Overview on S-Box Design Principles

Debdeep Mukhopadhyay

Assistant Professor

Department of Computer Science and Engineering

Indian Institute of Technology Kharagpur

INDIA -721302

What is an S-Box?

- **S-Boxes are Boolean mappings from $\{0,1\}^m \rightarrow \{0,1\}^n$**
  - **m x n mappings**
- **Thus there are n component functions each being a map from m bits to 1 bit**
  - **in other words, each component function is a Boolean function in m Boolean variables**

Boolean Function

- **A Boolean function is a mapping from $\{0,1\}^m \rightarrow \{0,1\}$**
- **A Boolean function on n-inputs can be represented in minimal sum (XOR +) of products (AND .) form:**

$$f(x_1,\ldots,x_n) = a_0 + a_1.\, x_1 + \ldots + a_n.\, x_n + a_{1,2}.x_1.x_2 + \ldots + a_{n-1,n}.x_{n-1}.x_n + \ldots + a_{1,2,\ldots,n}\, x_1.x_2 \ldots x_n$$

- **The ANF form is canonical…**
- **If the <u>and</u> terms have all zero co-efficients we have an affine function**
- If the constant term is further 0, we have a **<u>linear</u>** function

Boolean Function

- **A Boolean function is a mapping from $\{0,1\}^m \rightarrow \{0,1\}$**

$f : \Sigma^n \rightarrow \{0,1\}$ be a Boolean Function.
Binary sequence $(f(\alpha_0), f(\alpha_1), \ldots, f(\alpha_{2^n-1}))$
is called the Truth Table of f

- **Sequence of a Boolean Function:**

$\{(-1)^{f(\alpha_0)}, (-1)^{f(\alpha_1)}, \ldots, (-1)^{f(\alpha_{2^n-1})}\}$ is called sequence of f

Balanced Function

- **A Boolean function is said to be <u>balanced</u> if its truth table has equal number of ones and zeros.**
- **The Hamming weight of a binary sequence is the number of ones**

Scalar Product of Sequences

- **Consider f and g as two Boolean functions.**
- **Consider, η be the sequence of f and ε be the sequence of g.**
- **Define,**

$<\eta, \varepsilon> =$ (#no of cases when f=g)-(#no of cases when f $\neq$ g)

Non-linearity

- **The non-linearity of a Boolean function can be defined as the distance between the function and the set of all affine functions.**

$$\therefore N_f = \min_{g \in A_n} d(f,g)$$

where A_n is the set of all affine functions over Σ^n

$$d(f,g) = 2^{n-1} - \frac{1}{2} < \eta, \varepsilon >$$

$$\therefore N_f = 2^{n-1} - \frac{1}{2} \max_{i=0,1,\ldots,2^{n-1}} \{|\eta, l_i|\},$$

where l_i is the sequence of a linear function in x

A Compact Representation of all the linear functions

- **Hadamard Matrix: Any rxr matrix with elements in {-1,1} if $HH^T=rI_r$, where I_r is the identity matrix of dimension rxr.**
- **Walsh Hadamard Matrix:**

$$H_0 = 1,\ H_1 = \begin{bmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{bmatrix}, n = 1, 2, \ldots$$

- **Each row of H_n is the sequence of a linear function in *x* belonging to $\{0,1\}^n$**
- **Each row, l_i is the sequence of the Boolean function,**

$g(x) = < \alpha_i, x >, \alpha_i$ is the binary representation of i
Note that α_i and x are not sequences, but they are binary tuples of length n

Effect of Input Transformation on balanced-ness and Non-linearity

- **If a Boolean function, f(x) is balanced, then so is g=f(xB ^ A), A is an n-bit vector and B is an nxn 0-1 invertible matrix**
- **Non-linearity of f and g are same.**

Strict Avalanche Criteria

- **Informally, if one bit input is changed in an S-Box, then half of the output bits should be changed**
- **For a function, f to satisfy SAC the following condition is satisfied:**

$f(x) \oplus f(x \oplus \alpha)$ is balanced, where wt(α)=1

- **Higher order SAC, when more than one input bits change**
- **Both the SAC and the higher order SAC together make Propagation Criteria (PC)**

How to make a Boolean Function satisfy SAC?

- **Consider a Boolean function, f(x)**
- **Consider a non-singular {0,1} matrix of dimension nxn.**
- **If for each row of the matrix A if:**

$$f(x) \oplus f(x \oplus \gamma) \text{ is balanced, } \gamma \text{ is a row of the matrix A}$$

then g(x)=f(xA) satisfies the SAC.

Example

- **f(x)=x1x2 ^ x3 does not satisfy SAC?**
- **Why? Consider α=(001)**
- **f(x)^f(x^e1) is balanced, e1=(100)**
- **f(x)^f(x^e2) is balanced, e2=(010)**
- **f(x)^f(x^e3) is balanced, e3=(111)**

$$A=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

- **Check that g(x)=f(xA) satisfies SAC**

Bent Functions

- **Non-linearity of Boolean functions have an upper bound**

$$N_f \leq 2^{n-1} - 2^{\frac{n}{2}-1}$$

- **Functions which achieve this are called Bent functions**
- **They satisfy PC for all α**
- **But they are always unbalanced**
- **Bent functions exist for even values of n**

Example

- **f(x)=x1x2 ^ x3x4 is a Bent function in 4 variables**
- **If f is a Bent function**
  - **so is f ^ (affine function)**
  - **f(xA ^ B) for a non-singular binary matrix A is also Bent**
- **Bent functions are not balanced. Number of zeros, is $2^{n-1} \pm 2^{n/2-1}$**

Creating Balanced Non-linear function

- Take 2^{n-k}, k-variable linear function, where k>n/2
- Concatenate the truth-tables
- Thus, we obtain a nxk mapping which is non-linear
  - $N_f \geq 2^{n-1}-2^{k-1}$
- Balanced
- Can be made to satisfy SAC.

Is the S-Box good against LC and DC?

- Not only the component functions are good:
  - high non-linearity
  - satisfy PC
  - etc.
- but their non-zero linear combinations also have to satisfy.
  - Challenging problem

Design of S-Box is even more complex

- **Good S-Boxes from the cryptographic point of view when put in hardware are found to leak information, like power consumption etc**
- **They thus lead to attacks called Side Channel Attacks, which can break ciphers in minutes…after all the hard-work**
- **Then there are Algebraic Attacks…**
- **So, what to do? Open Research Problem(s)…**

Criteria of Good S-Box

- **Balanced Component functions**
- **Non-linearity of Component functions high**
- **Non-zero linear combinations of Component functions balanced and highly non-linear**
- **Satisfies SAC**
- **High Algebraic degree**

Exercise

- **Enumerate 8 distinct linear functions in 5 variables, x_1, x_2, x_3, x_4, x_5**
- **Concatenate their Truth-tables to obtain an 8 input, 5 output function.**
- **Store the resultant mapping as a 8x5 S-Box.**
- **What is the non-linearity of your SBox?**
- **Does is satisfy SAC? If not, modify the function to do so.**

Further Reading

- **J. Seberry, Zhang, Zhang, "Cryptographic Boolean Functions via Group Hadamard Matrices", AJC Journal of Combinatorics, vol 10, 1994**
- **K. Nyberg, "Differentially Uniform Mappings for Cryptography", Eurocrypt 1993**
- **K. Nyberg, "Perfect Non-linear SBoxes", Eurocrypt 1991**

Next Days Topic

- **Modes of operation of Block Ciphers**