An algebraic equation of a round function in CAST-128

Joo Yeon Cho

Center for Advanced Computing – Algorithms and Cryptography,
Department of Computing, Macquarie University,
NSW, Australia, 2109
{jcho}@ics.mq.edu.au

1 Brief description of round functions

The Cast-128 is a DES-like Substitution-Permutation Network (SPN) cryptosystem which appears resistant against known cryptanalysis method.[1] [2] In the algorithm specifications, there are three different round functions as follows. Note that the input of round function is b and output is f, respectively.

$$\begin{aligned}
f_1 : I &= (k_m + b) <<< k_r) \\
f &= (((S_1[I_a] \oplus S_2[I_b]) - S_3[I_c]) + S_4[I_d]) \\
f_2 : I &= (k_m \oplus b) <<< k_r) \\
f &= (((S_1[I_a] - S_2[I_b]) + S_3[I_c]) \oplus S_4[I_d]) \\
f_3 : I &= (k_m - b) <<< k_r) \\
f &= (((S_1[I_a] + S_2[I_b]) \oplus S_3[I_c]) - S_4[I_d])
\end{aligned}$$

Here, b, k_m, I and the output f are 32-bit words: I_a, I_b, I_c, and I_d are the first through fourth bytes of I, respectively. $S_1 - S_4$ are $8x32$ S-boxes and $\oplus$ is XOR, "'+"' and "'-"' are addition and subtraction modulo 2^{32}. $<<<$ is circular left rotation by a 5-bit value k_r.

2 Deriving an algebraic equation in the round functions

Let's consider f_1 round function at first. For simplicity, we assume that $k_r = 0$. If we denote the least significant bit (LSB) of S_i-box output as α_i, then, the LSB of f, which is f_0, is described as follows.

$$f_0 = \alpha_1 \oplus \alpha_2 \oplus \alpha_3 \oplus \alpha_4$$

where "'-"' should be considered as a negation but anyhow we can obtain a linear relation among LSB's.

Next, we are going to obtain a non-linear multivariate equation for each α_i. Since S-box has 8-bit input, which is denoted as $(I_{a_1}, I_{a_2}, \cdots, I_{a_8})$, each α_i is able to be expressed as a combination of monomials which consist of only input variables up to degree 8.

$$\alpha_1 = h_1(I_{a_1}, I_{a_2}, \cdots, I_{a_8}) \tag{1}$$
$$\alpha_2 = h_2(I_{b_1}, I_{b_2}, \cdots, I_{b_8}) \tag{2}$$
$$\alpha_3 = h_3(I_{c_1}, I_{c_2}, \cdots, I_{c_8}) \tag{3}$$
$$\alpha_4 = h_4(I_{d_1}, I_{d_2}, \cdots, I_{d_8}) \tag{4}$$
$$\tag{5}$$

where $h_1, \cdots, h_4$ are non-linear functions of up to degree 8.
In final, we can express I_{a_i} as a function of input b and round key k_m. (Let's assume $k_r = 0$ for simple example). From the addition properties, we know

$$I_{a_8} = I_{32} = b_{32} \oplus k_{m_{32}} \oplus c_{32} \quad (6)$$
$$I_{a_7} = I_{31} = b_{31} \oplus k_{m_{31}} \oplus c_{31} \quad (7)$$
$$\cdots \quad (8)$$
$$I_{b_8} = I_{24} = b_{24} \oplus k_{m_{24}} \oplus c_{24} \quad (9)$$
$$\cdots \quad (10)$$
$$I_{b_1} = I_{17} = b_{17} \oplus k_{m_{17}} \oplus c_{17} \quad (11)$$
$$\cdots \quad (12)$$
$$I_{d_8} = I_8 = b_8 \oplus k_{m_8} \oplus c_8 \quad (13)$$
$$\cdots \quad (14)$$
$$I_{d_1} = I_1 = b_1 \oplus k_{m_1} \quad (15)$$

where c_i can be expressed as a function of partial bits of b and k which are indexed lower than i.
Now, we are going to describe the modular addition in a different way as follows.

Lemma 1. *Let z_i be the i-th output bit of the modular addition $c = a + b$. Then, $z_0 = x_0 \oplus y_0$, $z_1 = x_1 \oplus y_1 \oplus x_0 y_0$ and for $2 \leq i \leq 31$,*

$$z_i = x_i \oplus y_i \oplus x_{i-1}y_{i-1} \oplus \sum_{t=0}^{i-2} x_t y_t \{ \prod_{r=t+1}^{i-1} (x_r \oplus y_r) \}$$

By Lemma (1), each z_i is expressed as a function of input bits of degree $i + 1$.

Theorem 1. *Let c_i, $24 \leq i \leq 31$ be the i-th output bit of modular addition $c = a \boxplus b$. If c_i is multiplied by $(1 \oplus a_{23} \oplus b_{23})$, then, the degree of $c_i \cdot (1 \oplus a_{23} \oplus b_{23})$ is reduced to $(i - 22)$.*

If we combine α_i equation with above properties, then,

$$\alpha_1 = h_1(I_{a_1}, I_{a_2}, \cdots, I_{a_8}) = h_1(I_{32}, \ldots, I_{25}) = h_1(b_{32}, \ldots, b_1, k_{m_{32}}, \ldots, k_{m_1})$$

with high degree.
By Theorem (1), I_i can be reduced to a multivariate equation of degree up to 9 by multiplying a carefully chosen polynomial. That is,

$$\alpha_1 \cdot (1 \oplus b_{24} \oplus k_{m_{24}}) = h_1(I_{32}, \ldots, I_{25}) \cdot (1 \oplus b_{24} \oplus k_{m_{24}}) = H_1(b_{32}, \ldots, b_{24}, k_{m_{32}}, \ldots, k_{m_{24}})$$

The exactly same approach is applicable to α_2, α_3 and α_4.

$$\alpha_2 \cdot (1 \oplus b_{16} \oplus k_{m_{16}}) = h_2(I_{24}, \ldots, I_{17}) \cdot (1 \oplus b_{16} \oplus k_{m_{16}}) = H_2(b_{24}, \ldots, b_{16}, k_{m_{24}}, \ldots, k_{m_{16}})$$
$$\alpha_3 \cdot (1 \oplus b_8 \oplus k_{m_8}) = h_3(I_{16}, \ldots, I_9) \cdot (1 \oplus b_8 \oplus k_{m_8}) = H_3(b_{16}, \ldots, b_8, k_{m_{16}}, \ldots, k_{m_8})$$
$$\alpha_4 = h_4(I_8, \ldots, I_1) = H_4(b_8, \ldots, b_1, k_{m_8}, \ldots, k_{m_1})$$

Conclusively, the LSB of output of a round function is expressed as a small degree of equation as follows.

$$
\begin{aligned}
&f_0 \cdot (1 \oplus b_{24} \oplus k_{m_{24}}) \cdot (1 \oplus b_{16} \oplus k_{m_{16}}) \cdot (1 \oplus b_8 \oplus k_{m_8}) \\
&= H_1(b_{32}, \ldots, b_{24}, k_{m_{32}}, \ldots, k_{m_{24}}) \cdot (1 \oplus b_{16} \oplus k_{m_{16}}) \cdot (1 \oplus b_8 \oplus k_{m_8}) \\
&\oplus H_2(b_{24}, \ldots, b_{16}, k_{m_{24}}, \ldots, k_{m_{16}}) \cdot (1 \oplus b_{24} \oplus k_{m_{24}}) \cdot (1 \oplus b_8 \oplus k_{m_8}) \\
&\oplus H_3(b_{16}, \ldots, b_8, k_{m_{16}}, \ldots, k_{m_8}) \cdot (1 \oplus b_{24} \oplus k_{m_{24}}) \cdot (1 \oplus b_{16} \oplus k_{m_{16}}) \\
&\oplus H_4(b_8, \ldots, b_1, k_{m_8}, \ldots, k_{m_1}) \cdot (1 \oplus b_{24} \oplus k_{m_{24}}) \cdot (1 \oplus b_{16} \oplus k_{m_{16}}) \cdot (1 \oplus b_8 \oplus k_{m_8})
\end{aligned}
$$

References

1. C. Adams, *The cast-128 encryption algorithm*, Internet Request for Comments RFC 2144, May 1997.
2. ———, *The cast-256 encryption algorithm*, Internet Request for Comments RFC 2612, June 1999.