Design of SAC/PC(l) of Order k Boolean Functions and Three Other Cryptographic Criteria

Kaoru Kurosawa[1] and Takashi Satoh *[2]

[1] Dept. of Computer Science,
Graduate School of Information Science and Engineering,
Tokyo Institute of Technology
[2] Dept. of Physical Electronics, Faculty of Engineering, Tokyo Institute of Technology
2–12–1 O-okayama, Meguro-ku, Tokyo 152, Japan
kurosawa@ss.titech.ac.jp, tsato@ss.titech.ac.jp

Abstract. A Boolean function f satisfies PC(l) of order k if $f(x) \oplus f(x \oplus \alpha)$ is balanced for any α such that $1 \leq W(\alpha) \leq l$ even if any k input bits are kept constant, where $W(\alpha)$ denotes the Hamming weight of α. This paper shows the first design method of such functions which provides $\deg(f) \geq 3$. More than that, we show how to design "balanced" such functions. High nonlinearity and large degree are also obtained. Further, we present balanced SAC(k) functions which achieve the maximum degree. Finally, we extend our technique to vector output Boolean functions.

1 Introduction

The security of block ciphers is often studied by viewing their S-boxes (or F functions) as a set of Boolean functions. SAC [15] and PC(l) [11] are important cryptographic criteria of such Boolean functions. Let $W(\alpha)$ denote the Hamming weight of $\alpha \in \{0,1\}^n$. For a Boolean function $f(x) = f(x_1, \ldots, x_n)$, define

$$\frac{Df}{D\alpha} \triangleq f(x) \oplus f(x \oplus \alpha) \ .$$

$f(x)$ is said to satisfy

- SAC if $Df/D\alpha$ is balanced for any α such that $W(\alpha) = 1$.
- SAC(k) if any function obtained from f by keeping any k input bits constant satisfies SAC.
- PC(l) if $Df/D\alpha$ is balanced for any α such that $1 \leq W(\alpha) \leq l$.
- PC(l) of order k if any function obtained from f by keeping any k input bits constant satisfies PC(l).

* This author was supported by the Telecommunications Advancement Foundation, Japan.

W. Fumy (Ed.): Advances in Cryptology - EUROCRYPT '97, LNCS 1233, pp. 434-449, 1997.

Well known bent functions satisfy both SAC and PC(l) for all $l \leq n$, but not necessarily SAC(k) nor PC(l) of order k for $k \geq 1$.

On the other hand, balancedness, algebraic degree and nonlinearity are another important cryptographic criteria.

- Let deg(f) denote the degree of the highest degree term in the algebraic normal form of f. Then deg(f) must be large. Actually, Jacobsen and Knudsen showed an attack against block ciphers with small deg(f) recently [2].
- The nonlinearity of a Boolean function f, denoted by $N(f)$, is defined as the minimum distance of f from the set of affine functions.

$$N(f) \triangleq \min_{a_0,\ldots,a_n} \left|\{x \mid f(x) \neq a_0 \oplus a_1 x_1 \oplus \cdots \oplus a_n x_n\}\right| .$$

  $N(f)$ must be large to avoid the linear attack [7].
- Preneel et al. showed a balanced SAC($n-2$) function for n =odd [11]. Lloyd [5] showed a condition such that SAC($n-3$) functions are balanced. Balanced SAC functions with high nonlinearity were constructed by [14]. Recently, other balanced SAC functions were given by [16].

However,

(1) No general methods are known which design Boolean functions satisfying PC(l) of order k except deg(f) $= 2$. (For deg(f) $= 2$, see [11, 12].)
(2) Balanced SAC(k) functions are not known for $1 \leq k \leq n-4$.
(3) Balanced functions satisfying PC(l) of order k are not known for any $l \geq 2$ and any k.

This paper shows a design method of PC(l) of order k functions. The proposed method is the first design method which provides deg(f) ≥ 3. We construct f as

$$f(x_1,\ldots,x_s,y_1,\ldots,y_t) \triangleq [x_1,\ldots,x_s]Q[y_1,\ldots,y_t]^T \oplus g(x_1,\ldots,x_s) , \qquad (1)$$

where Q is an $s \times t$ binary matrix and $g(x_1,\ldots,x_s)$ is any function. Then f satisfies PC(l) of order k if Q satisfies the following conditions.

- $W(Q\gamma_1) \geq k+1$ for any $t \times 1$ vector γ_1 such that $1 \leq W(\gamma_1) \leq l$.
- $W(\gamma_2 Q) \geq k+1$ for any $1 \times s$ vector γ_2 such that $1 \leq W(\gamma_2) \leq l$.

Such a matrix Q is obtained by the product of two generator matrices of error correcting codes. Further, it is shown that balanced f can be obtained by choosing g appropriately in (1). We can also obtain large degree and high nonlinearity such that

- deg(f) $= s/2$ and $N(f) \geq 2^{t+s-1} - 2^{t+s/2-1}$ for s =even.
- deg(f) $= (s-1)/2$ and $N(f) \geq 2^{t+s-1} - 2^{t+(s-1)/2}$ for s =odd.

The above $N(f)$ is almost the maximum if t is small. (The deg(f) and $N(f)$ for SAC(k) are obtained by substituting $t = k + 1$ and $s = n - k - 1$.)

Next, SAC(k) functions with the maximum deg(f) are obtained for $k \leq n/2 - 1$. This shows that an upper bound on deg(f) of SAC(k) functions given by Preneel et al. [11] is tight. Further, balanced SAC(k) functions with the same maximum degree are presented for $n - k - 1 =$ odd. This means that the bound of [11] is tight even for balanced SAC(k) functions if $k \leq n/2 - 1$ and $n - k - 1 =$ odd. It will be a further work to find a tight upper bound on deg(f) of balanced SAC(k) functions for $n - k - 1 =$ even.

Finally, we extend our technique to vector output Boolean functions. Vector output PC(2) of order $2^{r-1} - 1$ functions and vector output SAC(k) functions are obtained which also possess high nonlinearity and large degree.

2 Preliminaries

$f(x_1, \ldots, x_n)$ denotes a mapping from $\{0,1\}^n$ to $\{0,1\}$. For a binary string α, $W(\alpha)$ denotes the Hamming weight of α. We use square brackets to denote vectors like $[a_1, \ldots, a_n]$ and round brackets to denote functions like $f(x_1, \ldots, x_n)$.

2.1 Balance and Algebraic Degree

We say that $f(x)$ is balanced if

$$\big|\{x \mid f(x) = 0\}\big| = \big|\{x \mid f(x) = 1\}\big| = 2^{n-1} \ ,$$

where $x = [x_1, \ldots, x_n]$.

Definition 1. We call $f(x) = c \oplus a_1 x_1 \oplus \cdots \oplus a_n x_n$ an affine function.

Proposition 2. *A non-constant affine function is balanced.*

Proposition 3. *[14] $f(x_1, \ldots, x_s) \oplus g(y_1, \ldots, y_t)$ is balanced if f is balanced or g is balanced.*

The following form is called the *algebraic normal form* of f.

$$f(x_1, \ldots, x_n) = a_0 \oplus \bigoplus_{i=1}^{n} a_i x_i \oplus \bigoplus_{1 \leq i < j \leq n} a_{ij} x_i x_j \oplus \cdots \oplus a_{12\ldots n} x_1 x_2 \ldots x_n \ .$$

deg(f) denotes the degree of the highest degree term in the algebraic normal form of f.

2.2 Bent Function and Nonlinearity

Bent functions are defined as follows.

Definition 4. [13] $f(x_1, \ldots, x_n)$ is a bent function if

$$\left|\sum_x (-1)^{f(x)}(-1)^{\omega_1 x_1 + \cdots + \omega_n x_n}\right| = 2^{n/2} \tag{2}$$

for any $[\omega_1, \ldots, \omega_n] \in \{0,1\}^n$.

Define a distance between two Boolean functions $f(x)$ and $g(x)$ as

$$d(f,g) \triangleq \left|\{x \mid f(x) \neq g(x)\}\right| .$$

Definition 5. [10] The nonlinearity of a Boolean function f, denoted by $N(f)$, is defined as

$$N(f) \triangleq \min_{a_0,\ldots,a_n} d(f(x), a_0 \oplus a_1 x_1 \oplus \cdots \oplus a_n x_n) .$$

$N(f)$ is the distance of f from the set of affine functions and it should be large to avoid the linear attack. It is known that each bent function has the maximum $N(f)$.

Proposition 6. *[8, 13] $N(f) \leq 2^{n-1} - 2^{n/2-1}$.*

Proposition 7. *[8, 13] The equality of Proposition 6 is satisfied if and only if f is a bent function.*

2.3 SAC and SAC(k)

f satisfies SAC if complementing any single input bit changes the output bit with probability a half.

Definition 8. [1, 15]

(1) $f(x_1, \ldots, x_n)$ satisfies SAC (*the strict avalanche criterion*) if $f(x) \oplus f(x \oplus \alpha)$ is balanced for any $\alpha \in \{0,1\}^n$ such that $W(\alpha) = 1$.
(2) $f(x)$ satisfies SAC(k) if any function obtained from $f(x)$ by keeping any k input bits constant satisfies SAC. We say that f is an SAC(k) function if $f(x)$ satisfies SAC(k).

Proposition 9. *[1] There exist no SAC($n-1$) functions.*

Proposition 10. *[11]*

(1) If $f(x_1, \ldots, x_n)$ satisfies SAC($n-2$), then $\deg(f)=2$.
(2) If $f(x_1, \ldots, x_n)$ satisfies SAC(k) for $0 \leq k \leq n-3$, then

$$\deg(f) \leq n - k - 1 . \tag{3}$$

Preneel et al. showed a design method of SAC(k) functions for $\deg(f) = 2$.

Proposition 11. *[11] Suppose that $\deg(f) = 2$ and $n > 2$. Then, f satisfies SAC(k) if and only if every variable x_i occurs in at least $k+1$ second order terms of the algebraic normal form, where $0 \leq k \leq n-2$.*

2.4 PC(l) and PC(l) of Order k

f satisfies PC(l) if complementing any l or less input bits changes the output bit with probability a half.

Definition 12. [11]

(1) $f(x_1, \ldots, x_n)$ satisfies PC(l) if $f(x) \oplus f(x \oplus \alpha)$ is balanced for any $\alpha \in \{0,1\}^n$ such that $1 \leq W(\alpha) \leq l$.
(2) $f(x)$ satisfies PC(l) of order k if any function obtained from $f(x)$ by keeping any k input bits constant satisfies PC(l). We say that f is a PC(l) of order k function if $f(x)$ satisfies PC(l) of order k.

It is well known that f satisfies PC(n) if and only if f is a bent function [11]. Bent functions, however, do not necessarily satisfy PC(l) of order k.

PC(n) functions, therefore bent functions, exist only for n =even from (2). Preneel et al. [12] showed the following functions which have $\deg(f) = 2$.

Proposition 13. *There exists a PC($n-1$) of order 1 function for n =odd.*

Proposition 14. *[11] Let*

$$s_n(x_1, \ldots, x_n) \triangleq \bigoplus_{1 \leq i < j \leq n} x_i x_j \ .$$

Then s_n satisfies PC(l) of order k if $l + k \leq n - 1$ or if $l + k = n$ and l is even. Further,

(1) s_n is the only function which satisfies PC(1) of order $n-2$ (or SAC($n-2$)).
(2) s_n is the only function which satisfies PC(2) of order $n-2$.
(3) s_n is balanced if n =odd.

Proposition 15.

(1) There exists a balanced SAC($n-2$) function if n =odd.
(2) There exist no balanced SAC($n-2$) functions if n =even

Proof.

(1) From (1) and (3) of Proposition 14.
(2) From line 4 of p.171 of [11] and (1) of Proposition 14, a SAC($n-2$) function is a bent function if n =even. Further, bent functions cannot be balanced [13].

□

3 How to Design PC(l) of Order k Functions

This section shows the first design method of PC(l) of order k functions which provides $\deg(f) \geq 3$. (For $\deg(f) = 2$, see Sect. 2.4.) The proposed method is also a design method of SAC(k) functions since SAC(k) is equivalent to PC(1) of order k.

3.1 Basic Theorem

Theorem 16. *For positive integers l and k, suppose that there exists an $s \times t$ binary matrix Q such as follows.*

(1) $s \geq \max\{l, k+1\}$ *and* $t \geq \max\{l, k+1\}$.
(2) $W(Q\gamma_1) \geq k+1$ *for any* $t \times 1$ *vector* γ_1 *such that* $1 \leq W(\gamma_1) \leq l$.
(3) $W(\gamma_2 Q) \geq k+1$ *for any* $1 \times s$ *vector* γ_2 *such that* $1 \leq W(\gamma_2) \leq l$.

Now define

$$f(x_1, \ldots, x_s, y_1, \ldots, y_t) \triangleq [x_1, \ldots, x_s]Q[y_1, \ldots, y_t]^T \oplus g(x_1, \ldots, x_s) \ , \qquad (4)$$

where $g(x_1, \ldots, x_s)$ *is any function and* $n = s + t$*. Then* f *satisfies PC(*l*) of order* k*.*

Proof. Keep any k input bits constant. Without loss of generality, we can assume that

$$x_1 = b_1, \ldots, x_u = b_u, \quad y_1 = c_1, \ldots, y_v = c_v,$$

where $u + v = k$, $u < s$ and $v < t$. Substitute these bits into f and let

$$\hat{f}(x_{u+1}, \ldots, x_s, y_{v+1}, \ldots, y_t) \triangleq f(b_1, \ldots, b_u, x_{u+1}, \ldots, x_s, c_1, \ldots, c_v, y_{v+1}, \ldots, y_t) \ .$$

We have to prove that $\hat{f}(x) \oplus \hat{f}(x \oplus \alpha)$ is balanced for any α such that $1 \leq W(\alpha) \leq l$. For simplicity, we show a proof for $l = 2$. The proof for $l \geq 3$ is similar.

For $W(\alpha) = 2$, define

$$\frac{D\hat{f}}{Dx_{u+i}x_{u+j}} \triangleq \hat{f}(x_{u+1}, \ldots, x_s, y_{v+1}, \ldots, y_t) \oplus \hat{f}(\ldots, x_{u+i} \oplus 1, \ldots, x_{u+j} \oplus 1, \ldots)$$
$$\frac{D\hat{f}}{Dy_{v+i}y_{v+j}} \triangleq \hat{f}(x_{u+1}, \ldots, x_s, y_{v+1}, \ldots, y_t) \oplus \hat{f}(\ldots, y_{v+i} \oplus 1, \ldots, y_{v+j} \oplus 1, \ldots)$$
$$\frac{D\hat{f}}{Dx_{u+i}y_{v+j}} \triangleq \hat{f}(x_{u+1}, \ldots, x_s, y_{v+1}, \ldots, y_t) \oplus \hat{f}(\ldots, x_{u+i} \oplus 1, \ldots, y_{v+j} \oplus 1, \ldots) \ .$$

Let q_i be the i-th column vector of Q and p_i be the i-th row vector of Q. First, we obtain

$$\frac{D\hat{f}}{Dy_{v+i}y_{v+j}} = [b_1, \ldots, b_u, x_{u+1}, \ldots, x_s](q_{v+i} \oplus q_{v+j}) \ . \qquad (5)$$

From condition (2) of this theorem, $W(q_{v+i} \oplus q_{v+j}) \geq k+1$. On the other hand, $u \leq k$. Therefore, the right hand side of (5) is a non-constant affine function. Hence, $D\hat{f}/Dy_{v+i}y_{v+j}$ is balanced from Proposition 2.

Next, for g, define

$$\hat{g}(x_{u+1}, \ldots, x_s) \triangleq g(b_1, \ldots, b_u, x_{u+1}, \ldots, x_s) \ .$$

Further, define $\dfrac{D\hat{g}}{Dx_{u+i}}$ and $\dfrac{D\hat{g}}{Dx_{u+i}x_{u+j}}$ similarly to $\hat{f}$. Then we obtain

$$\frac{D\hat{f}}{Dx_{u+i}x_{u+j}} = (p_{u+i} \oplus p_{u+j})[c_1, \ldots, c_v, y_{v+i}, \ldots, y_t]^T \oplus \frac{D\hat{g}}{Dx_{u+i}x_{u+j}} \ .$$

From condition (3) of this theorem, $W(p_{u+i} \oplus p_{u+j}) \geq k+1$. On the other hand, $v \leq k$. Therefore, $(p_{u+i} \oplus p_{u+j})[c_1, \ldots, c_v, y_{v+i}, \ldots, y_t]^T$ is a non-constant affine function. Hence, $D\hat{f}/Dx_{u+i}x_{u+j}$ is balanced from Proposition 3.

Finally, we have

$$\frac{D\hat{f}}{Dx_{u+i}y_{v+j}} = p_{u+i}[c_1, \ldots, c_v, y_{v+i}, \ldots, y_t]^T$$
$$\oplus [b_1, \ldots, b_u, x_{u+1}, \ldots, x_s] q_{v+j} \oplus \frac{D\hat{g}}{Dx_{u+i}} \ .$$

Here, $p_{u+i}[c_1, \ldots, c_v, y_{v+i}, \ldots, y_t]^T$ is a non-constant affine function since $v \leq k$ and $W(p_{u+i}) \geq k+1$. Hence, $D\hat{f}/Dx_{u+i}y_{v+j}$ is balanced from Proposition 3.

Thus, we have proved that $\hat{f}(x) \oplus \hat{f}(x \oplus \alpha)$ is balanced for any α such that $W(\alpha) = 2$. Similarly, we can show that it is balanced for $W(\alpha) = 1$. Consequently, f satisfies PC(2) of order k. □

3.2 How to Find Q

This subsection shows that the matrix Q of Theorem 16 can be obtained by using generator matrices of error correcting codes.

Definition 17. A linear $[N, h, d]$ code is a binary linear code of length N, dimension h and the minimum Hamming distance at least d.

Definition 18. The dual code $C^{\perp}$ of a linear code C is defined as

$$C^{\perp} \triangleq \{u \mid u \cdot v = 0 \text{ for all } v \in C\} \ .$$

The dual minimum Hamming distance of C is defined as the minimum Hamming distance of $C^{\perp}$.

Theorem 19. *Let G_1 be a generator matrix of a linear $[t, h, d_1]$ code C_1 with the dual minimum Hamming distance d_1'. Let G_2 be a generator matrix of a linear $[s, h, d_2]$ code C_2 with the dual minimum Hamming distance d_2'. Let*

$$Q \triangleq G_2^T G_1 \ .$$

Then Q satisfies the conditions of Theorem 16 for

$$l = \min(d_1', d_2') - 1$$
$$k = \min(d_1, d_2) - 1 \ .$$

Proof. We first show that Q satisfies condition (2) of Theorem 16. Let γ_1 be a $t \times 1$ vector such that $1 \leq W(\gamma_1) \leq l$. γ_1 is not a codeword of $C_1^\perp$ because $W(\gamma_1) \leq l < d_1'$. Then,

$$G_1\gamma_1 \neq 0$$

because G_1 is a parity check matrix of $C_1^\perp$. Therefore,

$$Q\gamma_1 = G_2^T(G_1\gamma_1)$$

is a nonzero codeword of C_2 because G_2 is a generator matrix of C_2. Hence,

$$W(Q\gamma_1) \geq d_2 \geq k+1 \ .$$

Similarly, Q satisfies condition (3) of Theorem 16. □

By using Theorem 19, we can obtain the following results, for example.

Proposition 20. *[6, p.30] Let C be a $[2^r-1, 2^r-1-r, 3]$ Hamming code. Then $C^\perp$ is a $[2^r-1, r, 2^{r-1}]$ simplex code.*

Corollary 21. *For $r \geq 2$, there exists*

(1) a PC($2^{r-1}-1$) of order 2 function such that $n = 2^{r+1}-2$ and
(2) a PC(2) of order $2^{r-1}-1$ function such that $n = 2^{r+1}-2$.

Proposition 22. *[6, p.31] Let C be a $[2^r, 2^r-1-r, 4]$ extended Hamming code. Then $C^\perp$ is a $[2^r, r+1, 2^{r-1}]$ first order Reed–Muller code.*

Corollary 23. *For $r \geq 2$, there exists*

(1) a PC($2^{r-1}-1$) of order 3 function such that $n = 2^{r+1}$ and
(2) a PC(3) of order $2^{r-1}-1$ function such that $n = 2^{r+1}$.

4 Balance, Large Degree and High Nonlinearity

We can obtain "balanced" PC(l) of order k functions by choosing g appropriately in Theorem 16. Large degree and high nonlinearity can also be obtained.

4.1 Balanced PC(l) of Order k

Definition 24. We say that g is balanced for a matrix Q if

$$\left|\{x \mid g(x)=0, xQ=0\}\right| = \left|\{x \mid g(x)=1, xQ=0\}\right| \ . \tag{6}$$

Theorem 25. *In (4), f is balanced if g is balanced for Q.*

Proof. Substitute $x_1 = b_1, \ldots, x_s = b_s$ into (4), where $b_1, \ldots, b_s$ are constant bits. Then we have

$$f(b_1, \ldots, b_s, y_1, \ldots, y_t) = [b_1, \ldots, b_s]Q[y_1, \ldots, y_t]^T \oplus g(b_1, \ldots, b_s) \ . \tag{7}$$

If $[b_1, \ldots, b_s]Q \neq 0$, the right hand side of (7) is a non-constant affine function. Therefore, $f(b_1, \ldots, b_s, y_1, \ldots, y_t)$ is balanced from Proposition 2. For $[b_1, \ldots, b_s]$ such that $[b_1, \ldots, b_s]Q = 0$, we have

$$f(b_1, \ldots, b_s, y_1, \ldots, y_t) = g(b_1, \ldots, b_s) \ .$$

Then because g is balanced for Q, we see that $f(x_1, \ldots, x_s, \hat{y}_1, \ldots, \hat{y}_t)$ is balanced for Q for any fixed $(\hat{y}_1, \ldots, \hat{y}_t)$.

Consequently, $f(x_1, \ldots, x_s, y_1, \ldots, y_t)$ is balanced. □

We can find such g in the following way.

Lemma 26. *Suppose that $g(x_1, \ldots, x_n)$ is written as*

$$g(x_1, \ldots, x_s) = a_1x_1 \oplus \cdots \oplus a_sx_s \tag{8}$$

if $[x_1, \ldots, x_n]Q = 0$. Then g is balanced for Q if and only if $[a_1, \ldots, a_s]^T$ is linearly independent of the columns of Q.

Proof. First, it is easy to see that g of (8) is balanced for Q if and only if there is an x such that

$$xQ = 0 \text{ but } g(x) = 1 \ . \tag{9}$$

This condition is equivalent to say that the kernel (zero space) of Q^T is not contained in the zero space of the linear mapping

$$g(x) = [a_1, \ldots, a_s]x^T \ .$$

This holds if and only if $[a_1, \ldots, a_s]$ is linearly independent of the rows of Q^T. □

Corollary 27. *Let $xQ = [h_1(x), \ldots, h_t(x)]$. Define*

$$g(x_1, \ldots, x_s) \triangleq a_1x_1 \oplus \cdots \oplus a_sx_s \oplus h_1(x)h_2(x)\ldots h_t(x)H(x) \ ,$$

where $H(x)$ is any function. Then g is balanced for Q if and only if $[a_1, \ldots, a_s]^T$ is linearly independent of the columns of Q.

Another way of finding a balanced g for Q is to write its truth table.

4.2 Large Degree and High Nonlinearity

In (4), we can obtain $\deg(f) = s$ by letting

$$g(x_1, \ldots, x_s) = x_1 \ldots x_s \ .$$

Further, PC(l) of order k functions which possess high nonlinearity and large degree at the same time can be obtained as follows.

Theorem 28. *There exists a PC(l) of order k function f such that*

- $\deg(f) = s/2$ *and* $N(f) \geq 2^{t+s-1} - 2^{t+s/2-1}$ *for* s =*even.*
- $\deg(f) = (s-1)/2$ *and* $N(f) \geq 2^{t+s-1} - 2^{t+(s-1)/2}$ *for* s =*odd,*

where s and t are defined in Theorem 16.

Proof. For s =even, there exists a bent function $g(x_1, \ldots, x_s)$ such that $\deg(g) = s/2$. By choosing this g in (4), we obtain $\deg(f) = s/2$. Next, we compute the distance between this f and an affine function $A(x_1, \ldots, x_s, y_1, \ldots, y_t)$. Substitute $y_1 = c_1, \ldots y_t = c_t$ into f and A, where $c_1, \ldots, c_t$ are constant bits. Let

$$f_0(x_1, \ldots, x_s) \triangleq f(x_1, \ldots, x_s, c_1, \ldots c_t) = g(x_1, \ldots, x_s) \oplus B(x_1, \ldots, x_s)$$
$$A_0(x_1, \ldots, x_s) \triangleq A(x_1, \ldots, x_s, c_1, \ldots c_t) \ ,$$

where

$$B(x_1, \ldots, x_s) \triangleq [x_1, \ldots, x_s] Q [c_1, \ldots c_t]^T \ .$$

Then

$$d(f, A) = \sum_{c_1, \ldots c_t} d(f_0, A_0) = \sum_{c_1, \ldots c_t} d(g \oplus B, A_0)$$
$$= \sum_{c_1, \ldots c_t} d(g, A_0 \oplus B) \geq \sum_{c_1, \ldots c_t} N(g) = 2^t(2^{s-1} - 2^{s/2-1})$$

from Proposition 7. The above inequality holds for any affine function A. Therefore, $N(f) \geq 2^t(2^{s-1} - 2^{s/2-1})$.

For s = odd, let $\hat{g}(x_1, \ldots, x_{s-1})$ be a bent function with degree $(s-1)/2$ and let $g(x_1, \ldots, x_s) = \hat{g}(x_1, \ldots, x_{s-1})$. (Bent functions exist only for s = even.) □

Compare Theorem 28 with Proposition 6. Then we see that the above $N(f)$ is almost the maximum if t is small. (From condition (1) of Theorem 16, $t \geq \max\{l, k+1\}$, though.)

5 Balanced SAC(k) with the Maximum Degree

Proposition 10 gives an upper bound on the degree of SAC(k) functions. In Sect. 5.2, we will show that this bound is tight for $k \leq n/2 - 1$. Further, Sect. 5.3 will show that this bound is tight even for balanced SAC(k) functions for $k \leq n/2 - 1$ and $n - k - 1$=odd.

5.1 How to Design SAC(k) Functions

First, we can obtain SAC(k) functions as a special case of Theorem 16.

Corollary 29. *Let*

$$f(x_1,\dots,x_n) = (x_1 \oplus \cdots \oplus x_{n-k-1})(x_{n-k} \oplus \cdots \oplus x_n) \oplus g(x_1,\dots,x_{n-k-1}) \ , \quad (10)$$

where $g(x_1,\dots,x_{n-k-1})$ *is any function. Then* f *satisfies SAC*(k) *if* $k \leq \frac{n}{2} - 1$.

Proof. In Theorem 16, let

$$Q = \text{ the } (n-k-1) \times (k+1) \text{ matrix whose elements are all one.} \quad (11)$$

If $n-k-1 \geq k+1$, Q satisfies conditions (2) and (3) of Theorem 16 for $l = 1$. □

5.2 SAC(k) with the Maximum Degree

Theorem 30. *There exists an SAC*(k) *function* $f(x_1,\dots,x_n)$ *which meets the equality of (3) for* $k \leq \frac{n}{2} - 1$.

Proof. In Corollary 29, let $g(x_1,\dots,x_{n-k-1}) = x_1 \dots x_{n-k-1}$. Then we obtain $\deg(f) = n-k-1$ and the equality of (3) is satisfied. □

Remark. Proposition 11 shows that Proposition 10 is tight for $k = n-2$ and $n-3$.

5.3 Balanced SAC(k) with the Maximum Degree

Theorem 31. *There exists a balanced SAC*(k) *function* $f(x_1,\dots,x_n)$ *which meets the equality of (3) if* $k \leq \frac{n}{2} - 1$ *and* $k - n - 1 =$ *odd.*

Proof. In (10), let

$$g(x_1,\dots,x_{n-k-1}) = a_1x_1 \oplus \cdots \oplus a_{n-k-1}x_{n-k-1} \oplus x_1 \dots x_{n-k-1} \ ,$$

where

$$[a_1,\dots,a_{n-k-1}] \neq [0,\dots,0], \ [1,\dots,1] \ . \quad (12)$$

We show that this g is balanced for Q, where Q is given by (11). Let $x = [x_1,\dots,x_{n-k-1}]$. Note that $x_1 \dots x_{n-k-1} = 0$ if $W(x) < n-k-1 =$(odd). Also, $W(x) =$ even if $xQ = 0$. Therefore, $x_1 \dots x_{n-k-1} = 0$ if $W(x) =$even and hence if $xQ = 0$. Hence,

$$g(x_1,\dots,x_{n-k-1}) = a_1x_1 \oplus \cdots \oplus a_{n-k-1}x_{n-k-1}$$

if $xQ = 0$. Further, $[a_1,\dots,a_s]$ satisfying (12) is linearly independent of the columns of Q. Then g is balanced for Q from Lemma 26.

Consequently, f of (10) is balanced from Theorem 25. □

Theorem 32. *For $k - n - 1 =$ even, there exists a balanced SAC(k) function such that* $\deg(f) = n - k - 2$.

Proof. Let

$$g(x_1, \dots, x_{n-k-1}) = a_1 x_1 \oplus \cdots \oplus a_{n-k-1} x_{n-k-1} \oplus x_1 \dots x_{n-k-2} \oplus x_2 \dots x_{n-k-1} ,$$

where

$$[a_1, \dots, a_{n-k-1}] \neq [0, \dots, 0], \ [1, \dots, 1]$$

We can show that g is balanced for Q, where Q is given by (11). □

It will be a further work to find a tight upper bound on $\deg(f)$ of balanced SAC(k) functions for $n - k - 1 =$ even.

Remark.

(1) For balanced SAC($n-2$) functions, see Proposition 15.
(2) Lloyd [5] showed a condition such that SAC($n-3$) functions are balanced.
(3) Balanced SAC functions with high nonlinearity were constructed by [14]. Recently, other balanced SAC functions were given by [16].

6 Extension to Vector Output Boolean Functions

In this section, we extend our technique to vector output Boolean functions.

6.1 General Results

Let F denote a mapping from $\{0,1\}^n$ to $\{0,1\}^m$. We say that F is uniformly distributed if

$$\left|\{x \mid F(x) = \beta\}\right| = 2^{n-m}$$

for any $\beta \in \{0,1\}^m$.

Definition 33. We say that $F(x_1, \dots, x_n) = [f_1, \dots, f_m]$ is an (n, m)-SAC(k) function if any nonzero linear combination of $f_1, \dots, f_m$ satisfies SAC(k).

Definition 34. We say that $F(x_1, \dots, x_n) = [f_1, \dots, f_m]$ is an (n, m)-PC(l) of order k function if any nonzero linear combination of $f_1, \dots, f_m$ satisfies PC(l) of order k.

From Theorem 16, we obtain the following corollary.

Corollary 35. *Suppose that there exist $s \times t$ binary matrices $Q_1, \dots, Q_m$ such that any nonzero linear combination of $Q_1, \dots, Q_m$ satisfies the conditions of Theorem 16. For $1 \leq i \leq m$, let*

$$f_i(x_1, \dots, x_s, y_1, \dots, y_t) \triangleq [x_1, \dots, x_s] Q_i [y_1, \dots, y_t]^T \oplus g_i(x_1, \dots, x_s) ,$$

where g_i is any function. Then $F = [f_1, \dots, f_m]$ is an $(s + t, m)$-PC(l) of order k function.

Definition 36. For $F(x_1, \ldots, x_n) = [f_1, \ldots, f_m]$, define

$$\deg(F) \triangleq \min \deg(a_1 f_1 \oplus \cdots \oplus a_m f_m),$$
$$N(F) \triangleq \min N(a_1 f_1 \oplus \cdots \oplus a_m f_m),$$

where min is taken over all nonzero binary vectors $[a_1, \ldots, a_m]$.

Corollary 37. *In Corollary 35,*

(1) let $g_i = x_1 \ldots x_s / x_i$. Then $\deg(F) = s - 1$ if $m \leq s$.
(2) For $s =$ even and $m \leq s/2$, let $[g_1, \ldots, g_m]$ be a vector output bent function given by [9]. Then $N(f) \geq 2^{t+s-1} - 2^{t+s/2-1}$.
(3) If $s =$ odd and $m \leq (s-1)/2$, we can obtain $N(f) \geq 2^{t+s-1} - 2^{t+(s-1)/2}$.

The following corollary is obtained from Theorem 19.

Corollary 38. *Suppose that there exist*

(1) a linear $[t, h, k+1]$ code with the dual minimum Hamming distance at least $l+1$ and
(2) m matrices $G_{2,1}, \ldots G_{2,m}$ such that any nonzero linear combination of them is a generator matrix of a linear $[s, h, k+1]$ code with the dual minimum Hamming distance at least $l+1$.

Let $Q_i \triangleq G_{2,i}^T G_1$ for $1 \leq i \leq m$. Then $Q_1, \ldots, Q_m$ satisfy the condition of Corollary 35.

6.2 Vector Output PC(2) of Order k

Proposition 39. *[9] Consider a linear feedback shift register of length r and with a primitive feedback polynomial. Let D be the state transition function of such a shift register. Then D is a permutation of the space Z_2^r as well as the powers D^i of D, where*

$$D^i \triangleq D \circ \cdots \circ D, \; i = 1, 2, \ldots \; .$$

Moreover, any nonzero linear combination of $I, D, D^2, \ldots, D^{r-1}$ is also a permutation.

Lemma 40. *For any $r \geq 2$, there exist matrices $G_{2,1}, \ldots, G_{2,r}$ such that any nonzero linear combination of them is a generator matrix of the $[2^r - 1, r, 2^{r-1}]$ simplex code.*

Proof. Let $[i_1, \ldots, i_r]$ be the binary representation of i.

(1) Let $G_{2,1}$ be a $r \times (2^r - 1)$ matrix such that the i-th column vector is $[i_1, \ldots, i_r]^T$.

(2) For $2 \leq j \leq r$, let $G_{2,j}$ be a $r \times (2^r - 1)$ matrix such that the i-th column vector is $D^{j-1}(i_1, \ldots, i_r)$.

Then any nonzero linear combination of $G_{2,1}, \ldots, G_{2,r}$ is a parity check matrix of a $[2^r-1, 2^r-1-r, 3]$ Hamming code by Proposition 39. Equivalently, any nonzero linear combination of $G_{2,1}, \ldots, G_{2,r}$ is a generator matrix of a $[2^r - 1, r, 2^{r-1}]$ simplex code. □

Theorem 41. *For $r \geq 2$,*

(1) there exists a $(2^{r+1} - 2, r)$-PC(2) of order $2^{r-1} - 1$ function F with

$$\deg(F) = 2^r - 2 \ .$$

(2) there exists a $(2^{r+1} - 2, r)$-PC(2) of order $2^{r-1} - 1$ function F with

$$N(F) \geq 2^{2^{r+1}-3} - 2^{3\cdot 2^{r-1}-2} \ .$$

Proof. First, there exists a $[2^r - 1, r, 2^{r-1}]$ simplex code (see Proposition 20). Next, there exist matrices $G_{2,1}, \ldots, G_{2,r}$ such that any nonzero linear combination of them is a generator matrix of a $[2^r - 1, r, 2^{r-1}]$ simplex code from Lemma 40. Finally, the dual Hamming distance of a $[2^r - 1, r, 2^{r-1}]$ simplex code is 3. Hence, the conditions of Corollary 38 are satisfied.

Finally, apply Corollary 37 with $s = t = 2^r - 1$. □

6.3 Vector Output SAC(k)

Theorem 42. *For any $s > 0$,*

(1) there exists a $(2s, s-1)$-SAC(1) function F with $\deg(F) = s - 1$.
(2) there exists a $(2s, s-1)$-SAC(1) function F with

$$N(F) \geq \begin{cases} 2^{2s-1} - 2^{3s/2-1} & \text{if } s = \text{even} \\ 2^{2s-1} - 2^{(3s-1)/2} & \text{if } s = \text{odd} \ . \end{cases}$$

Proof. Let $I = (e_1, \ldots, e_s)$ be the $s \times s$ identity matrix and let P be a permutation matrix such that $P = (e_s, e_1, e_2, \ldots, e_{s-1})$. Define

$$Q_i = P^{(i-1)}(I + P) \tag{13}$$

for $1 \leq i \leq s - 1$. We show that $Q_1, \ldots, Q_{s-1}$ satisfy the condition of Corollary 35, that is the conditions of Theorem 16 with $s = t$. Let

$$Q = a_1 Q_1 + \cdots + a_{s-1} Q_{s-1} \ ,$$

where $[a_1, \ldots, a_{s-1}] \neq [0, \ldots, 0]$. Let q_i be the i-th column vector of Q and p_i be the i-th row vector of Q. Without loss of generality, we can assume that

(1) $a_1 = \cdots = a_{s-1} = 1$ or
(2) $a_1 = \cdots = a_j = 1$ and $a_{j+1} = 0$ for some $1 \leq j \leq s - 2$.

In case 1,

$$Q = I + P^{s-1} \ .$$

In case 2,

$$Q = I + P^j + X \ ,$$

where X cancels no elements of $I + P^j$. In any case, $W(q_i) \geq 2$ for any i and $W(p_i) \geq 2$ for any i. Thus, the conditions of Theorem 16 are satisfied for $l = 1$.

Finally, apply Corollary 37. □

Theorem 42 can be generalized as follows.

Theorem 43. *For any $k \geq 0$ and any $s \geq k+1$, let*

$$\gamma \triangleq \lceil (k+1)/2 \rceil \ , \ m \triangleq \lfloor (s-k-1)/\gamma + 1 \rfloor \ .$$

Then

(1) there exists a $(2s, m)$-SAC(k) function F with $\deg(F) = s - 1$.
(2) there exists a $(2s, m)$-SAC(k) function F with

$$N(F) \geq \begin{cases} 2^{2s-1} - 2^{3s/2-1} & \text{if } s = \text{even} \\ 2^{2s-1} - 2^{(3s-1)/2} & \text{if } s = \text{odd} \ . \end{cases}$$

Remark. In [3], we showed that there exists an (n, m)-SAC(k) function F if there exists a linear $[N, m, k+1]$ code such that

$$N = \begin{cases} n-1 & \text{if } n \text{ is even} \\ n-2 & \text{if } n \text{ is odd} \ . \end{cases} \tag{14}$$

In this construction,

(1) $\deg(F)$ and $N(F)$ are small. Actually, $\deg(F) = 2$.
(2) However, m can be larger than that of Theorem 42 and Theorem 43.

In other words, there is a tradeoff between the construction of [3] and Theorem 42 and Theorem 43 of this paper.

Acknowledgments

We would like to thank the anonymous referees for helpful comments. Especially, lemma 4.1 was improved.

References

1. R. Forré. The strict avalanche criterion : spectral properties of Boolean functions and an extend definition. In *Advances in Cryptology — CRYPTO '88 Proceedings, Lecture Notes in Computer Science* 403, pages 450–468. Springer-Verlag, 1990.
2. T. Jakobsen and L.R. Knudsen. The interpolation attack on block ciphers. In *Preproc. of Fast Software Encryption*, pages 28–40. January, 1997.
3. K. Kurosawa and T. Satoh. Generalization of higher order SAC to vector output Boolean functions. In *Advances in Cryptology — ASIACRYPT '96 Proceedings, Lecture Notes in Computer Science* 1163, pages 218–231. Springer-Verlag, 1996.
4. S. Lidl and Niederreiter. *Finite Fields, Encyclopedia of Mathematics and Its Applications 20*. Cambridge University Press, 1983.
5. S. Lloyd. Counting binary functions with certain cryptographic properties. *Journal of Cryptology*, 5:107–131, 1992.
6. F. J. MacWilliams and N. J. A. Sloane. *The theory of error-correcting codes*. North-Holland Publishing Company, 1977.
7. M. Matsui. Linear cryptanalysis method for DES cipher. In *Advances in Cryptology — EUROCRYPT '93 Proceedings, Lecture Notes in Computer Science* 765, pages 386–397. Springer-Verlag, 1994.
8. W. Meier and O. Staffelbach. Nonlinearity criteria for cryptographic functions. In *Advances in Cryptology — EUROCRYPT '89 Proceedings, Lecture Notes in Computer Science* 434, pages 549–562. Springer-Verlag, 1990.
9. K. Nyberg. Perfect nonlinear S-boxes. In *Advances in Cryptology — EUROCRYPT '91 Proceedings, Lecture Notes in Computer Science* 547, pages 378–386. Springer-Verlag, 1991.
10. J. Pieprzyk and G. Finkelstein. Towards effective nonlinear cryptosystem design. *IEE Proceedings Part E*, 35(6):325–335, November 1988.
11. B. Preneel, W. Van Leekwijck, L. Van Linden, R. Govaerts, and J. Vandewalle. Propagation characteristics of Boolean functions. In *Advances in Cryptology — EUROCRYPT '90 Proceedings, Lecture Notes in Computer Science* 473, pages 161–173. Springer-Verlag, 1991.
12. B. Preneel, R. Govaerts, and J. Vandewalle. Boolean functions satisfying higher order propagation criteria. In *Advances in Cryptology — EUROCRYPT '91 Proceedings, Lecture Notes in Computer Science* 547, pages 141–152. Springer-Verlag, 1991.
13. O. S. Rothaus. On bent functions. *Journal of Combinatorial Theory (A)*, 20:300–305, 1976.
14. J. Seberry and X.M. Zhang. Highly nonlinear 0-1 balanced Boolean functions satisfying strict avalanche criterion. In *Advances in Cryptology — AUSCRYPT '92 Proceedings, Lecture Notes in Computer Science* 718. Springer-Verlag, 1993.
15. A. F. Webster and S. E. Tavares. On the design of S-boxes. In *Advances in Cryptology — CRYPTO '85 Proceedings, Lecture Notes in Computer Science* 218, pages 523–534. Springer-Verlag, 1986.
16. A. M. Youssef, T. W. Cusick, P. Stănică, and S. E. Tavares. New bounds on the number of functions satisfying the strict avalanche criterion. In *Third Annual Workshop on Selected Areas in Cryptography*, 1996.