# Generating S-Boxes from Semi-fields Pseudo-extensions

Jean-Guillaume Dumas, Jean-Baptiste Orfila

# Generating S-Boxes from Semi-fields Pseudo-extensions

Jean-Guillaume Dumas Jean-Baptiste Orfila

Laboratoire J. Kuntzmann, Université de Grenoble. 51, rue des Mathématiques, umr CNRS 5224, bp 53X, F38041 Grenoble, France,
{Jean-Guillaume.Dumas,Jean-Baptiste.Orfila}@imag.fr.

**Abstract**

Block ciphers, such as the AES, correspond to a very important family of secret-key cryptosystems. The security of such systems partly relies on what is called the S-box. This is a vectorial Boolean function $f : \mathbb{F}_2^n \hookrightarrow \mathbb{F}_2^n$, where $n$ is the size of the blocks. It is often the only non linear operation in the algorithm. The most well-known attacks against block ciphers algorithms are the known-plaintext attacks called differential cryptanalysis [4, 10] and linear cryptanalysis [11]. To protect such cryptosystems against linear and differential attacks, S-boxes are designed to fulfill some cryptographic criteria (balancedness, high nonlinearity, high algebraic degree, avalanche, or transparency [2, 12]) and are usually defined on finite fields, like $\mathbb{F}_{2^n}$ [7, 3].

Unfortunately, it seems difficult to find good S-Boxes, at least for bijective ones: random generation does not work [8, 9] and the one used in the AES or Camellia are actually variations around a single function, the inverse function in $\mathbb{F}_{2^n}$. Would the latter function have an unforeseen weakness (for instance if more practical algebraic attacks are developped), it would be desirable to have some replacement candidates.

For that matter, we propose to weaken a little bit the algebraic part of the design of S-Boxes and use finite semi-fields instead of finite fields to build such S-Boxes. Finite semi-fields relax the associativity and commutativity of the multiplication law. While semi-fields of a given order are unique up to isomorphism, on the contrary semi-fields of a given order can be numerous: nowadays, on the one hand, it is for instance easy to generate all the 36 semi-fields of order $2^4$, but, on the other hand, it is not even known how many semi-fields are there of order $2^8$. Therefore, we propose to build S-Boxes via semi-fields pseudo extensions of the form $\mathbb{S}_{2^4}^2$, where $\mathbb{S}_{2^4}$ is any semi-field of order $2^4$, and mimic in this structure the use of the inverse function in a finite field.

We report here the construction of 10827 S-Boxes, 7052 non CCZ-equivalent, with maximal nonlinearity, differential invariants, degrees and bit interdependency. Among the latter 2963 had fix points, and among the ones without fix points, 3846 had the avalanche level of AES and 243

the better avalanche level of Camellia. Among the latter 232 have a better transparency level than the inverse function on a finite field.

# 1 Introduction

A substitution-box (abbreviate as s-box), is a tool used in symmetric ciphers in order to increase their resistance against known attacks, such as linear and differential cryptanalysis by breaking cipher linearity. Sboxes are commonly represented by boolean functions i.e. $S : \mathbb{F}_2^n \to \mathbb{F}_2^m$ , whose dimensions $n, m$ are depending on the cipher. For example, the AES sbox uses $n = m = 8$, views the finite field with 256 elements as a vector space on its base field, and is generated by:

$$\begin{array}{rcccc} T & : & \mathbb{F}_{2^8} & \to & \mathbb{F}_{2^8} \\ & & 0 & \mapsto & 0 \\ & & a & \mapsto & a^{-1} \end{array} \tag{1}$$

Once $T$ is computed, an affine transformation is applied [7], and it results in an excellent s-box from the point of view of security characteristics. More precisely, in the following, we will use the list of criteria described in [2]. These criteria measure s-boxes robustness with respect to possibles attacks. Among bijective s-boxes, only AES and Camellia's s-boxes have good scores on this measure and both are build on a modified inverse computation. Thus, would the latter function have an unforeseen weakness (for instance if more practical algebraic attacks are developped), it would be desirable to have some replacement candidates.

Rather than trying different constructions, some works [2], [8] have been made on random searches among the 256! possibilities of bijective s-boxes. Another approach is to design s-boxes via the use of chaotic maps [9]. Unfortunately, none of the s-boxes build from these searches have the resistance of AES against linear nor differential attacks.

Our idea is different, we replace the algebraic structure of AES and Camellia (namely viewing the vector space as a finite field) by another structure, a semifield. First, there exists different semifields of a given order up to isomorphism. Even when considering the more restrictive notion of isotopy [1], the semifields are still non unique. Therefore there could be several choices of underlying structure, even with a single function. Second, the nonzero elements of semifields still form a multiplicative group. Therefore an inverse-like function could very well preserve good cryptographic properties.

In Section 2, we recall the definition of semifields and propose a construction of a degree 2 pseudo-extension of semifields of order 16. From this construction we deduce bijective s-boxes over $\mathbb{F}_{256}$, that mimic the behavior of the function (1) above. We also present some efficient algorithms for semifields constructions in Section 3. Then we recall in Section 4, the criteria that we use to rank the obtained s-boxes. Finally, we show in Section 5 that our construction indeed yields novel s-Boxes that match the resistance of the best known ones.

## 2 Semi-fields pseudo-extensions

In this section, after defining semifields, we describe the construction of pseudo-extensions of semifields containing 16 elements.

**Definition 1.** *A finite semifield* $(\mathbb{S}, +, \times)$ *is a set* $\mathbb{S}$ *containing at least two elements, and associated with two binaries laws (addition and multiplication), such that:*

1. $(\mathbb{S}, +)$ *is a group with neutral* $0$
2. $\forall a, b \in \mathbb{S}, ab = 0 \Rightarrow a = 0$ *or* $b = 0$
3. $\forall a, b, c \in \mathbb{S} : a(b + c) = ab + ac$ *and* $(a + b)c = ac + bc$
4. $\forall a \in \mathbb{S}, \exists$ *a neutral element for* $\times$ *denoted as* $e$ *which satisfies:* $ea = ae = a$

Ideally, we would like to construct s-boxes using:

$$\begin{array}{rccc} T' & : & \mathbb{S}_{2^8} & \rightarrow & \mathbb{S}_{2^8} \\ & & 0 & \mapsto & 0 \\ & & a & \mapsto & a^{-1} \end{array}$$

Unfortunately, we do not know the complete classification of these semifields for the moment. Currently, the largest classification in characteristic 2 is of order 64 [13]. Thus, in order to obtain build s-boxes with 256 elements, we mimic the finite fields construction, based on a quotient structure: $\mathbb{F}_{2^8} = \mathbb{F}_{2^4}[X]/P_2$. However, the same notion of polynomial irreducibility is more difficult to define in semifields, due to the possible non-associativity.

Actually, we just need to build a bijection $T' : (\mathbb{S}_{2^4})^2 \rightarrow (\mathbb{S}_{2^4})^2$ as close as possible to the inverse function, in order to take advantage of its cryptographic properties. Therefore, we have to find an equivalent characterization to the polynomial irreducibility notion on in finite fields, applicable on semifields. Let $P(X) = X^2 + \alpha X + \beta$, with $\alpha, \beta \in \mathbb{F}_{2^4}$, be an irreducible polynomial of degree 2. Elements of $\mathbb{F}_{2^8}$ viewed as $\mathbb{F}_{2^4}[X]/P$ are polynomials of degree 1 of the form $aX + b$, denoted as couple $(a, b) \in \mathbb{F}_{2^4}^2$. Over the finite field $\mathbb{F}_{2^8}$, the inverse of $0X + b$ is $0X + u$, where $u = b^{-1} \in \mathbb{F}_{2^4}$ if $b \neq 0$. Then if $a \neq 0$, we let $\gamma \in \mathbb{F}_{2^4}$ be such that $\gamma = a^{-1}b$, in order to obtain an unitary couple and thus simplify the following computations. Then, the inverse of $aX + b$ is denoted $c'X + d'$ and we have $(aX + b)(c'X + d') = 1 \Leftrightarrow a(X + \gamma)(c'X + d') = 1$ or also $(X + \gamma)(cX + d) = 1$, with $c' = a^{-1}c$ and $d' = a^{-1}d$.
After degree identification, and replacing $X^2 = -\alpha X - \beta$, we obtain:

$$\left\{ \begin{array}{rcl} d\gamma - c\beta & = & 1 \\ c\gamma + d - c\alpha & = & 0 \end{array} \right. \tag{2}$$

Finally, we have:

$$\left\{ \begin{array}{rcl} c & = & [(\alpha - \gamma)\gamma - \beta]^{-1} \\ d & = & c(\alpha - \gamma) \end{array} \right. \tag{3}$$

From the previous equations, it is now easy to deduce the following alternative characterisation of irreducible polynomials of degree 2 over finite fields:

**Lemma 1.** *Let $P : X^2 + \alpha X + \beta, \in \mathbb{F}_{2^4}[X]$, $P$ is irreducible if and only if $\forall \gamma \in \mathbb{F}_{2^4}, [(\alpha - \gamma)\gamma - \beta] \neq 0$.*

Using Lemma 1, we thus propose the following definition over semifields:

**Definition 2** (Pseudo-irreducibility)**.** *Let $P = X^2 + \alpha X + \beta \in \mathbb{S}_{2^4}[X]$, $P$ is pseudo-irreducible if and only if $\forall \gamma \in \mathbb{S}_{2^4}, [(\alpha - \gamma)\gamma - \beta] \neq 0$.*

Thus, in the case where $\mathbb{S}_{2^4} \simeq \mathbb{F}_{2^4}$, our pseudo-irreducibility notion reduces to irreducibility. Now we are able to define our pseudo-inversion as:

**Lemma 2.** *Let $P : X^2 + \alpha X + \beta, \in \mathbb{S}_{2^4}[X]$ be a pseudo-irreducible polynomials. The transformation:*

$$\begin{array}{rccc} T' & : & (\mathbb{S}_{2^4})^2 & \rightarrow & (\mathbb{S}_{2^4})^2 \\ & & (0,0) & \mapsto & (0,0) \\ & & (0,b) & \mapsto & (0,b^{-1}) \\ & & (a,b) & \mapsto & (a^{-1}c, a^{-1}d) \end{array}$$

*such that $\gamma = a^{-1}b, c = [(\alpha - \gamma)\gamma - \beta]^{-1}$, and $d = c(\alpha - \gamma)$, is a bijection.*

*Proof.* In the case where $a = 0$, $T'$ is obviously one-to-one. Let us assume now that $a \neq 0$.

For proving injectivity, we suppose that $\exists \gamma_1, \gamma_2 \in \mathbb{S}_{2^4}$ such that $c(\alpha - \gamma_1) = c(\alpha - \gamma_2)$. Then $c\alpha - c\gamma_1 = c\alpha - c\gamma_2$, so that $c(\gamma_1 - \gamma_2) = 0$ and thus $c^{-1}c(\gamma_1 - \gamma_2) = 0$. Finally $\gamma_1 = \gamma_2$.

Then, as $\mathbb{S}_{2^4}^2$ has a finite cardinality, any injective endofunction is bijective. □

# 3 Semi-fields efficient generation

As a prerequisite for constructing pseudo-extensions, we need semifields of order $2^4$. In this section, we expose some results and optimizations about efficient generation of semifields.

Recent results about semifields are detailed in [6] and in particular, they show that we can represent semifields as matrix vector spaces:

**Proposition 1** ( [6], Prop 3.)**.** *There exists a finite semifield $\mathbb{S}$ of dimension $n$ over $\mathbb{F}_q \subseteq \mathbb{S}$ iff there exists a set of $n$ matrices $\{A_1, ...A_n\} \subseteq GL(n, q)$ such that:*

- *$A_1$ is the identity matrix*
- $\sum_{i=1}^{n} \lambda_i A_i \in GL(n, q), \forall (\lambda_1, ..., \lambda_n) \in \mathbb{F}_q^n \backslash \{0\}$

- *The first column of the matrix $A_i$ is the column vector with a 1 in the $i^{th}$ position, and 0 everywhere else.*

This proposition is fundamental, since it allows us to use efficients matrix computations to discover new semifields. In our case, we restrict this proposition for $q = 2$, and $n \leq 8$.

In order to generate semifields, we use the algorithms described in [13]. The idea is to select lists of matrices extracted from $GL(n, 2)$ with a prescibed first column. It is thus necessary to check invertibility of all possibles linear combinations, in order to gradually reduce the possible semifield candidates. In practice, the invertibility check is done by a determinant computation. Then, in order to accelerate the process, some combinations of matrices can be discarded, as they can yield already found spaces. This is formalized via the notion of *isotopy* of semifields:

**Definition 3** (Isotopy)**.** *Let $\mathbb{S}_1$ and $\mathbb{S}_2$ be two semifields over the same finite field $\mathbb{F}_p$, then an isotopy between $\mathbb{S}_1$ and $\mathbb{S}_2$ is a triple $(F, G, H)$ of bijective linear maps $\mathbb{S}_1 \to \mathbb{S}_2$ over $\mathbb{F}_p$ such that $H(ab) = F(a)G(b), \forall a, b \in \mathbb{S}_1$*

Definition 3 is used to define an equivalence relation between semifields, which can be verified with the help of matrix multiplications, see [6, Prop. 2].

Even if only square matrices with small size are involved, semifield generation remains complex for the large amount of computations involved. For instance, generation of all matrices constituting $GL(8, 2)$ could require $2^{64}$ determinant computations.

We thus propose in the following some optimzations for the computation of the determinant and of matrix multiplication, based on tabulation and Gray codes.

## 3.1 Optimizing determinant using Gray codes and tabulations

Classical determinant computations use Gaussian elimination, with a $O(\frac{2}{3}n^3)$ complexity for a single determinant computation. Thus, in order to build $GL(n, 2)$ by testing all the possible matrices, we obtain an overall complexity of $O(\frac{2}{3}n^3 2^{n^2})$. Here, we present two ways to reduce this complexity.

The first optimization is about tabulating the computations via the recurrence formula of the determinant:

- If $A$ is $1 \times 1$ matrix, $det(A) = a$, with $A = (a)$.
- Otherwise, $n \geq 2$,
$$det(A) = \sum_{i=1}^{n} (-1)^{i+j} a_j M_{1j}$$
with $M_{1j}$ the determinant of the submatrix defined as A deprived of its first row and of its $j^{th}$ column (we chose the first row deletion and the column development arbitrarilly).

More precisely, since we have to compute all determinants for each matrix size, the idea is to store them in order to accelerate the computations of the larger matrix dimension. By doing this, we replace a sub-determinant computation by a table access. The drawback of this method is the memory limitation, and we succeeded to apply it for square matrices up to size 6. Indeed, for $n = 7$, we should store $2^{49}$ computations, that being around 500 Tb of data.

Our second optimization is about improving the way of passing through all matrices. Since each matrix has a unique integer representation (using the $n^2$ bits as digits), the easiest method to go through all the determinants is to increment this integer representation until its largest value. However, it implies "random" modifications on the matrix binaries coefficients. By using a Gray code, which allows to pass from a value to another by modifying only one bit between them, we are thus able to pass from one determinant to the other by modifying only one term in the sum: the idea is to cut the matrix in two parts, the first row on the one hand, containing $n$ bits, and the remainding $n(n-1)$ coefficients, which we call the base, on the other hand. Then we apply two distinct Gray codes, one for each hand. First, we fix a value for the base, and then we go all over possibles values for the first row, following a Gray code on this row. Second, we change the base value with another dedicated Gray code, and go again through all possibles values from the first line. Memory exchange is thus reduced because we only need to access the lower dimension table $n$ times for each possible submatrix determinant, but for $2^n$ computations.

The complexity is also drastically reduced by linking successive computations. Indeed, by modifying only one bit between two values, the determinant computation is reduced to the following formula: $\Delta_k = \Delta_{k-1} \oplus M_{1j}$, where $M_{1j}$ is defined as in the previous formula, and $\Delta_0 = 0$, since in a Gray code the first number is 0. Thus, we reduced the determinant computation, which would normally requires $n - 1$ XOR operations to only one, for $n \leq 7$.

Finally, we obtain the following lemma:

**Lemma 3.** *Let $n$ be the dimension of squares matrices, $n \leq 6$, then the complexity of the determination of $GL(n, 2)$, using tabulation and Gray codes, is bounded by $D_n$ that satisfies:*

$$D_n = 2^{n^2} + O(2^{n^2-n-2}) \tag{4}$$

*Proof.* The complexity of the above algorithm is obtained by counting XOR operations and is given by the following recurrence formula:

$$\begin{cases} D_1 &= 0 \\ D_n &= D_{n-1} + 2^{n^2} - 1 \end{cases} \tag{5}$$

Therefore, we have:

$$\begin{aligned}
D_n &= \sum_{i=2}^{n} \left(2^{i^2} - 1\right) \\
&= \sum_{j=0}^{n-2} 2^{(n-j)^2} - (n-1) \\
&= 2^{n^2} \sum_{j=0}^{n-2} 2^{-2nj+j^2} - (n-1) \\
&\leq 2^{n^2} \sum_{j=0}^{n-2} 2^{-2nj+(n-2)j} - (n-1) \\
&= 2^{n^2} \sum_{j=0}^{n-2} 2^{(-n-2)j} - (n-1) \\
&= \frac{1-2^{(-n-2)(n-1)}}{1-2^{-n-2}} - (n-1) \\
&\leq 2^{n^2}(1 + \frac{1}{2^{n+2}-1}) - (n-1)
\end{aligned}$$

□

We thus have a gain of a factor $\frac{2}{3}n^3$ over the naive Gaussian elimination.

### 3.2 Optimization of matrix multiplication by tabulation

Similarly, we can optimize matrix multiplication with some tabulations. A first step consists in computing and storing all possible products between all values of the first row of the left matrix, and half the right one. Then, the matrix product $A\dot{B}$ will simply be obtained by two table accesses per row of $A$ (one for each half of $B$). Therefore, for computing $k$ products, we obtain a complexity bound of $O(2^{n+n\lceil\frac{n}{2}\rceil}(n-1)+k2n)$. As comparison, if we used a scalars products and transposition algorithms, we would have a complexity of $O(kn^2(n-1))$. As a conclusion, we see that our optimization are more efficient only if $k > \frac{2^{n+n\lceil\frac{n}{2}\rceil}}{n^2+\frac{2n}{n-1}}$. For semifields generation, the number of equivalence test is of the order of $2^{n^2}$ and the second term of the first complexity is therefore dominant. In this case our optimization allows to gain a factor of $n^2$ in the complexity bound.

## 4 S-boxes criterion

Several criteria have been defined to measure s-boxes resistance when faced to different types of attacks. In order to select our s-boxes, we have chosen the following criteria, following mostly [2]. We denote by $S$ the s-box function.

1. Bijectivity. By construction we only look for bijective functions.
2. Fixed Points. We favor functions without any fix points nor reverse fix points (as for the AES, this can be avoided by some affine transform on the trial).

3. Non-linearity. We return the linear invariant $\lambda_S$, defined as following:

$$\lambda_S = \max_{a,b \in \mathbb{F}_{2^n}, b \neq 0} \{| - 2^{n-1} + \#\{x \in \mathbb{F}_{2^n} : (a|x) \oplus (b|S(x)) = 0\}|\}$$

4. XOR table and differential invariant. A XOR table of S is based on the computation of $\delta_S(a,b) = \{x \in \mathbb{F}_{2^n} : S(x) \oplus S(a \oplus x) = b\}, \forall a, b \in \mathbb{F}_2^n$. The differential invariant $\delta_S$ is equal to $\max_{a,b \in \mathbb{F}_2^n, a \neq 0} \{\delta_S(a,b)\}$.

5. Avalanche. Strict avalanche criterion of order $k$ ($SAC(k)$) requires that the function $x \mapsto S(x) \oplus S(a \oplus x)$ stays balanced for all $a \in \mathbb{F}_{2^n}$ of weight inferior to $k$. The goal is to provide a $1/2$ probability of outputs modifications in case of $k$ bits complemented for entries. In our case, we measure the distance of the s-box to $SAC(1)$, component function by component function, and we denote by $A_S = max_{i=1..8}|2^{n-1} - \sum_x S_i(x) \oplus S_i(a \oplus x)|$ the maximum obtained.

6. Bit independance. Bit independancy is modelized by the computation of SAC(1) on the function defined by the sum of any two columns or any column of the matrix representation of S. As previously, we then measure its distance to SAC(1), and we denote it by $B_S$.

7. Transparency. This notion has been introduced by Prouff in [12], and allows to measure the resistance of s-boxes against differential power analysis. The definition is the following:

$$T_S = \max_{\beta \in \mathbb{F}_2^n} \Bigg( |n - 2H(\beta)| - \frac{1}{2^n(2^n-1)} \sum_{a \in \mathbb{F}_2^n} \Big| \sum_{v \in \mathbb{F}_2^n, H(v)=1} (-1)^{v\beta} W_{D_a,S}(0,v) \Big| \Bigg)$$

where $W_{D_a,S}(u,v) = \sum_{x \in \mathbb{F}_2^n} (-1)^{v[S(x)+S(x+a)]+ux}$ and $H(x)$ is the hamming weight.

By using this criteria, we are able to compare the efficiency of our s-boxes with the already existing ones.

For instance, the s-boxes of AES and Camellia have minimal non-linearity $\lambda_{AES} = \lambda_{Camellia} = 16$, minimal differential invariant among non-APN functions, $\delta_{AES} = \delta_{Camellia} = 4$, very good bit independence $B_{AES} = B_{Camellia} = 8$ and avalanche criterion with Camellia slightly better on the latter: $A_{AES} = 8$ and $A_{Camellia} = 6$.

# 5 Results of s-boxes based on semifields pseudo-extensions

We have implemented a simple matrix arithmetic using our optimizations, in order to generate semifields of order 16 plus their pseudo-extensions. We have

then constructed s-boxes with the help of the pseudo-inverse bijection, and apply all the tests of Section 4.

We succeed to generate 19336 semifields of order $2^4$ (with possible isomorphic ones). By testing all possible pseudo-irreducible polynomials for each semidfield, we obtained 10827 s-boxes, and 7052 were CCZ inequivalent [5], with maximal nonlinearity, differential invariants, degrees and bit interdependency.

Among the latter 2963 had fix points, and among the ones without fix points, 3846 had avalanche=8 (as good as AES) and 243 had avalanche=6 (as good as Camellia). Among them, 232 have a better transparency level than the inverse function on a finite field.

# 6 Conclusion

In order to construct new efficient $8 \times 8$ bijective s-boxes, we replace the usual finite fields algebraic structure by semifields. However, our current knowledge about this subject does not allow us to construct directly $\mathbb{S}_{2^8}$. We therefore build pseudo-extensions of degree 2 of $\mathbb{S}_{2^4}$. Pseudo-extensions are based on the notion of pseudo-irreducibility, derived from a characterisation of polynomial irreducibility in finite fields. This allows us to define in the product of semifields, a novel function as close as possible to the inverse function in a finite field. We call it a pseudo-inverse and use it exclusively in the building of new s-boxes. Many of the obtained s-boxes have then very good evaluations on different criterion for cryptographic resitance. Indeed, we obtained 232 s-boxes with better scores those of already known s-boxes, including AES and Camellia.

About bijective s-boxes, future search could be base on associativity variations of (3). It could also be interesting to try to adapt to semifields other functions (bijective or not), as the ones described in [2, §6]. It would thus also be worth to see if APN constructions over finite fields could be similarly adpated over semifields.

# References


[1] Abraham Adrian Albert. Finite division algebras and finite planes. In *Proceeding of Symposia in Applied Mathematics*, volume 10, pages 53–70, 1960.

[2] Rafael Alvarez and Gary McGuire. S-Boxes, APN functions and related codes. In Bart Preneel, Stefan Dodunekov, Vincent Rijmen, and Svetla Nikova, editors, *Enhancing Cryptographic Primitives with Techniques from Error Correcting CodesSoftware Agents, Agent Systems and Their Applications*, volume 23 of *NATO Science for Peace and Security Series - D: Information and Communication Security*, pages 49–62. IOS Press, 2008.

[3] Kazumaro Aoki, Tetsuya Ichikawa, Masayuki Kanda, Mitsuru Matsui, Shiho Moriai, Junko Nakajima, and Toshio Tokita. Camellia: A 128-bit

block cipher suitable for multiple platforms - design and analysis. In Douglas R. Stinson and Stafford E. Tavares, editors, *Selected Areas in Cryptography*, volume 2012 of *Lecture Notes in Computer Science*, pages 39–56. Springer, 2000. http://dx.doi.org/10.1007/3-540-44983-3_4.

[4] Eli Biham and Adi Shamir. Differential cryptanalysis of DES-like cryptosystems. *Journal of Cryptology*, 4(1):3–72, 1991.

[5] Lilya Budaghyan, Claude Carlet, and Alexander Pott. New classes of almost bent and almost perfect nonlinear polynomials. *IEEE Transactions on Information Theory*, 52(3):1141–1152, 2006. http://dx.doi.org/10.1109/TIT.2005.864481.

[6] Elías F. Combarro, I. F. Rúa, and J. Ranilla. New advances in the computational exploration of semifields. *International Journal of Computer Mathematics*, 88(9):1990–2000, 2011. http://dx.doi.org/10.1080/00207160.2010.548518, arXiv:http://dx.doi.org/10.1080/00207160.2010.548518.

[7] Joan Daemen and Vincent Rijmen. The block cipher rijndael. In Jean-Jacques Quisquater and Bruce Schneier, editors, *CARDIS*, volume 1820 of *Lecture Notes in Computer Science*, pages 277–284. Springer, 1998. http://dx.doi.org/10.1007/10721064_26.

[8] Vincent Danjean, Roland Gillard, Serge Guelton, Jean-Louis Roch, and Thomas Roche. Adaptive loops with kaapi on multicore and grid: applications in symmetric cryptography. In *Proceedings of the 2007 international workshop on Parallel symbolic computation*, PASCO '07, pages 33–42, New York, NY, USA, 2007. ACM. http://doi.acm.org/10.1145/1278177.1278185.

[9] Iqtadar Hussain, Tariq Shah, and MuhammadAsif Gondal. A novel approach for designing substitution-boxes based on nonlinear chaotic algorithm. *Nonlinear Dynamics*, 70(3):1791–1794, 2012. http://dx.doi.org/10.1007/s11071-012-0573-1.

[10] Lars R. Knudsen. Truncated and higher order differentials. In Bart Preneel, editor, *Fast Software Encryption: Second International Workshop*, volume 1008 of *Lecture Notes in Computer Science*, pages 196–211, Leuven, Belgium, 14–16 December 1994. Springer-Verlag.

[11] Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In Tor Helleseth, editor, *Advances in Cryptology—EUROCRYPT 93*, volume 765 of *Lecture Notes in Computer Science*, pages 386–397. Springer-Verlag, 1994, 23–27 May 1993.

[12] Emmanuel Prouff. DPA attacks and S-boxes. In Henri Gilbert and Helena Handschuh, editors, *Fast Software Encryption: 12th International Workshop, FSE 2005, Paris, France, February 21-23, 2005, Revised Selected Pa-*

*pers*, volume 3557 of *Lecture Notes in Computer Science*, pages 424–441. Springer, 2005. http://dx.doi.org/10.1007/11502760_29.

[13] I.F. Rúa, Elías F. Combarro, and J. Ranilla. Classification of semifields of order 64. *Journal of Algebra*, 322(11):4011 – 4029, 2009. http://www.sciencedirect.com/science/article/pii/S0021869309001343.