On Linear Redundancy in the AES S-Box

Joanne Fuller and William Millan
Information Security Research Centre
Queensland University of Technology
GPO Box 2434, Brisbane, Queensland, Australia 4001
FAX: 61-7-3221 2384
Email: {fuller,millan}@isrc.qut.edu.au

Abstract

We show the existence of a previously unknown linear redundancy property of the only nonlinear component of the AES block cipher. It is demonstrated that the outputs of the 8*8 Rijndael s-box (based on inversion in a finite field) are all equivalent under affine transformation. The method used to discover these affine relations is novel and exploits a new fundamental result on the invariance properties of local connection structure of affine equivalence classes. As well as increasing existing concerns about the security of the AES, these results may also have serious consequences for many other ciphers recently proposed for standardisation.

1 Introduction

The design of block ciphers has concentrated attention on the linear and differential properties of the nonlinear substitution boxes (s-boxes) that are used. Some interesting properties of $f(x) = x^{-1}$ in a finite field with characteristic 2 became well known following the Eurocrypt'93 paper by Nyberg[8]. For $n = 8$ inputs, the inversion mapping satisfies several nonlinearity criteria simultaneously, and for this reason it has become a cornerstone of modern symmetric cipher design. Good s-boxes allow ciphers to be designed provably secure against the powerful attacks differential cryptanalysis[2] and linear cryptanalysis[6], while maintaining an efficient implementation. The AES candidate Rijndael[3] uses the 8*8 inversion mapping 160 times, and has no other source of nonlinearity, so any weakness discovered in this mapping could have profound consequences for the overall security of the cipher.

Equivalence classes provide a powerful tool in both the construction and analysis of Boolean functions for cryptography. In particular, rather than considering the entire Boolean search space of 2^m functions, a reduced view can be found in the consideration of only one function from each class. To date however, this area of research has not been fully exploited and much is yet to be established regarding both a full characterisation of equivalence classes in relation to their cryptographic properties, as well as determination of the number of equivalence classes in existence for functions of more than five input variables. Another problem of particular importance in this area is that of determining a fast technique for determining whether two functions are equivalent when m is large.

In this paper, we show that all the output functions of the AES s-box can be mapped to each other using affine transformations, and hence they are all in the same affine equivalence class. Let $b_i(x)$ and $b_j(x)$ be two distinct outputs from the AES s-box, then there exists a non-singular matrix D_{ij} and a binary constant c_{ij} such that $b_j(x) = b_i(D_{ij}x) \oplus c_{ij}$. Examples of the D_{ij} matrices for the AES s-box are presented in Appendix C. In general there exist several such matrices. This property is a significant linear redundancy in the only nonlinear component of the declared AES, and it was not known before. The method used to discover the matrix is novel, very efficient, and exploits a new result on the local structure of Boolean functions and their equivalence classes. This equivalence property may lead to new attacks on the AES.

2 Background Theory

In this section background material on Boolean functions is presented. A Boolean function is a mapping from m binary inputs to one binary output, denoted $f(x) : \mathbf{Z}_2^m \rightarrow \mathbf{Z}_2$. We let $\mathcal{B}_m$ represent the set of all 2^{2^m} Boolean functions of m variables.

Truth Tables: The most direct representation of a Boolean function is as a truth table of 2^m bits. The truth table lists explicitly the output value, $f(x)$, for all m-bit input vectors $x = (x_1, x_2, \cdots, x_m)$, thus providing a unique representation. Alternatively, a Boolean function may be represented over $\{-1, 1\}$ in what is known as the polarity truth table, distinguished using the "hat" notation: $\hat{f}(x)$.

Hamming Weight: The *Hamming weight* of a Boolean function is defined to be the number of 1s in the binary truth table, or the number of -1s in the polarity truth table: $\mathbf{wt}(f) = \sum_x f(x) = \frac{1}{2}\left(2^m - \sum_x \hat{f}(x)\right)$. A function is considered to be *balanced* when half of the function values are equal to one: $\mathbf{wt}(f) = 2^{m-1}$ or alternatively, $\mathbf{wt}(\hat{f}) = 0$.

Hamming Distance: The *Hamming distance* between two functions $f \in \mathcal{B}_m$ and $g \in \mathcal{B}_m$ is defined as the number of truth table positions in which the functions differ and can be expressed as the Hamming weight of the `XOR` sum of two functions: $\mathbf{dist}(f, g) = \mathbf{wt}(f \oplus g)$.

Algebraic Normal Form: The *algebraic normal form* expresses a Boolean function as a sum modulo 2 of the 2^m possible products of the m inputs:

$$f(x_1, x_2, ..., x_m) = a_0 \bigoplus_{1 \leq i \leq m} a_i x_i \oplus \bigoplus_{1 \leq i \leq j \leq m} a_{ij} x_i x_j \oplus ... \oplus a_{12...m} x_1 x_2 ... x_m.$$

Algebraic Order: The *algebraic order* (or simply *order*), denoted $\mathbf{ord}(f)$, is defined to be the size of the largest product term used in the algebraic normal form of the function f.

Linear and Affine Functions: A *linear* function is defined as the `XOR` sum of a subset of the input variables, denoted $\mathcal{L}_\omega(x) = \omega_1 x_1 \oplus \omega_2 x_2 \oplus ... \oplus \omega_m x_m \oplus$ where $\omega \in \mathbf{Z}_2^m$. The set of *affine functions* are the linear functions and their complements, denoted $\mathcal{A}_{\omega,c}(x) = \mathcal{L}_\omega(x) \oplus c$.

Walsh-Hadamard Transform: The Walsh-Hadamard transform uniquely expresses a Boolean function in terms of its correlation to all linear functions. Denoted by $\hat{\mathcal{F}}(\omega)$, the Walsh-Hadamard transform is calculated from the polarity truth table as $\hat{\mathcal{F}}(\omega) = \sum_x (-1)^{\hat{f}(x)} \cdot (-1)^{\hat{\mathcal{L}}_\omega(x)}$, $\omega \in \mathbf{Z}_2^m$.

Nonlinearity: The *nonlinearity* of a Boolean function is defined as the maximum Hamming distance to the set of affine functions. Nonlinearity is given directly by observing $|\hat{\mathcal{F}}_{max}|$, the maximum absolute value occurring in $\hat{\mathcal{F}}(\omega)$, and calculated as $\mathcal{N}(f) = \frac{1}{2}(2^m - |\hat{\mathcal{F}}_{max}|)$.

Autocorrelation: The *autocorrelation function* gives an indication of the imbalance of all first order derivatives of a Boolean function. The derivative of a Boolean function $\hat{f}(x)$, taken with respect to a vector s is defined as

$\mathcal{D}_s f(x) = f(x) \oplus f(x \oplus s)$. The autocorrelation function of a Boolean function is defined as $\hat{\mathcal{R}}(s) = \sum_x \hat{f}(x)\hat{f}(x \oplus s)$.

2.1 Affine Transforms

An affine transform is defined in terms of the combination of a linear transform and a dyadic shift ($\oplus a$) on a function. A linear transform involves the multiplication of the input vector of a Boolean function by a non-singular binary matix $\mathcal{D}$. The addition of an affine function to the output of a Boolean function is also an affine transformation. Mathematically, we therefore express the affine transform using $\mathcal{D}$($m \times m$ invertible matrix), $(a, b) \in \mathbf{Z}_2^m$ and $c \in \mathbf{Z}_2$ as

$$g(x) = f(\mathcal{D}x \oplus a) \oplus b \cdot x \oplus c.$$

Thus, two Boolean functions may be considered equivalent if they are related by an affine transform.

Not much has been said about equivalence classes since the 1972 paper[1] on $m = 5$ described all 48 classes in terms of their algebraic normal form. It seems to be well known that the number of equivalence classes increases exponentially with m, for example see [4]. Concretely, the 1991 Maiorana paper[5] states that there exist 150,357 classes for $m = 6$, including 2082 different Walsh-Hadamard transform distributions, but there is no analysis of structure for cryptology. More recently, equivalence classes have been used to provide restricted inputs to random and heuristic searches seeking better Boolean functions[9].

Until now, the only known way to determine if two functions are affine equivalent has been to conduct exhaustive search on the set of affine transforms. This seems infeasible for moderate m, due to a matrix having m^2 bits. In fact it has been an open problem to demonstrate any other algorithm for distinguishing affine equivalence.

One approach to this problem is suggested by the observation (due perhaps to Preneel[10]) that the absolute values in the Walsh-Hadamard transform and the autocorrelation function are always re-arranged by affine transforms, so the frequency distribution of the absolute values in these transforms is invariant under affine transform. Investigating this, we have found by inspection that this pair of [value, frequency] distributions is sufficient to distinguish all 48 classes at $m = 5$. A stack based local search heuristic finds all 48 classes easily and very quickly. We note with interest that this novel WHT/AC transform analysis approach is much, much faster than searching over all the possible 5-bit affine transformations. However when applied to $m = 6$, these heuristic searches have revealed the existence of less than 31,000 different WHT/AC value frequency distributions. Compared to the 150,357 classes claimed to have been found by Maiorana[5], it is clear that the WHT/AC distributions are not sufficient to distinguish classes at $m = 6$ or higher. Clearly other methods are required.

In seeking other methods to approach the class distinguishing problem, we investigated the local structure by considering the set of functions at distance one from a given Boolean function.

3 Connectivity

The concept of local structure has been introduced in previous research[7] that examined the application of heuristic techniques to the discovery of Boolean functions possessing specific properties. This idea of examining the Boolean functions generated by any one position change to a function f is the basis for the definition of *connectivity.*

Definition 1. The *1-local neighbourhood* of a Boolean function $f \in \mathcal{B}_m$ consists of all 2^m Boolean functions $f_{(i)} \in \mathcal{B}_m$, $i \in \mathbf{Z}_2^m$, constructed such that `dist`(f, $f_{(i)}$)=1. We say that $f_{(i)}$ is *connected to* f. This can be expressed as,

$$f_{(i)}(x) = \begin{cases} f(x) & x \neq i \\ f(x) \oplus 1 & x = i \end{cases}$$

The ability to analyse the effect of this small modification in relation to the resulting functions transforms suggests the investigation of the action of affine transforms on the local neighbourhood. Specifically, we can determine that if f and g are equivalent, then there exists a function $g_{(j)}$ at Hamming distance one from g, that is equivalent to a function $f_{(i)}$ at Hamming distance one from f under the same affine transform relating f and g.

Theorem 1. Let $f_{(i)}$, defined as above, then there exists a connecting function $g_{(j)}$ of $g(x) = f(\mathcal{D}x \oplus a) \oplus b \cdot x \oplus c$ such that $g_{(j)}(x) = f_{(i)}(\mathcal{D}x \oplus a) \oplus b \cdot x \oplus c$ and $j = \mathcal{D}^{-1}(i \oplus a)$.

Proof. Let $g(x) = f(\mathcal{D}x \oplus a) \oplus b \cdot x \oplus c$ and

$$f_i(x) = \begin{cases} f(x) & x \neq i \\ f(x) \oplus 1 & x = i \end{cases}$$

Therefore,

$$f_{(i)}(\mathcal{D}x \oplus a) \oplus b \cdot x \oplus c = \begin{cases} f(\mathcal{D}x \oplus a) \oplus b \cdot x \oplus c & (\mathcal{D}x \oplus a) \neq i \\ f(\mathcal{D}x \oplus a) \oplus b \cdot x \oplus c \oplus 1 & (\mathcal{D}x \oplus a) = i \end{cases}$$

$$f_{(i)}(\mathcal{D}x \oplus a) \oplus b \cdot x \oplus c = \begin{cases} g(x) & x \neq \mathcal{D}^{-1}(i \oplus a) = j \\ g(x) \oplus 1 & x = \mathcal{D}^{-1}(i \oplus a) = j \end{cases}$$

And hence, $f_{(i)}(\mathcal{D}x \oplus a) \oplus b \cdot x \oplus c$ is equivalent to $g_{(j)}$, a neighbour function of g such that $j = \mathcal{D}^{-1}(i \oplus a)$.

Corollary 1. The 1-local neighbourhood of g is a permutation of the 1-local neighbourhood of f.

Proof. This follows directly from the non-singularity of $\mathcal{D}$.

We use these results in Section 4.

3.1 Experimental Results

For Boolean functions of $m \leq 5$ we can exhaustively examine function connectivity, including the frequency distribution of equivalence classes for equivalent functions. Specifically, an experiment was conducted in which for all functions of a given equivalence class, the distribution of equivalence classes of the 2^m connecting functions was recorded. The results of this experiment also show that for functions of $m = 4$ and $m = 5$, the equivalence class frequency distribution of the connectivity distribution of equivalent functions is unique. The following tables provide the results of these exhaustive experiments. For each class the table shows all other classes in the 1-local neighbourhood. For each of these connected classes, the number in brackets indicates the number of distinct 1-bit changes that lead to that class. Refer to Appendix A for specification of the class numbers.

Table 1: Equivalence Class Connectivity Distribution, $m = 3$

Class	*Class(frequency)* Distribution
1	C2(8)
2	C1(1), C3(7)
3	C2(8)

Table 2: Equivalence Class Connectivity Distribution, $m = 4$

Class	*Class(frequency)* Distribution
1	C2(16)
2	C1(1), C3(15)
3	C2(2), C4(14)
4	C3(3), C5(1), C6(12)
5	C4(16)
6	C4(8), C7(8)
7	C6(15), C8(1)
8	C7(16)

Table 3: Equivalence Class Connectivity Distribution, $m = 5$

Class	*Class(frequency)* Distribution
1	C2(32)
2	C1(1), C3(31)
3	C2(2), C4(30)
4	C3(3), C5(1), C6(28)
5	C4(4), C8(28)
6	C4(4), C7(24), C8(4)
7	C6(5), C9(1), C10(16), C11(10)
8	C5(1), C6(4), C11(24), C12(3)
9	C7(6), C13(16), C14(10)
10	C7(6), C13(6), C15(20)
11	C7(4), C8(2), C14(4), C15(16), C16(6)
12	C8(6), C16(24), C17(2)
13	C9(1), C10(6), C21(15), C22(10)
14	C9(1), C11(6), C22(16), C23(9)
15	C10(4), C11(3), C19(1), C21(3), C22(12), C24(9)
16	C11(6), C12(1), C20(1), C23(6), C24(16), C25(2)
17	C12(7), C18(1), C25(24)
18	C17(32)
19	C15(8), C29(24)
20	C16(16), C31(16)
21	C13(4), C15(4), C26(12), C27(8), C29(4)
22	C13(1), C14(1), C15(6), C26(9), C28(9), C29(6)
23	C14(8), C16(8), C28(16)
24	C15(6), C16(2), C27(2), C28(12), C29(6), C30(2), C31(2)

Table 3: Equivalence Class Connectivity Distribution, $m = 5$

Class	*Class(frequency)* Distribution
25	C16(14), C17(2), C30(16)
26	C21(3), C22(6), C32(1), C34(4), C35(12), C37(6)
27	C21(6), C24(3), C33(1), C35(18), C36(3), C38(1)
28	C22(8), C23(1), C24(8), C32(1), C35(4), C37(8), C39(2)
29	C19(1), C21(1), C22(4), C24(3), C35(6), C37(12), C38(2), C39(3)
30	C24(21), C25(3), C33(1), C38(7)
31	C20(1), C24(16), C36(2), C39(12), C40(1)
32	C26(16), C28(12), C41(4)
33	C27(28), C30(4)
34	C26(10), C41(10), C42(12)
35	C26(8), C27(4), C28(2), C29(4), C41(2), C42(8), C43(4)
36	C27(16), C31(2), C43(12), C44(2)
37	C26(6), C28(6), C29(12), C42(8)
38	C27(4), C29(24), C30(4)
39	C28(8), C29(16), C31(4), C43(4)
40	C31(30), C44(2)
41	C32(1), C34(16), C35(12), C48(3)
42	C34(6), C35(15), C37(10), C46(1)
43	C35(24), C36(3), C39(3), C47(1), C48(1)
44	C36(30), C40(1), C45(1)
45	C44(32)
46	C42(32)
47	C43(32)
48	C41(24), C43(8)

4 The AES S-box Functions

The existence of a technique to distinguish equivalent functions conclusively and indeed determine the specific affine transform relationship between equivalent functions would be of great benefit to the study of Boolean functions, particularly with regard to their use in cryptography. As equivalent functions are essentially the same functions, the use of equivalent functions as outputs of an s-box may provide an exploitable weakness. The eight Boolean functions of the s-box used in the AES cipher, Rijndael, are an example of this problem. The eight functions exhibit identical properties of order, nonlinearity and the absolute maximum value of the autocorrelation function. As well, these eight functions share the same absolute $\hat{\mathcal{F}}$ and absolute $\hat{\mathcal{R}}$ frequency distributions, suggesting that they are indeed all equivalent. However, to date, the question of whether the Rijndael s-box functions are equivalent has remained an open problem due to our inability to determine the exact affine transform relating the functions. In this section we will solve this problem by identifying the affine transforms that exist between the Rijndael s-box functions.

In general, for Boolean functions of $m \leq 5$, equivalent functions can easily be identified due to the unique combination of the absolute $\hat{\mathcal{F}}$ and absolute $\hat{\mathcal{R}}$ frequency distributions possessed by each equivalence class. For Boolean functions of $m \geq 6$, the only known technique for determining the affine transform relating two equivalent functions when $\mathcal{D}$ is not the identity matrix, is an exhaustive search. Such a search would be of magnitude $\mathrm{O}(2^m)^m$, which is becoming infeasible with increasing m. Certainly it is infeasible for $m \geq 8$. The theorem of connectivity tells us that the connecting functions of $f(x) \in \mathcal{B}_m$ and those of $g(x) = f(\mathcal{D}x \oplus a) \oplus b \cdot x \oplus c \in \mathcal{B}_m$ share the same equivalence mapping as f and g. Hence, rather than only two equivalent functions, we in fact have $2^m + 1$ pairs of equivalent functions under the same affine transform. We believe that, in general, this will provide sufficient data to uniquely determine $\mathcal{D}(m \times m$ invertible matrix), $(a, b) \in \mathbf{Z}_2^m$ and $c \in \mathbf{Z}_2$.

To illustrate the technique to be used, let us consider the eight AES s-box functions, specified as $b_i = \mathtt{SBox}[x \& (1 << (i-1))]$ where $b_i \in \mathcal{B}_8$, so that b_1 therefore represents the function taken from the least significant bit of each s-box value. Refer to Appendix B for a listing of SBox. More specifically, let us first consider functions b_1 and b_2. From Theorem 1, we know that connecting function i of b_1 will be equivalent to connecting function $j = \mathcal{D}_{12}^{-1}(i \oplus a)$ of b_2 and therefore $i = \mathcal{D}_{12} j \oplus a$. Note that x is a column vector where $x = (x_1, x_2, ..., x_m)^T$ and x_m is the least significant bit of x.

Step 1: Finding a

When $j = 0$ we know that $i = a$. Thus, if we can determine the connecting function $b_{2(j)}$ equivalent to connecting function $b_{1(0)}$, then a will have also been determined. When $b_{1(0)}$ was examined, it was found to have $|\hat{\mathcal{F}}|$ distribution:

$$\{(2,33),(6,52),(10,32),(14,40),(18,34),(22,28),(26,24),(30,13)\}$$

and $|\hat{\mathcal{R}}|$ distribution:

$$\{(4,84),(12,72),(20,62),(28,37),(256,1)\}$$

The only connecting function of b_2 to have the same distributions, and therefore the only connecting function that could possibly be equivalent to $b_{1(0)}$, was $b_{2(0)}$. Hence, $a = 0$.

Step 2: Finding $\mathcal{D}_{12}$

As a result of Step 1, we now can say that connecting function i of b_1 will be equivalent to connecting function $j = \mathcal{D}_{12}^{-1}(i)$ of b_2, and therefore $i = \mathcal{D}_{12}j$. Furthermore, when $j = \mathbf{e}_k$ such that $\mathbf{e}_k$ be the unit vector with 1 is position k and 0 elsewhere, we see that i^T will be the k^{th} column of $\mathcal{D}_{12}$. Thus, if we can determine which $b_{1(i)}$ are equivalent to $b_{2(j)}$ when $j = \mathbf{e}_k$ ($\forall k \leq m$), then we will have found the columns of $\mathcal{D}_{12}$ and essentially solved the equivalence problem.

Unlike in Step 1, the m connecting functions $b_{2(j)}$ ($j = \mathbf{e}_k$) each could possibly map to one of eight connecting functions of b_1. While this does mean that some further analysis is required, the search space for $\mathcal{D}$ has effectively been reduced to m^m, which for $m = 8$ is certainly feasible. The following table summarises the possible mappings between i and j.

j	column of $\mathcal{D}_{12}$	possible valid i
1	8	1,25,152,161,199,207,208,211
2	7	41,50,68,91,97,117,228,244
4	6	44,63,77,89,100,123,159,173
8	5	26,29,74,127,144,182,200,246
16	4	43,51,139,178,229,237,249,250
32	3	9,13,112,119,128,143,204,234
64	2	26,29,74,127,144,182,200,246
128	1	52,59,116,120,147,151,195,238

Using each combination of these potential values for the columns of $\mathcal{D}_{12}$, a computer experiment was conducted to determine which combination (if any) would provide a valid $\mathcal{D}_{12}$ matrix. Several valid $\mathcal{D}_{12}$ matrices were produced. The matrix which was selected is listed in the following step.

Step 3: Finding b and c

As a result of Step 2, the only two remaining unknown variables of the transform relating b_1 and b_2 are $b \in \mathbf{Z}_2^m$ and $c \in \mathbf{Z}_2$. It was a simple task at this stage to simply apply $\mathcal{D}_{12}$ to b_1 and then determine which affine function addition was required to generate b_2. In fact the task was a simple one, as the application of a transform involving only $\mathcal{D}_{12}$ was found to provide the complete affine transform of b_1 to b_2. Hence, we can say:

$$b_2(x) = b_1(\mathcal{D}_{12}x)$$

where

$$\mathcal{D}_{12} = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

Step 4: Other S-box Function Equivalence Relationships

Steps 1 to 4 can be applied to each of the remaining six s-box functions, $b_3 ... b_8$, to determine their equivalence relationship with b_1. The results of which are given in Appendix C. The inverses of these $\mathcal{D}_{ij}$ matrices are also provided in Appendix D.

5 Conclusions and Future Research

It has been shown that the local structure of class connectivity remains invariant under all affine transforms. This observation finds application as a method to efficiently discover a non-singular matrix that defines an affine transform between two given Boolean functions. Should this algorithm fail, then the functions are not equivalent. The method has been applied to the output functions of the Rijndael s-box, revealing that all eight of these functions are in the same affine equivalence class. It follows directly that this kind of redundancy is shared by the operation of inversion in the finite field (for n=8 at least), and hence this weakness also exists in all other s-boxes based on this mathematical operation. We note that none of the other AES finalists use finite field inversion, so they are not affected by this result.

References

[1] E.R. Berlekamp and L.R. Welch. Weight Distributions of the Cosets of the $(32, 6)$ Reed-Muller Code. *IEEE Transactions on Information Theory*, 18(1):203–207, January 1972.

[2] E. Biham and A. Shamir. Differential cryptanalysis of DES-like cryptosystems. In *Advances in Cryptology - Crypto '90, Proceedings*, volume 537 of *Lecture Notes in Computer Science*, pages 2–21. Springer-Verlag, 1991.

[3] J. Daemen and V. Rijmen. Aes proposal: Rijndael. www.esat.kuleuven.ac.be/ rijmen/rijndael/.

[4] J.D. Denev and V.D. Tonchev. On the Number of Equivalence Classes of Boolean Functions under a Transformation Group. *IEEE Transactions on Information Theory*, 26(5):625–626, September 1980.

[5] J.A. Maiorana. A Classificationn of the Cosets of the Reed-Muller code $r(1, 6)$. *Mathematics of Computation*, 57(195):403–414, July 1991.

[6] M. Matsui. Linear Cryptanalysis Method for DES Cipher. In *Advances in Cryptology - Eurocrypt '93, Proceedings*, volume 765, pages 386–397. Springer-Verlag, 1994.

[7] W. Millan, A. Clark, and E. Dawson. Smart Hill Climbing Finds Better Boolean Functions. In *Workshop on Selected Areas in Cryptology 1997, Workshop Record*, pages 50–63, 1997.

[8] K. Nyberg. Differentially uniform mappings for cryptography. In *Advances in Cryptology - Eurocrypt '93, Proceedings*, volume 765 of *Lecture Notes in Computer Science*, pages 55–64. Springer-Verlag, 1994.

[9] E. Pasalic, T. Johansson, S. Maitra, and P. Sarkar. New constructions of resilient and correlation immune boolean functions achieving upper bounds on nonlinearity, 2001.

[10] B. Preneel. *Analysis and Design of Cryptographic Hash Functions.* PhD thesis, Cathoic University of Leuven, 1994.

Appendix A - Equivalence Classes Properties

Equivalence Class Properties, $m = 3$

class	*example*	*ord*	*nl*	*abs. WHT distribution*	*abs. AC distribution*
1	0xaa	1	0	{(0,7),(8,1)}	{(8,8)}
2	0xab	3	1	{(2,7),(6,1)}	{(4,7),(8,1)}
3	0xac	2	2	{(0,4),(4,4)}	{(0,6),(8,2)}

Equivalence Class Properties, $m = 4$

class	*example*	*ord*	*nl*	*abs. WHT distribution*	*abs. AC distribution*
1	0xaa55	1	0	{(0,15),(16,1)}	{(16,16)}
2	0xab55	4	1	{(2,5),(14,1)}	{(12,15),(16,1)}
3	0xbb55	3	2	{(0,8),(4,7),(12,1)}	{(8,14),(16,2)}
4	0xaba5	4	3	{(2,12),(6,3),(10,1)}	{(4,12),(12,3),(16,1)}
5	0xaaff	2	4	{(0,12),(8,4)}	{(0,12),(16,4)}
6	0xaba4	3	4	{(0,6),(4,8),(8,2)}	{(0,9),(8,6),(16,1)}
7	0xab12	4	5	{(2,10),(6,6)}	{(4,15),(16,1)}
8	0xac90	2	6	{(4,16)}	{(0,15),(16,1)}

Equivalence Class Properties, $m = 5$

class	*example*	*ord*	*nl*	*abs. WHT distribution*	*abs. AC distribution*
1	0xaa55aa55	1	0	(0,31),(32,1)	(32,32)
2	0xaa55ab55	5	1	(2,31),(30,1)	(28,31),(32,1)
3	0xaa55bb55	4	2	(0,16),(4,15),(28,1)	(24,30),(32,2)
4	0xaa5dbb55	5	3	(2,24),(6,7),(26,1)	(20,28),(28,3),(32,1)
5	0xaaddbb55	3	4	(0,24),(8,7),(24,1)	(16,28),(32,4)
6	0xaa5dbb51	4	4	(0,12),(4,16),(8,3),(24,1)	(16,25),(24,6),(32,1)
7	0x2a5dbb51	5	5	(2,20),(6,10),(10,1),(22,1)	(12,21),(20,10),(32,1)
8	0xaaddbb51	5	5	(2,24),(6,4),(10,3),(22,1)	(12,24),(20,4),(28,3),(32,1)
9	0x2a5dbf51	3	6	(4,30),(12,1),(20,1)	(8,16),(16,15),(32,1)
10	0x6a5dbb51	4	6	(0,10),(4,15),(8,6),(20,1)	(8,16),(16,15),(32,1)
11	0x2addbb51	4	6	(0,12),(4,14),(8,4),(12,1),(20,1)	(8,19),(16,9),(24,3),(32,1)
12	0xa8ddbb51	4	6	(0,16),(4,12),(12,3),(20,1)	(8,24),(24,6),(32,2)
13	0xaeddda51	5	7	(2,15),(6,15),(10,1),(18,1)	(4,10),(12,21),(32,1)
14	0x0a5dbf51	5	7	(2,18),(6,12),(14,1),(18,1)	(4,16),(12,9),(20,6),(32,1)

Equivalence Class Properties, $m = 5$

class	*example*	*ord*	*nl*	*abs. WHT distribution*	*abs. AC distribution*
15	0x8addda51	5	7	(2,19),(6,9),(10,3),(18,1)	(4,13),(12,15),(20,3),(32,1)
16	0xa8dd9b51	5	7	(2,22),(6,6),(10,2),(14,1),(18,1)	(4,18),(12,6),(20,6),(28,1),(32,1)
17	0x88ddbb51	5	7	(2,28),(14,3),(18,1)	(4,24),(28,7),(32,1)
18	0x88ddbb11	2	8	(0,28),(16,4)	(0,24),(32,8)
19	0x8c5dda51	3	8	(0,19),(8,12),(16,1)	(0,9),(8,16),(16,6),(32,1)
20	0xa89d9b51	3	8	(0,22),(8,8),(16,2)	(0,18),(16,12),(32,2)
21	0x8eddda51	4	8	(0,7),(4,16),(8,8),(16,1)	(0,6),(8,22),(16,3),(32,1)
22	0xaefdda51	4	8	(0,9),(4,15),(8,6),(12,1),(16,1)	(0,9),(8,16),(16,6),(32,1)
23	0x025dbf51	4	8	(0,10),(4,16),(8,4),(16,2)	(0,16),(8,4),(16,9),(24,2),(32,1)
24	0x88ddda51	4	8	(0,11),(4,14),(8,4),(12,2),(16,1)	(0,11),(8,13),(16,6),(24,1),(32,1)
25	0x88dd9b51	4	8	(0,14),(4,14),(12,2),(16,2)	(0,17),(8,7),(24,7),(32,1)
26	0xceddda51	5	9	(2,15),(6,13),(10,3),(14,1)	(4,22),(12,9),(32,1)
27	0x0eddda51	5	9	(2,15),(6,13),(10,3),(14,1)	(4,24),(12,6),(20,1),(32,1)
28	0x425dbf51	5	9	(2,18),(6,10),(10,2),(14,2)	(4,20),(12,9),(20,2),(32,1)
29	0x8cddda51	5	9	(2,19),(6,7),(10,5),(14,1)	(4,21),(12,9),(20,1),(32,1)
30	0x88dddb51	5	9	(2,21),(6,7),(10,1),(14,3)	(4,24),(20,7),(32,1)
31	0x289d9b51	5	9	(2,22),(6,4),(10,4),(14,2)	(4,18),(12,12),(28,1),(32,1)
32	0x86fdda51	3	10	(4,28),(12,4)	(0,12),(8,16),(16,3),(32,1)
33	0x88dddb71	3	10	(4,28),(12,4)	(0,24),(16,7),(32,1)
34	0xcefdda51	4	10	(0,6),(4,15),(8,10),(12,1)	(0,15),(8,16),(32,1)
35	0x0efdda51	4	10	(0,8),(4,14),(8,8),(12,2)	(0,15),(8,14),(16,2),(32,1)
36	0x288d9b51	4	10	(0,8),(4,14),(8,8),(12,2)	(0,17),(8,13),(24,1),(32,1)
37	0x8cfdda51	4	10	(0,10),(4,13),(8,6),(12,3)	(0,12),(8,16),(16,3),(32,1)
38	0x8cdddb51	4	10	(0,12),(4,12),(8,4),(12,4)	(0,18),(8,6),(16,7),(32,1)
39	0x8ccdda51	4	10	(0,12),(4,12),(8,4),(12,4)	(0,8),(8,21),(16,1),(24,1),(32,1)
40	0x289d9b41	4	10	(0,16),(4,10),(12,6)	(8,30),(32,2)
41	0x488ddb51	5	11	(2,12),(6,16),(10,4)	(4,28),(12,3),(32,1)
42	0xccfdda51	5	11	(2,16),(6,10),(10,6)	(4,25),(12,6),(32,1)
43	0x688d9b51	5	11	(2,16),(6,10),(10,6)	(4,27),(12,3),(20,1),(32,1)
44	0x288d9b41	5	11	(2,16),(6,10),(10,6)	(4,30),(28,1),(32,1)
45	0x288d1b41	2	12	(0,16),(8,16)	(0,30),(32,2)
46	0xdcfdda51	3	12	(0,16),(8,16)	(0,15),(8,16),(32,1)
47	0x68ad9b51	3	12	(0,16),(8,16)	(0,27),(16,4),(32,1)
48	0x688ddb51	4	12	(0,4),(4,16),(8,12)	(0,24),(8,6),(16,1),(32,1)

Appendix B - The AES S-box

```
SBox[256] = { 0x63, 0x7C, 0x77, 0x7B, 0xF2, 0x6B, 0x6F, 0xC5,
              0x30, 0x01, 0x67, 0x2B, 0xFE, 0xD7, 0xAB, 0x76,
              0xCA, 0x82, 0xC9, 0x7D, 0xFA, 0x59, 0x47, 0xF0,
              0xAD, 0xD4, 0xA2, 0xAF, 0x9C, 0xA4, 0x72, 0xC0,
              0xB7, 0xFD, 0x93, 0x26, 0x36, 0x3F, 0xF7, 0xCC,
              0x34, 0xA5, 0xE5, 0xF1, 0x71, 0xD8, 0x31, 0x15,
              0x04, 0xC7, 0x23, 0xC3, 0x18, 0x96, 0x05, 0x9A,
              0x07, 0x12, 0x80, 0xE2, 0xEB, 0x27, 0xB2, 0x75,
              0x09, 0x83, 0x2C, 0x1A, 0x1B, 0x6E, 0x5A, 0xA0,
              0x52, 0x3B, 0xD6, 0xB3, 0x29, 0xE3, 0x2F, 0x84,
              0x53, 0xD1, 0x00, 0xED, 0x20, 0xFC, 0xB1, 0x5B,
              0x6A, 0xCB, 0xBE, 0x39, 0x4A, 0x4C, 0x58, 0xCF,
              0xD0, 0xEF, 0xAA, 0xFB, 0x43, 0x4D, 0x33, 0x85,
              0x45, 0xF9, 0x02, 0x7F, 0x50, 0x3C, 0x9F, 0xA8,
              0x51, 0xA3, 0x40, 0x8F, 0x92, 0x9D, 0x38, 0xF5,
              0xBC, 0xB6, 0xDA, 0x21, 0x10, 0xFF, 0xF3, 0xD2,
              0xCD, 0x0C, 0x13, 0xEC, 0x5F, 0x97, 0x44, 0x17,
              0xC4, 0xA7, 0x7E, 0x3D, 0x64, 0x5D, 0x19, 0x73,
              0x60, 0x81, 0x4F, 0xDC, 0x22, 0x2A, 0x90, 0x88,
              0x46, 0xEE, 0xB8, 0x14, 0xDE, 0x5E, 0x0B, 0xDB,
              0xE0, 0x32, 0x3A, 0x0A, 0x49, 0x06, 0x24, 0x5C,
              0xC2, 0xD3, 0xAC, 0x62, 0x91, 0x95, 0xE4, 0x79,
              0xE7, 0xC8, 0x37, 0x6D, 0x8D, 0xD5, 0x4E, 0xA9,
              0x6C, 0x56, 0xF4, 0xEA, 0x65, 0x7A, 0xAE, 0x08,
              0xBA, 0x78, 0x25, 0x2E, 0x1C, 0xA6, 0xB4, 0xC6,
              0xE8, 0xDD, 0x74, 0x1F, 0x4B, 0xBD, 0x8B, 0x8A,
              0x70, 0x3E, 0xB5, 0x66, 0x48, 0x03, 0xF6, 0x0E,
              0x61, 0x35, 0x57, 0xB9, 0x86, 0xC1, 0x1D, 0x9E,
              0xE1, 0xF8, 0x98, 0x11, 0x69, 0xD9, 0x8E, 0x94,
              0x9B, 0x1E, 0x87, 0xE9, 0xCE, 0x55, 0x28, 0xDF,
              0x8C, 0xA1, 0x89, 0x0D, 0xBF, 0xE6, 0x42, 0x68,
              0x41, 0x99, 0x2D, 0x0F, 0xB0, 0x54, 0xBB, 0x16}
```

Appendix C - AES S-box Equivalence Relationships

$$b_3(x) = b_1(\mathcal{D}_{13}x) \oplus 1 \text{ where } \mathcal{D}_{13} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$b_4(x) = b_1(\mathcal{D}_{14}x) \oplus 1 \text{ where } \mathcal{D}_{14} = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$b_5(x) = b_1(\mathcal{D}_{15}x) \oplus 1 \text{ where } \mathcal{D}_{15} = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

$$b_6(x) = b_1(\mathcal{D}_{16}x) \text{ where } \mathcal{D}_{16} = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

$$b_7(x) = b_1(\mathcal{D}_{17}x) \text{ where } \mathcal{D}_{17} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}$$

$$b_8(x) = b_1(\mathcal{D}_{18}x) \oplus 1 \text{ where } \mathcal{D}_{18} = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Appendix D - Inverse Equivalence Relationships

$$\mathcal{D}_{21} = \mathcal{D}_{12}^{-1} = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

$$\mathcal{D}_{31} = \mathcal{D}_{13}^{-1} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathcal{D}_{41} = \mathcal{D}_{14}^{-1} = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix}$$

$$\mathcal{D}_{51} = \mathcal{D}_{15}^{-1} = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

$$\mathcal{D}_{61} = \mathcal{D}_{16}^{-1} = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\mathcal{D}_{71} = \mathcal{D}_{17}^{-1} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}$$

$$\mathcal{D}_{81} = \mathcal{D}_{18}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$