Calculating Cryptographic Degree of an S-Box

Prasanna Raghaw Mishra

February 26, 2014

Abstract

In this paper we propose an efficient technique to compute algebraic degree of an S-box (minimum of algebraic degrees of its component functions). Using our technique we have calculated algebraic degree of a 26×64 S-box.

1 Introduction:

We define cryptographic degree of an S-box as the least of the degrees of its component functions in their Algebraic Normal Forms. Here we present an efficient algorithm for calculation of cryptographic degree of S-boxes. Before discussing the algorithm, we discuss some of the pre-requisites.

1.1 Algebraic Normal Form:

An n-variable Boolean function may be represented as a multivariate polynomial over $\mathbb{F}_2$.

$$f(x_1, x_2, \ldots, x_n) = \bigoplus_{I \in \mathcal{P}} a_I \left(\prod_{i \in I} x^i \right) = \bigoplus_{I \in \mathcal{P}} a_I x^I$$

where $\mathcal{P}(N)$ denotes power set of $N = \{1, 2, \ldots, n\}$. This representation in fact belongs to $\mathbb{F}_2[x_1, x_2, \ldots, x_n] / \left(x_1^2 \oplus x_1, x_2^2 \oplus x_2, \ldots, x_n^2 \oplus x_n \right)$. We define a map $Supp : \mathbb{F}_2^n \to \mathcal{P}(N)$ as for a given $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{F}_2^n,\ i \in Supp(\mathbf{x})$ if and only if $x_i = 1$. We call $Supp(\mathbf{x})$ as support of vector $\mathbf{x}$.This map is a one-to-one correspondance between $\mathbb{F}_2^n$and $\mathcal{P}(N)$. Using this map, we can reindex our formula as

$$f(x_1, x_2, \ldots, x_n) = \bigoplus_{u \in \mathbb{F}_2^n} a_u \left(\prod_{j=1}^{n} x_j^{u_j} \right)$$

We, now, associate a Boolean function f^0as $f^0(\mathbf{x}) = a_{\mathbf{x}}$. It can be shown that $f^0(\mathbf{x})$ can be directly calculated from f as

$$f^0(\mathbf{x}) = \bigoplus_{\mathbf{y} \in \mathbb{F}_2^n / Supp(\mathbf{y}) \subseteq Supp(\mathbf{x})} f(\mathbf{y}).$$

To compute f^0, there exists a divide and conquer recursive algorithm. The algorithm is given as

Algorithm:

1. Write the truth-table of f, in which the binary vectors of length n are in lexicographic order.
2. Let f_0 be the restriction of f to$\mathbb{F}_{n-1}^2 \times \{0\}$ and f_1 the restriction of f to$\mathbb{F}_{n-1}^2 \times \{1\}$ the truth-table of f_0 (resp. f_1) corresponds to the upper (resp. lower) half of the table of f; replace the values of f_1 by those of $f_0 \oplus f_1$;
3. Apply recursively step 2, separately to the functions now obtained in the places of f_0 and f_1.

When the algorithm ends (i.e. when it arrives to functions on one variable each), the global table gives the values of the ANF of f. The complexity of this algorithm is $O(n2^n)$.

1.2 Calculation of Degree of a Boolean function from the Truth table:

The Truth table can be converted into the above mentioned form of ANF. The table gives the value of coefficients of the monomials written in lexicographic ordering. The degree of a monomial is equal to weight of the monomials when the monomial is represented as a member of $\mathbb{F}_2^n$. Thus, the degree of Boolean function is equal to the maximum weight of the monomial whose coefficient is one. In case, the ANF is sorted in descending order of their weights of monomials, the weight of the monomial corresponding to leftmost non-zero entry will be the degree of the Boolean function.

1.3 Row Echelon Form

A matrix is in row echelon form if it satisfies the following conditions.

1. All nonzero rows are above any rows of all zeroes.
2. The first nonzero number from the left in any non zero row (called leading coefficient or pivot) is 1.
3. The leading coefficient of a nonzero row is always strictly to the right of the leading coefficient of the row above it.

The row echelon form of a matrix is not unique and it depends on the algorithm that has been used to compute it. Below is an example of a matrix in row echelon form:

$$\begin{pmatrix} 1 & 4 & 3 & 5 & 6 \\ 0 & 1 & 4 & 3 & 1 \\ 0 & 0 & 2 & 3 & 3 \\ 0 & 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

A matrix can be transformed to its row echelon form by applying a finite number of elementary row operation. In other words, a matrix A of order $n \times m$ and its row echelon form E are connected as $E = PA$, P being a non-singular square matrix of the order m.

1.4 Row Echelon form and Rank

Let A be a matrix of order $n \times m$ and E be a row echelon form of A. Then the number of non-zero rows in E is unique and is equal to row rank of the matrix A. Another important conclusion we can draw from this fact is:

Theorem 1. *Let A be a matrix of order $n \times m$ and E be its row echelon form. The position of leading coefficient in the last non-zero is independent of E.*

Proof. Let $A = (a_{ij})_{n \times m}$. If possible suppose E and E_1 be two row echelon forms of matrix A such that the positions of leading coefficient in the last non-zero row are different (say l_1 and l_2). Without loss of generality we can assume that $l_1 < l_2$. Let the k^{th} row be the last non-zero row of E and E_1 where the counting starts from zero. Note that this is same for both E and E_1 as given in observation above. For E, E_1 there exist non-singular matrices P and P_1 of order m such that $A = PE = P_1E_1$. Take l such that $l_1 < l \leq l_2$. We consider the truncated matrix A' of order $n \times l$ such that every row is truncated after first l entries. Clearly, both P^{-1} and P_1^{-1} reduce A' to row echelon form say E' and E_1' as $E' = P^{-1}A'$ and $E_1' = P_1^{-1}A'$. Now E' will have more than $n - k-$ all zero rows while E_1' will have exactly $n - k - 1$ all zero rows. This implies that rank of A' as deduced from E' is strictly less than $k + 1$ while from E_1' it is $k + 1$ which is contradiction. Hence the assumption is true. □

1.5 Linear Combination of Rows

We establish connection between linear combinations of rows of a matrix and that of its row echelon form.

Theorem 2. *Let A be a square matrix of order n. Any linear combination of rows of A is also a linear combination (may be different) of its row echelon form E.*

Proof. Let $A = (R_1, R_2, \ldots, R_n)^T$ where $R_1, R_2, \ldots, R_n$ denotes the rows of matrix A. Consider a linear combination of rows of A i.e., $R = a_1R_1 + a_2R_2 + \ldots + a_nR_n$. Equivalently, we can write $R = QA$, Q being the row vector $(a_1, a_2, \ldots, a_n)$. As E is a row echelon form of A, there exists a non-singular square matrix P of order n such that $A = PE$. Therefore, $R = QA = QPE = SE$, where $S = QP$ and S is a row vector. Hence the theorem. □

2 Calculation of Cryptographic Degree of an S-Box

We have done optimization in the following ways:

2.1 Calculation of degrees of co-ordinate functions

Let $S : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be an S-Box. Let $f_1, f_2, \ldots, f_n$ be its co-ordinate functions. We have to compute

$$\deg(S) = \min\{\deg(\mathbf{b}.S), \mathbf{b} \in \mathbb{F}_2^m\}.$$

We observe that the algorithm (See section 1.1) that is used to find the ANF of a Boolean function can be directly applied to S-Box. This map an S-Box to another S-Box formed by concatenation of ANFs of co-ordinate functions. Once we convert S-Box to the desired form, we sort the entries as per weights of input. We use the ordering:

$$\begin{aligned} \mathbf{x} \leq \mathbf{y} \implies & \text{ either weight of } \mathbf{x} \leq \text{ weight of } \mathbf{y} \\ & \text{or weight of } \mathbf{x} = \text{ weight of } \mathbf{y} \text{ and } \mathbf{x} \leq \mathbf{y} \text{ lexicographically.} \end{aligned}$$

Once the sorting is done, the ANF co-ordinates are extracted.

2.2 Matrix formation

Each co-ordinate function is a binary array of 2^n bits. We form a matrix M of order $m \times 2^n$. In the row echelon form, observe the last row. If it is a zero row, the degree is 0 otherwise, weight of the monomial corresponding to the leading coefficient of the last row will be cryptographic degree of the S-Box (Theorem 1 and 2).

3 Example

Consider a 3×3 s-box S=0,2,1,7,4,7,7,0. We apply divide and conquer algorithm on S as

Input	S	I iteration	II iteration	III iteration
0	0	0	0	0
1	2	2	2	$0\oplus2=2$
2	1	1	$0\oplus1=1$	1
3	7	7	$7\oplus2=5$	$1\oplus5=4$
4	4	$0\oplus4=4$	4	4
5	7	$2\oplus7=5$	5	$4\oplus5=1$
6	7	$1\oplus7=6$	$6\oplus4=2$	2
7	0	$7\oplus0=7$	$5\oplus7=2$	$2\oplus2=0$

The output after IIIrd iteration is: 0,2,1,4,4,1,2,0. Sorting this array with respect to input weight gives:

Input	Output
0	0
1	2
2	1
4	4
3	4
5	1
6	2
7	0

Separating the components and filling them into the rows of 3×8 matrix from right to left we get matrix M as

$$M = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

The row echelon form of M is:

$$M = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

In the last row, the first 1 appears at 4th position from left. The corresponding input is 3 whose weight is 2. Hence the degree of S-box is 2.

4 Complexity of our algorithm

An straight forward approach to calculate degree of an $n \times m$ S-box involves 2^m invocation of divide-and-conquer algorithm for degree calculation. The complexity of this algorithm is $O(n2^n)$. Thus the overall complexity of degree calculation is $O(n2^{m+n})$.

Our approach mainly involves calculation of ANF of m co-ordinate function and reduction of a matrix of size $m \times 2^n$to its row echelon form. The complexity of algorithm is the complexity of this reduction i.e., $O(mn2^n + m^2 2^n) = O(m(m+n)2^n)$.

5 Experimental Results

We have calculated algebraic degree of a 26×64 S-box on an i7 PC with 64-bit Ubuntu 12.04 LTS. The calculations took around 78 seconds. In our case the computation from a nave approach would be of order $26 \times 2^{26+64} > 2^{94}$ while our algorithm computes this with complexity of order $26(26+64)2^{26} < 2^{38}$.