} = \text{`}1F' \cdot \boldsymbol{Y}^{128} + D \cdot X^{128}$ and linearly independent *two sets of eight quadratic equations* in GF(2) with extra terms in the form of $x_i x_j$ and $z_i z_j$
4. $X^4 \cdot Z = 1F'' \cdot \boldsymbol{X}^4 \cdot \boldsymbol{Y} + D \cdot \boldsymbol{X}^4 = 1F'' \cdot \boldsymbol{X}^3 + D \cdot \boldsymbol{X}^4$
5. $X^{64} \cdot Z =' 1F' \cdot \boldsymbol{X}^{64} \cdot \boldsymbol{Y} + D \cdot \boldsymbol{X}^{64}$
   $X^{128} \cdot Z^2 = 4E \cdot \boldsymbol{X}^{128} \cdot \boldsymbol{Y}^2 + D^2 \cdot \boldsymbol{X}^{128}$ (squaring)
   $X^{128} \cdot Z^2 = 4E \cdot \boldsymbol{Y}^{128} \cdot \boldsymbol{Y} + D^2 \cdot \boldsymbol{X}^{128}$

In the first instance, the new representation $Z = 1F \cdot Y + D$ looks like to be a candidate to produce extra eight equations more but the outcome is linear dependent with the batch three bi-affine equations. The eight other bi-affine equations derived from the combined representation of the Inverse S-box are linear dependent with the batch 1 equations.

Important to note that five equations of set 1 do contain a constant part and are therefore only valid when $x \neq 0$. By linearly combining those five equations with each other, four equations can be derived without a constant part and one equation will remain with a constant part, having a probability of 255/256.

Through the combined S-box representation we are able to obtain with probability 1, a total of 23 bi-affine equations in 81 terms, or a total of 39 fully quadratic equations in 137 terms. Linear combinations within the sets 1, 2 and 3 are granting the 24 bi-affine equations in GF(2) which were derived by Courtois and Pieprzyk [3].

3.2 S-box algebraic complexity and immunity

The lowest known algebraic complexity of the AES S-box is 9 terms. With the introduction of the combined S-box representation the *algebraic complexity* of the AES S-box became *6 terms*.

The immunity of an $n \times n$ S-box or the Resistance against Algebraic Attacks for r equations in t terms, is reflecting the difficulty of solving multivariate equations and can be measured by the criterion of Cheon and Lee [4], denoted by $\boldsymbol{\Gamma}$ and defined as

$$\boldsymbol{\Gamma} = \left(\frac{\boldsymbol{t-r}}{\boldsymbol{n}}\right)^{(t-r)/n}$$

Courtois and Pieprzyk [3] are utilising an approximation for the algebraic immunity measurement, namely

$$\boldsymbol{\Gamma}_{\mathbf{appr}} = \left(\frac{\boldsymbol{t}}{\boldsymbol{n}}\right)^{t/r}$$

For the S-box combined representation with only bi-affine equations we obtained the same number as Courtois and Pieprzyk [3] did achieve, namely 31 equations in 81 terms, resulting in an algebraic immunity of $\boldsymbol{\Gamma} = (58/8)^8 \simeq 2^{22.9}$ and $\boldsymbol{\Gamma}_{\mathbf{appr}} = (81/8)^4 \simeq 2^{13.4}$.

3.3 Improved S-boxes versus S-box combined representation

Even there has never been a successful attack on the AES, many cryptanalysis investigated the cryptographic properties of the S-box. At the end, improved S-boxes were proposed and those do own better cryptographic properties than the AES S-box.

One can say that the new combined S-box representation can be constructed for any modulus which is determining the multiplicative inverses, for any additive constants and affine matrix. Changes in modulus with or without additive constant change, will not affect the algebraic complexity of the S-box combined representation.

However, a change of the affine matrix will affect the S-box combined representation in terms of algebraic complexity. It's important to emphasise that a combined S-box representation can be written too for the improved S-box achieved by modifying the AT, then applying the multiplicative inverse and finally adding again an AT [5].

Regarding the modulus, used to determine the multiplicative inverse, and the additive constant after the affine matrix, Das *et al.* [6] sorted out that irreducible polynomial {11*D*}, and an additive constant '31' generates higher Proportion-of-Passing values (PoP) by the 15 NIST Tests than the AES S-box. As stated above, the combined representation is independent of any modulus with or without additive constant change. Therefore, similar to the workout of the S-box combined representation in the Appendix, the combined representation for {11*D*} modulus and additive constant '31' can be worked out

$$\boldsymbol{S_i(Z) = X = \frac{1}{'C1' \cdot Z + \text{`}8B' \cdot z_0 + \text{`}0D' \cdot z_1 + \text{`}1A' \cdot z_5 + \text{`}34' \cdot z_6 + \text{`}72' \cdot z_7 + \text{`}05'}}$$

$$S_{\{11D_{31}\}}(x) = z = \frac{\grave{}1F'}{X(t)} + B4'' \cdot y_7 + 54'' \cdot y_6 + 24'' \cdot y_5 + 1C'' \cdot y_4 + 31$$

owning an algebraic complexity of *6 terms*.

Another important property of the S-box is the AT period. Cui *et al.* [5] presented an improved S-box by selecting a new AT and using it twice. The selected AT has the largest period, namely 16, and the iterative period of the improved S-box reaches 256, the maximum.

As a first step we will present the combined representation for an improved S-box which is constructed by the known structure, first taking the multiplicative inverse and then applying an AT. The applied AT is a new AT owning the largest possible transformation period and selected by Cui *et al.* [5]

$$z = L_{5B,5D}(y)$$

or

$$\begin{bmatrix} z_7 \\ z_6 \\ z_5 \\ z_4 \\ z_3 \\ z_2 \\ z_1 \\ z_0 \end{bmatrix} = \begin{bmatrix} 11011010 \\ 01101101 \\ 10110110 \\ 01011011 \\ 10101101 \\ 11010110 \\ 01101011 \\ 10110101 \end{bmatrix} \begin{bmatrix} y_7 \\ y_6 \\ y_5 \\ y_4 \\ y_3 \\ y_2 \\ y_1 \\ y_0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \end{bmatrix}$$

with $L_{u,v}$ an AT, $u \in GF(2^8)/\{0\}$ and $v \in GF(2^8)$.

Based on the work out for the combined representation (see the Appendix), this improved S-box, with only an affine matrix and additive constant change, is presented as follows:

$$S_{L_{5B,5D}}(x) = z = \frac{\grave{}5B'}{x} + \grave{}CF' \cdot y_7 + \grave{}E7' \cdot y_6 + \grave{}FE' \cdot y_5 + \grave{}72' \cdot y_4 + \grave{}34' \cdot y_3 + \grave{}1A' \cdot y_2 + \grave{}5D'$$

The algebraic complexity became *8 terms* by changing the AES AT into an AT with the highest transformation period.

More interesting is whether the combined representation can be constructed for the improved S-box construction, consisting of

- applying the AT $L_{5B,5D}$, denoted as

$$x' = L_{5B,5D}(x)$$

- taking the multiplicative inverse of x'

$$x'' = (x')^{-1}$$

- applying the AT $L_{5B,5D}$ again

$$z = L_{5B,5D}(x'')$$

The authors denoted the inverse AT as $L_{0E,25}$, which has to construct the improved inverse S-box. The *improved inverse S-box* constructions are consisting of

- applying the inverse AT $L_{0E,25}$, denoted as

$$x'' = L_{0E,25}(z)$$

- taking the multiplicative inverse of x''

$$x' = (x'')^{-1}$$

- applying the inverse AT $L_{0E,25}$ again

$$x = L_{0E,25}(x')$$

In order to construct a combined representation for such an improved S-box we need to make use of a meet-in-the-middle strategy. In other words, first we work out a combined representation for the improved S-box, $L_{5B,5D}(x)$, without performing the next step, namely the multiplicative inverse. This will grant us

$$x' =' 5B' \cdot x +' CF' \cdot x_7 +' E7' \cdot x_6 +' FE' \cdot x_5 +' 72' \cdot x_4 +' 34' \cdot x_3 +' 1A' \cdot x_2 +' 5D'$$

From the improved inverse S-box we construct the combined representation for $L_{0E,25}(z)$, which is step 1:

$$x'' =' 0E' \cdot z +' 46' \cdot z_7 +' 2E' \cdot z_6 +' 1A' \cdot z_5 + 25$$

The inverse of this combined representation

$$\frac{1}{x''} = x' = \frac{1}{'0E' \cdot z +' 46' \cdot z_7 +' 2E' \cdot z_6 +' 1A' \cdot z_5 +' 25'}$$

equals the combined representation of the improved S-box's first step and therefor

$$[\grave{}0E' \cdot z + \grave{}46' \cdot z_7 + \grave{}2E' \cdot z_6 + \grave{}1A' \cdot z_5 + \grave{}25'] \times [\grave{}5B' \cdot x + \grave{}CF' \cdot x_7 + \grave{}E7' \cdot x_6 + \grave{}FE' \cdot x_5 + \grave{}72' \cdot x_4 + \grave{}34' \cdot x_3 + \grave{}1A' \cdot x_2 + \grave{}5D'] = 1$$

Cui *et al.* [5] remark that sparse equations and overdefined equations, required for XL, XLS and Grobner base attacks, can be avoided due to the obtained algebraic complexity of 255 terms for the improved S-box and a complexity of 253 terms for the improved inverse S-box. The opposite is true if we look at the obtained combined representation for the improved S-box. This representation is generating 24 bi-linear equations with probability 1 in 81 terms through

$$x' \cdot x'' = 1$$
$$(x')^{128} \cdot x'' = (x'')^{128}$$
$$x' \cdot (x'')^{128} = (x')^{128}$$

providing an algebraic immunity or resistance slightly less than the AES S-box.

The algebraic immunity for this improved S-box is

$$\mathbf{\Gamma} = \left(\frac{57}{8}\right)^8 \simeq 2^{22.7}$$

$$\mathbf{\Gamma_{appr}} = \left(\frac{81}{8}\right)^4 \simeq 2^{13.4}$$

This calculation is a succinct illustration of the importance of a combined representation for S-boxes.

3.4 Improvement on 6-round 128-bit AES attack

Most known effective attack for the reduced-round version of 128-bit AES is the Partial Sum Attack introduced by Ferguson *et al.* [7]. The Partial Sum Attack improves the Square Attack presented by Daemen *et al.* [8]. Instead of guessing four bytes of the initial round key, the Partial Sum Attack uses 2^{32} plaintexts such that one column of the states in the first round after MixColumns operation varies over all possible values and the other bytes are constant. For any value of the first key round, the encryptions consist of 2^{24} groups of 2^8 encryptions that vary in a single byte at the end of the first round. The attack for a 6 round 128-bit AES can be set up by guessing four bytes of the 6th round key and one byte of the 5th round key. Based on guessing those five key bytes, the partial decryption to a single byte at the end of round 4 can be performed.

Finally, the sum of this single byte over all 2^{32} encryptions is checked for a zero result.

The improvement published by Ferguson *et al.* is the efficient organisation of the partial decryption through partial sums. By performing such efficiency, the Partial Sum Attack owns a computational complexity of 2^{44} along with 6.2^{32} plaintexts, to uniquely identify the proper value of the five key bytes.

In order to present improvements on the Partial Sum Attack with the S-box combined representation, the following notations are introduced:

Δ-set is one group of the utilised 2^{32} plaintexts,
$\boldsymbol{l}$ is the index of the lth element of a Δ-set,

and for the simplicity

$\boldsymbol{b_i^{(l)}}$ is the state byte at the 4th round output and therefor the result of the partial decryption,
$\boldsymbol{c_{l,m}}$ is the byte of the ciphertext with m the number of the four bytes that we use during partial decryption,
$\boldsymbol{k_0}, \boldsymbol{k_1}, \boldsymbol{k_2}, \boldsymbol{k_0}$ and $\boldsymbol{k_4}$ denote the five keys bytes that we are guessing,
$\boldsymbol{S_{in}}$ is the Inverse Rijndael S-box,
$\boldsymbol{i}, \boldsymbol{m} \in \{0, 1, 2, 3\}$ and $1 \leq l \leq 2^{32}$.

The efficiency of partial sums introduced by Ferguson *et al.* is performed within the partial decryption computation

$$\sum_{l=1}^{2^{32}} b_0^{(l)} = \sum_{l=1}^{2^{32}} S_{in}[\text{`}0E' \cdot S_{in}(c_{l,0} + k_0) + \text{`}0B' \cdot S_{in}(c_{l,1} + k_1) + \text{`}0D' \cdot S_{in}(c_{l,2} + k_2) + \text{`}09' \cdot S_{in}(c_{l,3} + k_3) + k_4]$$

After the Partial Sum Attack was published, researchers like Tunstall [9] and Aldà *et al.* [10] worked on finding further improvements on the number of sets of 2^{32} encryptions. The main focus of these works is based on analysing more information per Δ-set.

Checking the sum of a state byte at the 4th round output over all 2^{32} encryptions of a single Δ-set is expected to reject 255/256 of all wrong key guesses or in other words to eliminate all but ∼1 key value. And based on the number of guessed key bytes the required number of Δ-sets is set in order to verify the guessed key bytes correctness. At each positive verification with one Δ-set the key space is reduced by a factor 2^{-8} because the probability that the sum of a sequence of 256 random values in GF(2^8) equals zero is

$$\Pr\left(\sum_{l=1}^{256} X^{(l)} = 0\right) = \frac{1}{2^8}$$

Given the inverse of a combined representation of the Inverse S-box

$$S_{in}^{-1}(\boldsymbol{Z}) = \text{`}C1' \cdot \boldsymbol{Z} + \text{`}8\text{B}' \cdot z_0 + \text{`}0D' \cdot z_1 + \text{`}1A' \cdot z_5 + \text{`}34' \cdot z_6 + \text{`}72' \cdot z_7 + \text{`}05'$$

one can easily apply the inverse of the combined representation of the Inverse S-box on a state byte at the 4th round output and sum the new value over all 2^{32} encryptions. By performing this operation the 'new states' at the end of the 4th round are still balanced

$$\sum_{l=1}^{2^{32}} S_{in}^{-1}\left(b_0^{(l)}\right) = 0$$

and the computation of the partial decryption is adapted with one additional operation, namely S_{in}^{-1}. This adaptation does not have any negative impact on the amount of work during partial sums

$$\sum_{l=1}^{2^{32}} S_{in}^{-1}[S_{in}[\text{`}0E' \cdot S_{in}(c_{l,0} + k_0) + \text{`}0B' \cdot S_{in}(c_{l,1} + k_1) + \text{`}0D' \cdot S_{in}(c_{l,2} + k_2) + \text{`}09' \cdot S_{in}(c_{l,3} + k_3) + k_4]]$$

In order to understand whether applying the inverse of the combined representation of the Inverse S-box does impact the required number of Δ-sets, the key space reduction per positive verification with one Δ-set needs to be reconsidered. Therefore the probability that the sum of a sequence of random values equals zero is calculated again

$$\Pr\left(\sum_{l=1}^{256} S_{in}^{-1}\left(X^{(l)}\right) = 0\right) = \Pr\left(\text{`}C1' \cdot \sum_{l=1}^{256} X^{(l)} + \sum_{l=1}^{256} \text{`}8B' \cdot x_0^{(l)} + \text{`}0D' \cdot x_1^{(l)} + \text{`}1A' \cdot x_5^{(l)} + \text{`}34' \cdot x_6^{(l)} + \text{`}72' \cdot x_7^{(l)} = 0\right) \simeq \frac{1}{288}$$

with $x_0, x_1, x_5, x_6, x_7 \in GF(2)$.

Applying the inverse of the combined representation of the Inverse S-box on a state byte at the end of the 4th round output and summing this new value over all 2^{32} encryptions of a single Δ-set, is expected to reject 287/288 of all wrong key guesses or in other words to eliminate slightly more than 1 key value. Only five Δ-sets are required to determine the five key bytes.

By utilising the combined representation of the Inverse S-box, the Partial Sum Attack can be improved in data complexity from 6. 2^{32} to 5. 2^{32} chosen plaintexts. The improvement in the time complexity is negligible to be mentioned and remains 2^{44}.

4 AES system transformation

With the AES system transformation we aim to evolve the AES system of algebraic equations into a more solvable algebraic system by utilising the S-box's combined representation. Two methods are proposed for AES system transformation

- Byte multiplication per equation.
- Field trace of AES system.

An ideal solvable AES algebraic system is achieved when after transformation the S-box remainder bits $y_7y_6y_5y_4$ are not available anymore and we have to deal with multiplicative inverses in every equation.

First of all, the combined representation of the S-box assumes that '00''s multiplicative inverse is '00', and this enables the AES system of algebraic equations to work without any exception or any other restriction.

4.1 Byte multiplication or FD multiplication

Each equation in the AES algebraic system consisting of an S-box can be multiplied with the field element 'FD' providing another shape for the combined representation of the S-box:

$$S(X(t)) = \frac{\text{`}1F'}{X(t)} + \text{`}96' \cdot y_7 + \text{`}46' \cdot y_6 + \text{`}2E' \cdot y_5 + \text{`}1A' \cdot y_4 + \text{`}63'$$

$$\text{FD} \cdot S(X(t)) = \frac{\text{`}25'}{X(t)} + 0F \cdot y_7 + 07 \cdot y_6 + 03 \cdot y_5 + 01 \cdot y_4 + B1$$

$$\text{FD} \cdot S(X(t)) = \frac{\text{`}25'}{X(t)} + 0m + B1$$

with unknown $m \in \text{GF}(2^4)$.

Such a new shape provides a system of equations with a new constant of 'half-byte unknowns' in the form of '0m' or '1m' in the

Table 1 Field Trace of GF(2^8)

Tr	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
D	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
E	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

cipher part depending whether an odd number of y_7 bits is 'set' in the equation at the columns multiplied by '02' and '03' due to the MixColumns operation. An example can be given with the first equation in round 1

$$\text{`02'}\ S(p_0) + \text{`03'}\ S(p_5) + S(p_{10}) + S(p_{15}) + \ b_0 = q_0$$
$$b_0 = S(a_{13}) + a_0 + 01$$

Having both multiplied with 'FD' results in

$$\frac{\text{`4}A'}{p_0} + \frac{\text{`6}F'}{p_5} + \frac{\text{`25'}}{p_{10}} + \frac{\text{`25'}}{p_{15}} + \text{`0}m_0' + \text{`}B1' + \text{`FD'}b_0 = \text{`FD'}q_0$$

or

$$\frac{\text{`4}A'}{p_0} + \frac{\text{`6}F'}{p_5} + \frac{\text{`25'}}{p_{10}} + \frac{\text{`25'}}{p_{15}} + \text{`1}m_0' + \text{`}B1' + \text{`FD'}b_0 = \text{`FD'}q_0$$

and

$$\text{FD}b_0 = \frac{\text{`25'}}{a_{13}} + B1 + 0m_0' + \text{FD}a_0 + \text{FD}$$

with unknowns $m_0, m_0' \in GF(2^4)$.

$0m_0$ is valid when Most Significant Bits (MSB) y_7 of $1/p_0$ and y_7 of $1/p_5$ are equal: $y_{7@(1/p_0)} + y_{7@(1/p_5)} = 0$, and $\text{`}1m_0'$ is valid when MSB bits y_7 of $1/p_0$ and y_7 of $1/p_5$ are unequal: $y_{7@(1/p_0)} + y_{7@(1/p_5)} = 1$.

By assigning three half-byte unknowns '0m' or '1m' for the first three equations in each block of four equations per round, the half-byte unknown of the 4th equation can be calculated by XOR'ing the three assigned half-byte unknowns. This can be proven easily in GF(2) equations

$$y_{7@(1/p_0)} + y_{7@(1/p_5)} = n_0$$
$$y_{7@(1/p_5)} + y_{7@(1/p_{10})} = n_1$$
$$y_{7@(1/p_{10})} + y_{7@(1/p_{15})} = n_2$$
$$y_{7@(1/p_0)} + y_{7@(1/p_{15})} = n_0 + n_1 + n_2$$

with $n_0, n_1, n_2 \in GF(2)$.

Because the MixColumns operation is not applied in the key schedule part, there's no need for assignments at that part and therefor the half-byte unknown '0m' is available in all the equations with S-box. Given that 12 assignments are required for '0m' or '1m' per round, a complexity of 2^{108} for 10 rounds AES can be achieved with the condition that the newly constructed AES system of algebraic equations with half-byte unknowns '1m' and '0m' is able to be solved.

An example of such newly transformed AES algebraic system is shown for the first two blocks of four equations in round 1 in which the first three half-byte unknowns are assigned and the 4th half-byte unknown is calculated (in bold)

$$\frac{\text{`4}A'}{p_0} + \frac{\text{`6}F'}{p_5} + \frac{\text{`25'}}{p_{10}} + \frac{\text{`25'}}{p_{15}} + \text{`0}m_0' + \text{`}B1' + \text{`}FD'b_0 = \text{`}FD'q_0$$
$$\frac{\text{`25'}}{p_0} + \frac{\text{`4}A'}{p_5} + \frac{\text{`6}F'}{p_{10}} + \frac{\text{`25'}}{p_{15}} + \text{`1}m_1' + \text{`}B1' + \text{`}FD'b_1 = \text{`}FD'q_1$$
$$\frac{\text{`25'}}{p_0} + \frac{\text{`25'}}{p_5} + \frac{\text{`4}A'}{p_{10}} + \frac{\text{`6}F'}{p_{15}} + \text{`0}m_2' + \text{`}B1' + \text{`}FD'b_2 = \text{`}FD'q_2$$
$$\frac{\text{`6}F'}{p_0} + \frac{\text{`25'}}{p_5} + \frac{\text{`25'}}{p_{10}} + \frac{\text{`4}A'}{p_{15}} + \text{`}\mathbf{1}m_3' + \text{`}B1' + \text{`}FD'b_3 = \text{`}FD'q_3$$

$$\frac{\text{`4}A'}{p_4} + \frac{\text{`6}F'}{p_9} + \frac{\text{`25'}}{p_{14}} + \frac{\text{`25'}}{p_3} + \text{`1}m_4' + \text{`}B1' + \text{`}FD'b_4 = \text{`}FD'q_4$$
$$\frac{\text{`25'}}{p_4} + \frac{\text{`4}A'}{p_9} + \frac{\text{`6}F'}{p_{14}} + \frac{\text{`25'}}{p_3} + \text{`1}m_5' + \text{`}B1' + \text{`}FD'b_5 = \text{`}FD'q_5$$
$$\frac{\text{`25'}}{p_4} + \frac{\text{`25'}}{p_9} + \frac{\text{`4}A'}{p_{14}} + \frac{\text{`6}F'}{p_3} + \text{`0}m_6' + \text{`}B1' + \text{`}FD'b_6 = \text{`}FD'q_6$$
$$\frac{\text{`6}F'}{p_4} + \frac{\text{`25'}}{p_9} + \frac{\text{`25'}}{p_{14}} + \frac{\text{`4}A'}{p_3} + \text{`}\mathbf{0}m_7' + \text{`}B1' + \text{`}FD'b_7 = \text{`}FD'q_7$$

4.2 Field trace of AES system

Based on the S-box's combined representation, a nearly function of multiplicative inverse, a transformation can be conducted through the Absolute Trace, namely the Field Trace of the AES system of algebraic equations. There has been studies around Field Traces [11] applied on AES elements, aiming to achieve a more compact S-box and therefor to minimise the circuitry and the chip area required for the S-box.

With Lidl and Niederreiter's [12] definition of Traces, one can easily apply the transformation from finite extension $\boldsymbol{F} = \mathbb{F}_{2^8} = GF(2^8)$ to finite field $\boldsymbol{K} = \mathbb{F}_2 = GF(2)$ for any $\boldsymbol{\alpha} \in \mathbb{F}_{2^8}$ to $\boldsymbol{\beta} \in \mathbb{F}_2$ defined by

$$\boldsymbol{Tr}_{\boldsymbol{F}/\boldsymbol{K}}(\boldsymbol{\alpha}) = \boldsymbol{\beta} = \boldsymbol{\alpha} + \boldsymbol{\alpha}^2 + \boldsymbol{\alpha}^4 + \boldsymbol{\alpha}^8 + \boldsymbol{\alpha}^{16} + \boldsymbol{\alpha}^{32} + \boldsymbol{\alpha}^{64} + \boldsymbol{\alpha}^{128}$$

$\boldsymbol{Tr}_{\boldsymbol{F}/\boldsymbol{K}}(\boldsymbol{\alpha})$ is the absolute trace of $\boldsymbol{\alpha}$ and we will simply denote $\boldsymbol{Tr}(\boldsymbol{\alpha})$ and call it the Field Trace. An example shows the field trace transformation of '63' to GF(2) (see equation below) and as a result we obtained the field traces for all elements in GF(2^8) as presented in Table 1.

We require two properties of the Field Trace transformation

$$\boldsymbol{Tr}_{\boldsymbol{F}/\boldsymbol{K}}(\boldsymbol{\alpha} + \boldsymbol{\beta}) = \boldsymbol{Tr}_{\boldsymbol{F}/\boldsymbol{K}}(\boldsymbol{\alpha}) + \boldsymbol{Tr}_{\boldsymbol{F}/\boldsymbol{K}}(\boldsymbol{\beta}) \quad (1)$$

$$\boldsymbol{Tr}_{\boldsymbol{F}/\boldsymbol{K}}(c\boldsymbol{\alpha}) = c\boldsymbol{Tr}_{\boldsymbol{F}/\boldsymbol{K}}(\boldsymbol{\alpha}) \quad \text{forall} \quad c \in K, \boldsymbol{\alpha} \in F \quad (2)$$

to end up with a more compact expression for the S-box's transformation to the finite field $\boldsymbol{K} = \mathbb{F}_2$. By using (1) for S-box's Field Trace

$$\boldsymbol{Tr}(63) = 63 + (\text{`}63')^2 + (\text{`}63')^4 + (\text{`}63')^8 + (\text{`}63')^{16} + (\text{`}63')^{32} + (\text{`}63')^{64} + (\text{`}63')^{128}$$
$$\boldsymbol{Tr}(\text{`}63') = \text{`}63' + \text{`}C2' + \text{`}35' + \text{`}66' + \text{`}D3' + \text{`}2F' + \text{`}39' + \text{`}36' = 1$$

$$Tr[S(X(t))] = Tr\left[\frac{\grave{}1F'}{X(t)} + \grave{}96' \cdot y_7 + \grave{}46' \cdot y_6 + \grave{}2E' \cdot y_5 + \grave{}1A' \cdot y_4 + \grave{}63'\right]$$

we get

$$Tr[S(X(t))] = Tr\left[\frac{\grave{}1F'}{X(t)}\right] + Tr[\grave{}96' \cdot y_7] + Tr[\grave{}46' \cdot y_6] + Tr[\grave{}2E' \cdot y_5] + Tr[\grave{}1A' \cdot y_4] + Tr[\grave{}63']$$

and finally (2) enables to give us that more compact expression when the Field Traces of the available GF(2^8) elements are filled in

$$Tr[S(X(t))] = Tr\left[\frac{\grave{}1F'}{X(t)}\right] + y_7 Tr[\grave{}96'] + y_6 Tr[\grave{}46'] + y_5 Tr[\grave{}2E'] + y_4 Tr[\grave{}1A'] + Tr[\grave{}63']$$

$$Tr[S(X(t))] = Tr\left[\frac{\grave{}1F'}{X(t)}\right] + y_7 \cdot 1 + y_6 \cdot 0 + y_5 \cdot 1 + y_4.0 + 1$$

$$Tr[S(X(t))] = Tr\left[\frac{\grave{}1F'}{X(t)}\right] + y_7 + y_5 + 1$$

The basic S-box's Field Trace is not going to support directly an ideal Field Trace of the AES system as we are still in the possession of bits, $\boldsymbol{y}_7$ and $\boldsymbol{y}_5$, per S-box but it will serve as the base for the extended S-box Field Trace hereafter.

The extended S-box Field Trace can be introduced as the 'only' operation that is able to transform the S-box into a pure function of a multiplicative inverse. Based on the basic S-box Field Trace workout, before transforming the S-box with Field Trace, it can be multiplied by the field element 'C8'

$$C8 \cdot S(X(t)) = C8 \cdot \left[\frac{\grave{}1F'}{X(t)} + \grave{}96' \cdot y_7 + \grave{}46' \cdot y_6 + \grave{}2E' \cdot y_5 + \grave{}1A' \cdot y_4 + \grave{}63'\right]$$

and worked out through

$$C8 \cdot S(X(t)) = \frac{\grave{}60'}{X(t)} + 09 \cdot y_7 + 56 \cdot y_6 + F4 \cdot y_5 + A5 \cdot y_4 + FB$$

towards an extended S-box Field Trace

$$Tr[\mathrm{C8} \cdot S(X(t))] = Tr\left[\frac{\grave{}60'}{X(t)} + \grave{}09' \cdot y_7 + \grave{}56' \cdot y_6 + \grave{}F4' \cdot y_5 + \grave{}A5' \cdot y_4 + \grave{}FB'\right]$$

$$Tr[\mathrm{C8} \cdot S(X(t))] = Tr\left[\frac{\grave{}60'}{X(t)}\right] + Tr[\grave{}09' \cdot y_7] + Tr[\grave{}56' \cdot y_6] + Tr['F4' \cdot y_5] + Tr[\grave{}A5' \cdot y_4] + Tr['FB']$$

$$Tr[\mathrm{C8} \cdot S(X(t))] = Tr\left[\frac{\grave{}60'}{X(t)}\right] + y_7 Tr[\grave{}09'] + y_6 Tr[\grave{}56'] + y_5 Tr[\grave{}F4'] + y_4 Tr[\grave{}A5'] + Tr[\grave{}FB']$$

$$Tr[\mathrm{C8} \cdot S(X(t))] = Tr\left[\frac{\grave{}60'}{X(t)}\right] + y_7 \cdot 0 + y_6 \cdot 0 + y_5 \cdot 0 + y_4 \cdot 0 + 0$$

to reach a final format

$$\boldsymbol{Tr}[\mathrm{C8} \cdot \boldsymbol{S}(\boldsymbol{X}(\boldsymbol{t}))] = \boldsymbol{Tr}\left[\frac{\grave{}60'}{\boldsymbol{X}(\boldsymbol{t})}\right]$$

We need to remark that besides 'C8', six other field elements – '35', '76', '43', 'BE', 'FD' and '8B' – are able to create extended S-box Field Traces too in which no bit remainder or any constant can be found.

On the other hand, eight other field elements – '29', '97, 'A2', 'D4', 'E1', '1C', '5F', '6A' – can generate extended S-box Field Traces without bit remainders but with a constant remainder. An extended S-box Field Trace with field element '29' is worked out

$$Tr[29 \cdot S(X(t))] = Tr\left[\frac{\grave{}2A'}{X(t)}\right] + y_7 Tr[\grave{}17'] + y_6 Tr[\grave{}58'] + y_5 Tr['F2'] + y_4 Tr['A7'] + Tr[\grave{}82']$$

$$Tr[29 \cdot S(X(t))] = Tr\left[\frac{\grave{}2A'}{X(t)}\right] + y_7 \cdot 0 + y_6 \cdot 0 + y_5 \cdot 0 + y_4 \cdot 0 + 1 = Tr\left[\frac{\grave{}2A'}{X(t)}\right] + Tr[\grave{}82']$$

Looking back at the AES system of algebraic equations, and before we can name the Field Trace of the AES system, we have to notice that besides the $S(x)$ term, '02'$S(x)$ and '03'$S(x)$ terms are available too in the cipher part equations.

From analysis, we can report that the seven GF(2^8) elements of group Extended S-box Field Trace (ESFT) = {'*C*8', '35', 'FD', '97', '*A*2', '5*F*', '6*A*'} are the only ones that can be used for the extended S-box Field Trace to transform an AES cipher part algebraic equation into an equation with only multiplicative inverses and constants, and no bit remainders. An illustration is given why '76' is not suited as the extended S-box Field Trace for the transformation of the AES system. First by multiplying round 1's first equation

$$76[\grave{}02'S(p_0) + \grave{}03'S(p_5) + S(p_{10}) + S(p_{15}) + b_0] = \grave{}76'q_0$$

$$\grave{}EC'S(p_0) + \grave{}9A'\ S(p_5) + \grave{}76'S(p_{10}) + \grave{}76'S(p_{15}) + \grave{}76'b_0 = \grave{}76'q_0$$

and taking the Field Trace

$$Tr\left[\frac{'D1'}{p_0}\right] + \grave{}3A' \cdot y_{7@(1/p_0)} + Tr\left[\frac{\grave{}34'}{p_5}\right] + \grave{}27' \cdot y_{7@(1/p_5)} + Tr\left[\frac{'E5'}{p_{10}}\right] + Tr\left[\frac{'E5'}{p_{15}}\right] + Tr[\grave{}76'b_0] = Tr[\grave{}76'q_0]$$

2 bit remainders are part of the Field Trace transformation in one equation, whereas e.g. field element 'FD' provides no bit remainders. Given the same AES equation of round 1 and 'FD' multiplied

$$\mathrm{FD}[\grave{}02'S(p_0) + \grave{}03'S(p_5) + S(p_{10}) + S(p_{15}) + b_0] = \grave{}\mathrm{FD}'q_0$$

$$\grave{}E1'S(p_0) + \grave{}1C'S(p_5) + \grave{}\mathrm{FD}'S(p_{10}) + \grave{}\mathrm{FD}'S(p_{15}) + \grave{}\mathrm{FD}'b_0 = \grave{}\mathrm{FD}'q_0$$

results in a Field Trace transformation

$$Tr\left[\frac{\grave{}4A'}{p_0}\right] + Tr[\grave{}79'] + Tr\left[\frac{\grave{}6F'}{p_5}\right] + Tr['C8'] + Tr\left[\frac{\grave{}25'}{p_{10}}\right] + Tr\left[\frac{\grave{}25'}{p_{15}}\right] + Tr['\mathrm{FD}'b_0] = Tr['\mathrm{FD}'q_0]$$

$$Tr\left[\frac{\grave{}4A'}{p_0}\right] + Tr\left[\frac{\grave{}6F'}{p_5}\right] + Tr\left[\frac{\grave{}25'}{p_{10}}\right] + Tr\left[\frac{\grave{}25'}{p_{15}}\right] + Tr['B1'] + Tr['\mathrm{FD}'b_0] = Tr['\mathrm{FD}'q_0]$$

with only multiplicative inverses and a constant. Moreover, the group ESFT can be further reduced to a new group with only three elements, namely Extended S-box Field Trace Generators (ESFTG) = {'C8', '97', '6A'} since any combination of the ESFTs created by these three elements will generate all possible ESFTs created by the ESFT group's elements.

We can reveal now that the Field Trace of the AES system, carried through ESFTG elements, is IDENTICAL to the Field Trace of a system of algebraic equations with multiplicative inverses and constants, instead of S-boxes, and carried through the same ESFTG elements. As a basic example the first block of four equations in round 1, carried through ESFTG element '*C*8', is given as an illustration to enlighten the theorem's idea of 'identical'. The Field Trace of round 1's first four equations in AES system

$$\begin{aligned}
&C8 \cdot [\text{`}02' S(p_0) + \text{`}03' S(p_5) + S(p_{10}) + S(p_{15}) + b_0] \\
&\quad = \text{`}C8' \cdot q_0 \\
&C8 \cdot [S(p_0) + \text{`}02' S(p_5) + \text{`}03' S(p_{10}) + S(p_{15}) + b_1] \\
&\quad = \text{`}C8' \cdot q_1 \\
&C8 \cdot [S(p_0) + S(p_5) + \text{`}02' S(p_{10}) + \text{`}03' S(p_{15}) + b_2] \\
&\quad = \text{`}C8' \cdot q_2 \\
&C8 \cdot [\text{`}03' S(p_0) + S(p_5) + S(p_{10}) + \text{`}02' S(p_{15}) + b_3] \\
&\quad = \text{`}C8' \cdot q_3
\end{aligned}$$

is IDENTICAL to the Field Trace of the following four algebraic equations:

$$\begin{aligned}
&\frac{'C0'}{p_0} + \frac{'A0'}{p_5} + \frac{\text{`}60'}{p_{10}} + \frac{\text{`}60'}{p_{15}} + \text{`}C8'b_0 = \text{`}C8'q_0 \\
&\frac{\text{`}60'}{p_0} + \frac{'C0'}{p_5} + \frac{'A0'}{p_{10}} + \frac{\text{`}60'}{p_{15}} + \text{`}C8'b_1 = \text{`}C8'q_1 \\
&\frac{\text{`}60'}{p_0} + \frac{\text{`}60'}{p_5} + \frac{'C0'}{p_{10}} + \frac{'A0'}{p_{15}} + \text{`}C8'b_2 = \text{`}C8'q_2 \\
&\frac{'A0'}{p_0} + \frac{\text{`}60'}{p_5} + \frac{\text{`}60'}{p_{10}} + \frac{'C0'}{p_{15}} + \text{`}C8'b_3 = \text{`}C8'q_3
\end{aligned}$$

5 Conclusion

Expressing the AES as an organised GF(2^8) view of algebraic equations and utilising the newly presented combined representation of the S-box, enables us to make use of external operations on the system. Two methods for such operations are proposed and explored in this paper. Studies on external operations to transform the S-box expression, available in almost every equation of the AES system, into an algebraic format have not been performed yet. This paper does not describe an attack on AES and even we are far from the solution for AES, this paper is building the first steps in breaking the AES. One can remark that the S-box's combined representation may be a true added value for improvements at known algebraic attacks and efficient algorithm implementations [13–16].

First of all the two proposed methods in this paper rely on the capability of solving the AES system in GF(2^8) with the S-box replaced by a multiplicative inverse function. For that reason the methods on the 10 round AES system are depending on computation complexity and the feasibility of solving a huge system of multiplicative inverse functions instead of S-boxes. Even the methods' computation and feasibility challenges, we need to study and experiment the effect of different algebraic attacks' efficiency based on the new S-box representation through small scale variants of the AES family, presented by Cid *et al.* [17] and denoted by *SR*(*n*,*r*,*c*,*e*). This study with small scale variants may use an extended variant *SR'*(*n*,*r*,*c*,*e*,*b*) with an additional parameter *b*-the number of the remainder ' y_i *bits*' of the S-box's combined representation – and such a variant can exhibit more AES features than the existing small scale variants of AES.

Working out the FD multiplication method requires algebraic techniques and concepts that may not exist or we are unaware of their existence. A complexity of 2^{108} for the 10 round AES system through FD multiplication may appeal to imagination but to achieve this complexity more investigation on the interaction of equations with half-byte unknowns or development on techniques is required.

It's a fact that the new concept, Field Trace of systems, is an interesting but an unhandled domain and as demonstrated in detail in this paper, the Field Trace of the AES system can be transformed to the Field Trace of a new system without S-boxes. Future research is certainly required to investigate the solutions of such new system and how its variables' Field Traces may apply to the AES variables' Field Traces, and in particular the AES key variable.

6 References

[1] Daemen, J., Rijmen, V.: 'AES Proposal: Rijndael, AES Algorithm Submission, September 1999

[2] Ferguson, N., Schroeppel, R., Whiting, D.: '*A simple algebraic representation of Rijndael*' (Springer-Verlag London, UK, 2001)

[3] Courtois, N., Pieprzyk, J.: 'Cryptanalysis of block ciphers with overdefined systems of equations'. IACR eprint server 2002/044, March 2002

[4] Cheon, J.H., Lee, D.H.: '*Resistance of S-boxes against algebraic attacks*', (Lecture Notes in Computer Science 3017, Springer, Berlin, Heidelberg, 2004), pp. 83–93

[5] Cui, J., Huang, L., Zhong, H., *et al.*: 'An improved AES S-box and its performance analysis', *Int. J. Innov. Comput. Inf. Control*, 2011, **7**, (5), pp. 2291–2302

[6] Das, S., Zaman, J. U., Ghosh, R.: 'Generation of AES S-boxes with various modulus and additive constant polynomials and testing their randomization', *Proc. Technol.*, 2013, **10**, pp. 957–962

[7] Ferguson, N., Kelsey, J., Lucks, S., *et al.*: *Improved cryptanalysis of Rijndael'*, In Fast Software Encryption' (Springer, Berlin, Heidelberg, 2001), pp. 213–230

[8] Daemen, J., Knudsen, L., Rijmen, V.: '*The block cipher square*', In Fast Software Encryption' (Springer, 1997), pp. 149–165

[9] Tunstall, M.: 'Improved 'partial sums'-based square attack on AES'. Int. Conf. on Security and Cryptography (SECRYPT 2012), Rome, Italy, 2012, pp. 25–34

[10] Aldà, F., Aragona, R., Nicolodi, L., *et al.*: 'Implementation and improvement of the partial sum attack on 6-round AES'. Workshop on Communication Security (WCS 2014), Ancona, Italy, 2014

[11] Canright, D.: 'A very compact Rijndael S-box'. Technical report, Naval Postgraduate School, Monterey, CA, USA, May 2005

[12] Lidl, R., Niederreiter, H.: '*Introduction to finite fields and their applications*' (Cambridge University Press, Cambridge, United Kingdom, 1986), Chapter 2 Structure of Finite Fields pp. 50–51

[13] Stoffelen, K.: 'Optimizing S-box implementations for several criteria using SAT solvers', 2016, eprint.iacr.org/2016/198.pdf

[14] Bogdanov, A., Pyshkin, A.: 'Algebraic Side-channel collision attacks on AES', eprint.iacr.org/2007/477.pdf, January 2007

[15] Standaert, F.-X., Rouvroy, G., Quisquater, J.J., *et al.*: Efficient implementation of Rijndael encryption in reconfigurable hardware'. The Proc. of CHES, Cologne, Germany, 2003, (Lecture Notes in Computer Science, 2779), pp. 334–350

[16] Rijmen, V., Oswald, E.: 'Representation and Rijndael descriptions', In Advanced Encryption Standard - AES, 2004

[17] Cid, C., Murphy, S., Robshaw, M.J.B.: 'Small scale variants of the AES', eprint.iacr.org/2017/007.pdf, January 2008

7 Appendix

In the classical polynomial representation with GF(2^8) the S-box's outcome is written with coefficients in {0,1}. If we take $X(t)$ as input value and $Z(X(t)) = S(X(t))$ as outcome of the S-box, both are represented as follows:

$$\begin{aligned}
X(t) &= x_7t^7 + x_6t^6 + x_5t^5 + x_4t^4 + x_3t^3 + x_2t^2 + x_1t + x_0 \\
S(X(t)) = Z(t) &= z_7t^7 + z_6t^6 + z_5t^5 + z_4t^4 \\
&\quad + z_3t^3 + z_2t^2 + z_1t + z_0
\end{aligned}$$

with $X(t), Z(t) \in GF(2^8)$; $x_7, x_6, x_5, x_4, x_3, x_2, x_1, x_0 \in GF(2)$ and $z_7, z_6, z_5, z_4, z_3, z_2, z_1, z_0 \in GF(2)$.

S-box coefficients $z_7z_6z_5z_4z_3z_2z_1z_0$ are obtained by processing the $X(t)$'s multiplicative inverse's coefficients, namely $y_7y_6y_5y_4y_3y_2y_1y_0$, through the AT over GF(2). As

$$Y(t) = \frac{1}{X(t)} = y_7t^7 + y_6t^6 + y_5t^5 + y_4t^4 + y_3t^3 + y_2t^2 + y_1t + y_0$$

and given the AT over GF(2),

$$\begin{bmatrix} z_0 \\ z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \\ z_7 \end{bmatrix} = \begin{bmatrix} 10001111 \\ 11000111 \\ 11100011 \\ 11110001 \\ 11111000 \\ 01111100 \\ 00111110 \\ 00011111 \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

an S-box representation may be represented too as a combination of polynomial coefficients in {0,1} and bytes. Working out the AT over GF(2) with the $Y(t)$ polynomial coefficients and adding the last matrix in byte format, we obtain an initial S-box notation as a combination of GF(2) coefficients in {0,1} and bytes in hexadecimal notion

$$\begin{aligned} S(X(t)) = Z(t) = {} & (y_3 + y_4 + y_5 + y_6 + y_7)t^7 \\ & + (y_2 + y_3 + y_4 + y_5 + y_6)t^6 \\ & + (y_1 + y_2 + y_3 + y_4 + y_5)t^5 \\ & + (y_0 + y_1 + y_2 + y_3 + y_4)t^4 \\ & + (y_0 + y_1 + y_2 + y_3 + y_7)t^3 \\ & + (y_0 + y_1 + y_2 + y_6 + y_7)t^2 \\ & + (y_0 + y_1 + y_5 + y_6 + y_7)t \\ & + (y_0 + y_4 + y_5 + y_6 + y_7) + \text{`}63' \end{aligned}$$

This combined representation of the S-box can be divided into six parts as follows:

1. a1) $y_7t^7 + y_6t^6 + y_5t^5 + y_4t^4 + y_3t^3 + y_2t^2 + y_1t + y_0$
2. b1) $y_6t^7 + y_5t^6 + y_4t^5 + y_3t^4 + y_2t^3 + y_1t^2 + y_0t$
3. c1) $y_5t^7 + y_4t^6 + y_3t^5 + y_2t^4 + y_1t^3 + y_0t^2$
4. d1) $y_4t^7 + y_3t^6 + y_2t^5 + y_1t^4 + y_0t^3$
5. e1) $y_3t^7 + y_2t^6 + y_1t^5 + y_0t^4$
6. f1) $y_7t^3 + y_7t^2 + y_6t^2 + y_7t + y_6t + y_5t + y_7 + y_6 + y_5 + y_4 + 63$

Further simplification of parts (a1), (b1), (c1), (d1) and (e1) are resulting, respectively, into

(a2) $(a2)\ Y(t)$
(b2) $(b2)\ t \cdot Y(t) + y_7t^8$
(c2) $t^2 \cdot Y(t) + y_7t^9 + y_6t^8$
(d2) $t^3 \cdot Y(t) + y_7t^{10} + y_6t^9 + y_5t^8$
(e2) $t^4 \cdot Y(t) + y_7t^{11} + y_6t^{10} + y_5t^9 + y_4t^8$
(f) $y_7t^3 + y_7t^2 + y_6t^2 + y_7t + y_6t + y_5t + y_7 + y_6 + y_5 + y_4 + 63$

Such a simplification already enables us to start defining a combined representation of the S-box

$$\begin{aligned} S(X(t)) = Z(t) = {} & \left(1 + t + t^2 + t^3 + t^4\right) \cdot Y(t) \\ & + y_7\left(t^{11} + t^{10} + t^9 + t^8 + t^3 + t^2 + t + 1\right) \\ & + y_6\left(t^{10} + t^9 + t^8 + t^2 + t + 1\right) \\ & + y_5\left(t^9 + t^8 + t + 1\right) + y_4\left(t^8 + 1\right) + 63 \end{aligned}$$

$1 + t + t^2 + t^3 + t^4$ is the polynomial representation of the byte with hexadecimal value '1F', and this is guiding us to a more simplified representation as follows:

$$\begin{aligned} S(X(t)) = Z(t) = {} & 1F \,.\, Y(t) \\ & + y_7\left(t^{11} + t^{10} + t^9 + t^8 + t^3 + t^2 + t + 1\right) \\ & + y_6\left(t^{10} + t^9 + t^8 + t^2 + t + 1\right) \\ & + y_5\left(t^9 + t^8 + t + 1\right) + y_4\left(t^8 + 1\right) + 63 \end{aligned}$$

Four elements within this new combined representation own a binary polynomial degree above 8, which can be reduced by the irreducible binary polynomial of degree 8 as given in [2], namely $m(t) = t^8 + t^4 + t^3 + t + 1$.

(1) For $y_7 : t^{11} + t^{10} + t^9 + t^8 + t^3 + t^2 + t + 1$ modulo $\left(t^8 + t^4 + t^3 + t + 1\right) = t^7 + t^4 + t^2 + t$
(2)
For $y_6 : t^{10} + t^9 + t^8 + t^2 + t + 1$ modulo $\left(t^8 + t^4 + t^3 + t + 1\right) = t^6 + t^2 + t$
(3)
For $y_5 : t^9 + t^8 + t + 1$ modulo $\left(t^8 + t^4 + t^3 + t + 1\right) = t^5 + t^3 + t^2 + t$
(4) For $y_4 : t^8 + 1$modulo $\left(t^8 + t^4 + t^3 + t + 1\right) = t^4 + t^3 + t$.

After the elements are reduced by the irreducible binary, they can be represented by bytes in hexadecimal notion

(1) For $y_7 : t^7 + t^4 + t^2 + t = 96$.
(2) For $y_6 : t^6 + t^2 + t = '46'$.
(3) For $y_5 : t^5 + t^3 + t^2 + t = 2E$.
(4) For $y_4 : t^4 + t^3 + t = 1A$.

The final combined representation of the S-box, in GF(2) and bytes in hexadecimal notion, shows us a S-box notation which can serve as a base for AES's algebraic cryptanalysis

$$\begin{aligned} \boldsymbol{S}(\boldsymbol{X}(\boldsymbol{t})) = {} & 1F \cdot \boldsymbol{Y}(\boldsymbol{t}) + 96 \cdot \boldsymbol{y}_7 + 46 \cdot \boldsymbol{y}_6 + 2E \cdot \boldsymbol{y}_5 \\ & + 1A \cdot \boldsymbol{y}_4 + \text{`}63' \\ = {} & \frac{'1\boldsymbol{F}'}{\boldsymbol{X}(\boldsymbol{t})} + 96 \cdot \boldsymbol{y}_7 + 46 \cdot \boldsymbol{y}_6 + 2E \cdot \boldsymbol{y}_5 \\ & + 1A \cdot \boldsymbol{y}_4 + 63 \end{aligned}$$

with '1*F*', '96', '46', '2*E*', '1*A*', '63' bytes in hexadecimal notion and $y_7, y_6, y_5, y_4 \in GF(2)$; $Y(t) = 1/X(t)$.