# Fourier Analysis of Boolean Maps
# – A Tutorial –

Klaus Pommerening
Fachbereich Mathematik
der Johannes-Gutenberg-Universitaet
Saarstrasse 21
D-55099 Mainz

May 30, 2000 – last revision May 12, 2005

Boolean functions and maps are of central importance in cryptology. They are the basic building blocks of bitblock and bitstream ciphers. An essential criterion of their usefulness is nonlinearity: The S-boxes for bitblock ciphers and the combiners of linear shift registers for bitstream ciphers should be as nonlinear as possible. What does that mean, and how can we measure nonlinearity?

In the past years cryptologists introduced several measures that complement each other in revealing hidden linearity. If a Boolean map is close to linear by such a measure, then there is an attack on the cipher system that uses this map as a building block. The most prominent examples are *linear* and *differential cryptanalysis* of bitblock ciphers and *correlation analysis* of bitstream ciphers.

The most important mathematical tool for the study of hidden linearity is the Walsh (or Hadamard) transform, the characteristic 2 special case of the discrete Fourier transform. Its systematic use leads to a uniform, elegant, and efficient treatment of nonlinearity; although it is conceptually very simple, it is surprisingly powerful for theoretical as well as for practical purposes, it almost magically reduces many proofs from the original papers to a few lines, and it makes computing linear and differential profiles—or linear approximation tables and difference tables—an unmitigated pleasure.

This article gives a short, but complete exposition of the essential parts of the theory of nonlinearity of Boolean maps. It is a short version of a tutorial in german language that contains much more material and many examples. The german version is in the web under

`http://www.staff.uni-mainz.de/pommeren/Kryptologie/Bitblock/A_Nonlin/`

It also contains a comprehensive bibliography. Furthermore the C sources of all the procedures and of the entire analysis program (with english annotations) can be found there as well as an online call; look at

http://www.staff.uni-mainz.de/pommeren/Kryptologie/Bitbl-e.html

# 1 The Algebraic Normal Form

Booelan maps can be expressed by polynomials—this is the algebraic normal form (ANF). The degree as a polynomial is a first obvious measure of nonlinearity—linear (or affine) maps have degree 1.

In this section we show how to determine the ANF and the degree of a Boolean map that is given by its value table.

## 1.1 Boolean functions and maps

We denote by $\mathbb{F}_2$ the Galois field with two elements. We use algebraic notation: $+$ is the addition in the field $\mathbb{F}_2$ and in vector spaces over $\mathbb{F}_2$. We reserve the character $\oplus$ for direct sums.

A **Boolean function** of $n$ variables is a function

$$f\colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2.$$

A **Boolean map** (or vector valued Boolean function) is a map

$$f\colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q.$$

Cryptologists like to call this an "S-box" or "substitution box".

Let $\mathcal{F}_n$ be the set of all Boolean functions on $\mathbb{F}_2^n$. We identify the set of all mappings $\mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ with $\mathcal{F}_n^q$ in a natural way.

Usually a Boolean function is given by its **truth table**—that is by its value table or the graph of the function. This table is—in a canonical way—lexicographically ordered by the arguments $x \in \mathbb{F}_2^n$. In other words, by the natural order of the numbers $a = 0, \ldots, 2^n - 1$, represented in base 2 as

$$a = x_1 \cdot 2^{n-1} + \cdots + x_{n-1} \cdot 2 + x_n$$

and identified with the vectors $(x_1, \ldots, x_n) \in \mathbb{F}_2^n$.

The logical negation of the function $f \in \mathcal{F}_n$ is the function $\bar{f} = f + 1$.

Let $\mathcal{L}_n$ be the set of all linear forms, that is the dual space of $\mathbb{F}_2^n$. Let $\{e_1, \ldots, e_n\}$ be the canonical basis of $\mathbb{F}_2^n$ and $\cdot$ the canonical dot product. The identification of the linear form $x \mapsto u \cdot x$ with the vector $u \in \mathbb{F}_2^n$ gives the (basis dependent) isomorphism $\mathbb{F}_2^n \cong \mathcal{L}_n$ of vector spaces.

Let $\mathcal{A}_n$ be the set of affine functions $\mathbb{F}_2^n \longrightarrow \mathbb{F}_2$. There are $2^{n+1}$ of them—the linear functions and their negations, that is the functions

$$f(x) = \alpha(x) + c \qquad \text{where } \alpha \in \mathcal{L}_n \text{ and } c \in \mathbb{F}_2.$$

Let $\chi : \mathbb{F}_2 \longrightarrow \mathbb{C}^\times$ be the only nontrivial group homomorphism ("character"): $\chi(0) = 1$, $\chi(1) = -1$, or $\chi(a) = (-1)^a = 1 - 2a$, the last equation "par abus de notation" (identifying $1 \in \mathbb{F}_2$ with $1 \in \mathbb{R}$). In particular $\chi$

is real valued. With each Boolean function $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ we associate its **character form** as $\chi_f := \chi \circ f : \mathbb{F}_2^n \longrightarrow \mathbb{R}^\times \subseteq \mathbb{C}^\times$, in short

$$\chi_f(x) = (-1)^{f(x)}.$$

Obviously $\chi_{f+g} = \chi_f \chi_g$. The formula for the product of two Boolean function is slightly more complicated. From the table

| $a$ | $b$ | $a+b$ | $ab$ | $\chi(a)$ | $\chi(b)$ | $\chi(a+b)$ | $\chi(ab)$ |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | −1 | −1 | 1 |
| 1 | 0 | 1 | 0 | −1 | 1 | −1 | 1 |
| 1 | 1 | 0 | 1 | −1 | −1 | 1 | −1 |

we get the formula

$$\chi(a+b) + 2\chi(ab) = 1 + \chi(a) + \chi(b) \quad \text{for all } a, b \in \mathbb{F}_2.$$

Therefore for $f, g \in \mathcal{F}_n$ we have the product formula

$$2\chi_{fg} = 1 + \chi_f + \chi_g - \chi_f \chi_g.$$

**Definition 1** For two Boolean functions $f, g : \mathbb{F}_2^n \to \mathbb{F}_2$ the **Hamming distance** is the number of arguments where the functions differ:

$$d(f,g) := \#\{x \in \mathbb{F}_2^n \mid f(x) \neq g(x)\};$$

in other words: the number of ones in the truth table of $f + g$.

**Remarks**

1. $d$ is a metric on $\mathcal{F}_n$. The transitivity of $d$ for $f, g, h \in \mathcal{F}_n$ is shown as follows: If $f(x) \neq h(x)$, then $f(x) \neq g(x)$ or $g(x) \neq h(x)$; therefore

$$\begin{aligned} d(f,g) + d(g,h) &= \#\{x \mid f(x) \neq g(x)\} + \#\{x \mid g(x) \neq h(x)\} \\ &\geq \#\{x \mid f(x) \neq h(x)\} = d(f,h). \end{aligned}$$

2. If $\bar{g} = g + 1$ is the negation of $g$, then $d(f, \bar{g}) = 2^n - d(f,g) =$ the number of arguments, where $f$ and $g$ coincide.

## 1.2 Boolean linear forms

For $u, x \in \mathbb{F}_2^n$ we can write the canonical dot product as

$$u \cdot x = \sum_{i=1}^{n} u_i x_i = \sum_{u_i = 1} x_i = \sum_{i \in \mathrm{Supp}(u)} x_i$$

with the "support" of $u$,

$$\mathrm{Supp}(u) = \{i = 1, \ldots, n \mid u_i \neq 0\} = \{i = 1, \ldots, n \mid u_i = 1\}.$$

This means that the dot product with a fixed vector $u$ is the partial sum over the coordinates of $x$ in the support $I \subseteq \{1, \ldots, n\}$ of $u$ or the **parity** of $x$ over $I$. Since every linear form on a finite dimensional vector space can be written as a dot product with a fixed vector, we have shown:

**Proposition 1** *The linear forms on $\mathbb{F}_2^n$ are the parity functions over the subsets $I \subseteq \{1, \ldots, n\}$.*

In other words every linear form can be written as

$$\alpha_I(x) = \sum_{i \in I} x_i \qquad \text{for all } x = (x_1, \ldots, x_n) \in \mathbb{F}_2^n$$

with a subset $I \subseteq \{1, \ldots, n\}$. Thereby we have a natural bijection between the $2^n$ element set $\mathcal{L}_n$ and the power set $\mathfrak{P}(\{1, \ldots, n\})$.

Other common expressions are—for $I = \{i_1, \ldots, i_r\}$—:

$$\alpha_I(x) = x[I] = x[i_1, \ldots, i_r] = x_{i_1} + \cdots + x_{i_r}.$$

### 1.3 Functions and polynomials

Let $T = (T_1, \ldots, T_n)$ be an $n$-tuple of indeterminates. Every polynomial $p \in \mathbb{F}_2[T]$ defines a function $\Psi(p) \in \mathcal{F}_n$ by substitution:

$$\Psi(p)(x_1, \ldots, x_n) := p(x_1, \ldots, x_n).$$

The **substitution homomorphism**

$$\Psi : \mathbb{F}_2[T] \longrightarrow \mathcal{F}_n,$$

is a homomorphism of $\mathbb{F}_2$-algebras.

**Lemma 1** *$\Psi$ is surjective.*

*Proof.* (By induction over $n$) The induction begin $n = 0$ is trivial—the two constant polynomials correspond to the two constant functions. Now let $n \geq 1$. Let $x' = (x_1, \ldots, x_{n-1}) \in \mathbb{F}_2^{n-1}$ be the first $n-1$ components of $x = (x_1, \ldots, x_n) \in \mathbb{F}_2^n$. Consider a function $f \in \mathcal{F}_n$. By induction we have for $b = 0, 1$

$$f(x', b) = p_b(x') \qquad \text{for all } x' \in \mathbb{F}_2^{n-1}$$

where $p_0, p_1 \in \mathbb{F}_2[T_1, \ldots, T_{n-1}]$; in the case $n = 1$ these are constants. Then

$$f(x', x_n) = (1 + x_n)p_0(x') + x_n p_1(x') \qquad \text{for all } x = \in \mathbb{F}_2^n.$$

Therefore $f = \Psi(p)$ with $p = p_0 + (p_0 + p_1)T_n$. $\diamond$

**Note.** An analogous statement holds over any finite field. The proof in the general case is slightly more complicated and uses interpolation; this is useful in cryptology too, since—over the Galois field $\mathbb{F}_{2^n}$—it is the basis for interpolation attacks on block ciphers. The following proposition 2 generalizes in the same way.

What is the kernel of the homomorphism $\Psi$? Since $b^2 = b$ for all $b \in \mathbb{F}_2$, all the polynomials $T_1^2 - T_1, \ldots, T_n^2 - T_n$ are in the kernel, so is the ideal

$$\mathfrak{a} \trianglelefteq \mathbb{F}_2[T]$$

they generate. The induced homomorphism

$$\bar{\Psi} : \mathbb{F}_2[T]/\mathfrak{a} \longrightarrow \mathcal{F}_n$$

is surjective. Each element of the factor algebra $\mathbb{F}_2[T]/\mathfrak{a}$ can be written as a linear combination of the monomials that have a degree $\leq 1$ in each $T_i$. There are $2^n$ of these, namely the products

$$T^I := T_{i_1} \cdots T_{i_r}$$

for arbitrary subsets

$$I = \{i_1, \ldots, i_r\} \subseteq \{1, \ldots, n\}.$$

So the pre-image of $\bar{\Psi}$ is a vector space of dimension $\leq 2^n$ over $\mathbb{F}_2$. Therefore its dimension must be $= 2^n$, and $\bar{\Psi}$ must be an isomorphism. We have shown:

**Proposition 2 (Algebraic normal form, ANF)** *Every Boolean function*

$$f \colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$$

*can uniquely be written as a polynomial in n indeterminates that has degree* $\leq 1$ *in each single indeterminate:*

$$f(x_1, \ldots, x_n) = \sum_{I \subseteq \{1,\ldots,n\}} a_I x^I,$$

*where the monomial* $x^I$ *is the product*

$$x^I = \prod_{i \in I} x_i,$$

*and* $a_I = 0$ *or* $1$.

| $k \in \mathbb{N}$ | $u \in \mathbb{F}_2^3$ | $I \subseteq \{1,2,3\}$ | monomial |
|---|---|---|---|
| 0 | 000 | $\emptyset$ | 1 |
| 1 | 001 | $\{3\}$ | $T_3$ |
| 2 | 010 | $\{2\}$ | $T_2$ |
| 3 | 011 | $\{2,3\}$ | $T_2T_3$ |
| 4 | 100 | $\{1\}$ | $T_1$ |
| 5 | 101 | $\{1,3\}$ | $T_1T_3$ |
| 6 | 110 | $\{1,2\}$ | $T_1T_2$ |
| 7 | 111 | $\{1,2,3\}$ | $T_1T_2T_3$ |

Table 1: Various interpretations of a binary vector, example $n = 3$

The various interpretations of a binary vector $u \in \mathbb{F}_2^n$ are illustrated by a simple example in table 1, as well as the "canonical"' associations between them.

There is an alternative derivation of the algebraic normal form using normalization of Boolean expressions. This however doesn't generalize to other finite fields.

**Corollary 1** *Every Boolean map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ is given by a q-tuple of polynomials $(p_1, \ldots, p_q) \in \mathbb{F}_2[T_1, \ldots, T_n]$ that have all partial degrees $\leq 1$.*

(By "partial degree" we mean the degree in a single indeterminate $T_i$.)

**Corollary 2** *Every Boolean map $f \colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ has a unique expression as*

$$f(x_1, \ldots, x_n) = \sum_{I \subseteq \{1,\ldots,n\}} x^I a_I$$

*with coefficients $a_I \in \mathbb{F}_2^q$.*

**Definition 2** The degree of a Boolean map $f$ as a polynomial,

$$\operatorname{Deg} f = \max\{\#I \mid a_I \neq 0\},$$

is called **(algebraic) degree** of $f$.

**Remarks**

1. In general $\operatorname{Deg} f \leq n$.
2. $f$ is affine $\Leftrightarrow \operatorname{Deg} f \leq 1$.
3. The degree of a Boolean map $f$ is the maximum of the degrees of its component polynomials $p_1, \ldots, p_q$.

The algebraic degree is a first measure of nonlinearity of $f$. It is obviously invariant under affine transformations in preimage and image.

**Exercise** How many functions $\mathbb{F}_2^2 \longrightarrow \mathbb{F}_2$ are there? Enumerate them all.

## 1.4 Evaluation and interpolation

The algebraic normal form has the advantage that we can immediately read off the algebric degree. We also easily recognize the "structure" of a Boolean map, moreover we could comfortably classify orbits under the action of affine transforms and find "reduced" normal forms. On the other hand the truth table more clearly shows the "behaviour" of a map, and—as we shall see—leads to better detection of hidden linearity.

Therefore a method of switching between these two representations is hihgly needed. The transition from the algebraic normal form to the truth table is the evaluation of the component polynomials at all arguments. The inverse transformation is interpolation as in the proof of proposition 2. Here we show how to do this by an efficient algorithm.

The naive evaluation of a Boolean function $f \in \mathcal{F}_n$, or $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$, at all arguments $x \in \mathbb{F}_2^n$ costs $2^n$ evaluations $f(x)$ each with up to $2^n$ summands with up to $n-1$ multiplications. This makes an order of magnitude of $n \cdot 2^n \cdot 2^n$; since the size of the input is $N = 2^n$, the expense is essentially quadratic: $N^2 \cdot {}^2\log(N)$. A binary recursion or "divide and conquer strategy"—a recursive division in two partial tasks of half the input size—will lead to a significantly faster algorithm.

To begin with let's write the algebraic normal form in a slightly modifyed way:

$$f = \sum_{u \in \mathbb{F}_2^n} \alpha_f(u) T^{(u)} \qquad \text{with the monomial} \quad T^{(u)} = \prod_{i \in \mathrm{Supp}(u)} T_i.$$

The association between a binary vector $u$ and a monomial $T^{(u)}$ is as in table 1. Then the $\alpha_f(u)$ themselves form a function $\alpha_f \in \mathcal{F}_n$ that we call the **coefficient representation** of $f$. On the other hand the truth table is represented by the family $(f(x))_{x \in \mathbb{F}_2^n}$, that means simply by $f \in \mathcal{F}_n$ itself. With this interpretation the evalution is the transformation

$$\Theta_n : \mathcal{F}_n \longrightarrow \mathcal{F}_n, \qquad \alpha_f \mapsto f.$$

The binary recursion starts with the unique decomposition

$$f = f_0 + T_1 f_1 \qquad \text{with} \quad f_0, f_1 \in \mathbb{F}_2[T_2, \ldots, T_n]$$

from the proof of lemma 1 (where the notation was different). Then for $y \in \mathbb{F}_2^{n-1}$ we have

$$\begin{aligned} f(0,y) &= f_0(y), \\ f(1,y) &= f_0(y) + f_1(y). \end{aligned}$$

In general let $0 \leq i \leq n$, $u \in \mathbb{F}_2^{n-i}$, and $f_u \in \mathbb{F}_2[T_{n-i+1}, \ldots, T_n]$ be defined by

$$f_u := \sum_{v \in \mathbb{F}_2^i} \alpha_f(u,v) T^{(v)}.$$

Then in the case $i = n$ and $u = 0 \in \mathbb{F}_2^0$

$$f_u = \sum_{v \in \mathbb{F}_2^n} \alpha_f(v) T^{(v)} = f.$$

On the other extreme, in the case $i = 0$ and $u \in \mathbb{F}_2^n$, we have

$$f_u = \alpha_f(u) \quad \text{constant.}$$

In between, for $1 \leq i \leq n$ and $u \in \mathbb{F}_2^{n-i}$, we have

$$f_u = f_{(u,0)} + T_{n-i+1} f_{(u,1)}.$$

Therefore the evaluation follows the recursion (for $y \in \mathbb{F}_2^{i-1}$)

$$\begin{aligned} f_u(0,y) &= f_{(u,0)}(y), \\ f_u(1,y) &= f_{(u,0)}(y) + f_{(u,1)}(y). \end{aligned}$$

We convert this recursion into an iterative procedure by defining a sequence of vectors $x^{(i)} = (x_u^{(i)})_{u \in \mathbb{F}_2^n}$ with coefficients in $\mathbb{F}_2$ as follows: Let

$$x^{(0)} := (\alpha_f(u))_{u \in \mathbb{F}_2^n},$$

be the initial vector. For $i = 1, \ldots, n$ let the $n$-bit index decompose into $u\xi v$ with $n - i$ bits $u$, one bit $\xi$, and $i - 1$ bits $v$, and let

$$\begin{aligned} x_{u0v}^{(i)} &:= x_{u0v}^{(i-1)}, \\ x_{u1v}^{(i)} &:= x_{u0v}^{(i-1)} + x_{u1v}^{(i-1)}. \end{aligned}$$

By induction we get:

**Proposition 3** *Let $(x^{(i)})$ be the recursively defined sequence as above. Then*

$$x_{(u,y)}^{(i)} = f_u(y) \qquad \textit{for all} \quad u \in F_2^{n-i}, y \in \mathbb{F}_2^i;$$

*in particular*

$$x^{(n)} = (f(u))_{u \in \mathbb{F}_2^n}$$

*is the truth table of $f$.*

In the opposite direction the iteration has the same structure:

$$\begin{aligned} x^{(i-1)}_{u0v} &:= x^{(i)}_{u0v}, \\ x^{(i-1)}_{u1v} &:= x^{(i)}_{u0v} + x^{(i)}_{u1v}. \end{aligned}$$

Therefore the inverse transformation of $\Theta_n$, the construction of the coefficient representation given the truth table, follows exactly the same procedure, so it must be identical to $\Theta_n$:

**Corollary 1** *The evaluation transformation $\Theta_n$ is an involution of $\mathcal{F}_n$.*

In particular *the inverse application of $\Theta_n$ also determines the algebraic degree of a Boolean function* that is given by its truth table.

To implement the evaluation procedure in a common programming language we interpret the indices as natural numbers $k = \sum k_{n-i}2^i$ in the integer interval $[0 \dots 2^n - 1]$ as in table 1. Then in the iteration formula we have $u1v = u0v + 2^i$, and the equations become

$$x^{(i+1)}_k = \begin{cases} x^{(i)}_k, & \text{if } k_{n-i} = 0, \\ x^{(i)}_{k-2^i} + x^{(i)}_k, & \text{if } k_{n-i} = 1, \end{cases}$$

for $k = 0, \dots, 2^n - 1$. The bit $k_{n-i}$ is extracted from $k$ by the formula

$$k_{n-i} = \left\lfloor \frac{k}{2^i} \right\rfloor \bmod 2 = (k \gg i) \bmod 2,$$

where $k \gg i$ is the $i$ bit shift to the right. Now the entire algorithm in "pidgin Pascal" looks as follows:

**Prozedur [REV]** (Recursive evaluation)
**Input and output parameters:** A vector $x$ of length $2^n$, $x[0], \dots, x[2^n - 1]$.

**Local variables:** A vector $y$ of length $2^n$, $y[0], \dots, y[2^n - 1]$.
Loop counters $i = 0, \dots, n-1$ and $k = 0, \dots, 2^n - 1$.

**Instructions:**
For $i = 0, \dots, n-1$:
For $k = 0, \dots, 2^n - 1$:
If $(((k \gg i) \bmod 2) = 1)$ then $y[k] := x[k - 2^i]$ XOR $x[k]$
else $y[k] := x[k]$
For $k = 0, \dots, 2^n - 1$:
$x[k] := y[k]$

Here $x$ and $y$ are vectors over $\mathbb{F}_2$, that means bit sequences, and the addition in $\mathbb{F}_2$ is represented by the Boolean operator XOR.

The expense is $n \cdot 2^n$ loop executions, each with one binary addition, one shift, and one single bit complement. In summary that makes $3n \cdot 2^n$ "elementary" operations. The procedure essentially needs $2 \cdot 2^n$ bits of memory. Expressed as a function if the input size $N = 2^n$, the expense is almost linear: $3N \cdot {}^2\log N$.

The corresponding C procedure in the sources is called `rev`.

# 2 The Walsh Transformation

For the moment we consider *real valued* functions $\varphi : \mathbb{F}_2^n \longrightarrow \mathbb{R}$. These make up the $\mathbb{R}$-algebra $\mathcal{C}_n = \mathbb{R}^{\mathbb{F}_2^n}$.

## 2.1 Definition of the Walsh transformation

**Definition 1** The **Walsh Transformation** (or Hadamard-Walsh-Transformation)

$$\Phi : \mathcal{C}_n \longrightarrow \mathcal{C}_n, \qquad \varphi \mapsto \hat{\varphi},$$

is defined by the formula

$$\hat{\varphi}(u) := \sum_{x \in \mathbb{F}_2^n} \varphi(x) \cdot (-1)^{u \cdot x}$$

(where $u \cdot x$ is the canonical dot product in $\mathbb{F}_2^n$).

**Remarks**

1. Obviously $\Phi$ is a $\mathbb{R}$-linear map.
2. $\Phi$ is a special case of the discrete Fourier transformation. In the general case, instead of $-1$ in the formula one takes the complex $N$-th root of unity $\zeta = e^{2\pi i/N}$, and transforms complex valued functions over the ring $\mathbb{Z}/N\mathbb{Z}$—or functions on $\mathbb{Z}^n$, that have period $N$ in each variable. [Character sums are a further generalization.]
3. Clearly $\hat{0} = 0$ for the constant function $0 \in \mathcal{C}_n$. The other constant function 1 tranforms to $\hat{1}$ = the "point mass" in 0:

$$\begin{aligned} \hat{1}(0) &= 2^n, \\ \hat{1}(u) &= 0 \quad \text{else.} \end{aligned}$$

This follows from the next lemma:

**Lemma 1** *For $u \in \mathbb{F}_2^n$ we have*

$$\sum_{x \in \mathbb{F}_2^n} (-1)^{u \cdot x} = \begin{cases} 2^n, & \text{if } u = 0, \\ 0 & \text{else.} \end{cases}$$

*Proof.* If $u = 0$, then all exponents are 0, all summands 1, and we have $2^n$ of them.

For $u \neq 0$ let $H$ be the hyperplane $\{x \in \mathbb{F}_2^n \mid x \cdot u = 0\}$. Then $\bar{H} = \{x \in \mathbb{F}_2^n \mid x \cdot u = 1\}$ is its complement, hence $\mathbb{F}_2^n = H \bigcup \bar{H}$, $H \bigcap \bar{H} = \emptyset$, and $\#H = \#\bar{H} = 2^{n-1}$. For $x \in H$ the summand is 1, for $x \in \bar{H}$ it's $-1$. Therefore the sum is 0. $\diamond$

**Definition 2** For a Boolean function $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ the transformed function $\hat{\chi_f} : \mathbb{F}_2^n \longrightarrow \mathbb{R}$ of the character form $\chi_f$ is called the **(Walsh) spectrum** of $f$.

We have

$$\begin{aligned}\hat{\chi}_f(u) &= \sum_{x \in \mathbb{F}_2^n} \underbrace{(-1)^{f(x)+u\cdot x}}_{\begin{cases} 1, & \text{if } f(x) = u \cdot x, \\ -1, & \text{if } f(x) \neq u \cdot x, \end{cases}} \\ &= \#\{x \mid f(x) = u \cdot x\} - \#\{x \mid f(x) \neq u \cdot x\}.\end{aligned}$$

If we denote the first of these sets by

$$L_f(u) := \{x \mid f(x) = u \cdot x\}$$

then we have shown:

**Corollary 1** *The spectrum of a Boolean function $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ equals*

$$\hat{\chi}_f(u) = 2 \cdot \#L_f(u) - 2^n.$$

*In particular $\hat{\chi}_f(u)$ is always even, and*

$$-2^n \leq \hat{\chi}_f(u) \leq 2^n.$$

The lower bound is taken for $f(x) = u \cdot x + 1$, the upper one for $f(x) = u \cdot x$. In general the spectrum reflects the coincidence or deviation between a Boolean function and all linear and affine functions.

**Corollary 2** *Let $\alpha$ be the linear form $\alpha(x) = u \cdot x$ corresponding to $u$. Then*

$$d(f, \alpha) = 2^n - \#L_f(u) = 2^{n-1} - \frac{1}{2}\,\hat{\chi}_f(u).$$

**Remarks**

4. $\hat{\chi}_{f+1} = -\hat{\chi}_f$ for all $f$.

**Exercise 1** How does the spectrum change under an affine transformation of the argument space?

**Exercise 2** Calculate the spectrum of an affine function and of the function $f(x_1, x_2) = x_1 x_2$ of two variables.

## 2.2 The inversion formula

Let's apply the Walsh transformation $\Phi$ again to an already transformed function $\hat{\varphi}$:

$$\begin{aligned}
\hat{\hat{\varphi}}(w) &= \sum_{u\in\mathbb{F}_2^n} \hat{\varphi}(u)\cdot(-1)^{u\cdot w} \\
&= \sum_{u\in\mathbb{F}_2^n}\sum_{x\in\mathbb{F}_2^n} \varphi(x)\cdot(-1)^{u\cdot x}\cdot(-1)^{u\cdot w} \\
&= \sum_{x\in\mathbb{F}_2^n} \varphi(x) \underbrace{\left[\sum_{u\in\mathbb{F}_2^n}(-1)^{u\cdot(x+w)}\right]}_{=\begin{cases}2^n, & \text{if } x+w=0,\\ 0 & \text{else,}\end{cases}} \\
&= 2^n\varphi(w).
\end{aligned}$$

We have shown, that $\Phi\circ\Phi(\varphi) = 2^n\varphi$ for all $\varphi\in\mathcal{C}_n$:

**Proposition 1** *The Walsh transformation $\Phi : \mathcal{C}_n \longrightarrow \mathcal{C}_n$ is bijective, and it's inverse transformation is given by*

$$\Phi^{-1} = \frac{1}{2^n}\Phi.$$

**Corollary 1**

$$\varphi(0) = \frac{1}{2^n}\sum_{u\in\mathbb{F}_2^n}\hat{\varphi}(u).$$

**Corollary 2** *For every Boolean function $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ we have*

$$\sum_{u\in\mathbb{F}_2^n}\hat{\chi}_f(u) = \begin{cases}2^n, & \textit{if } f(0)=0,\\ -2^n & \textit{else.}\end{cases}$$

## 2.3 The convolution

**Definition 3** For $\varphi,\psi : \mathbb{F}_2^n \longrightarrow \mathbb{R}$ the **convolution** $\varphi * \psi : \mathbb{F}_2^n \longrightarrow \mathbb{R}$ is defined by

$$\varphi*\psi(w) := \sum_{x\in\mathbb{F}_2^n}\varphi(x)\psi(w-x).$$

This gives a bilinear map $* : \mathcal{C}_n\times\mathcal{C}_n \longrightarrow \mathcal{C}_n$.

Let's calculate the value at 0 for the convolution of the character forms of two Boolean functions $f, g\colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$:

$$\begin{aligned}\chi_f * \chi_g(0) &= \sum_{x\in\mathbb{F}_2^n} \chi_f(x)\chi_g(x)\\ &= \sum_{x\in\mathbb{F}_2^n} (-1)^{f(x)+g(x)}\\ &= 2^n - 2\cdot d(f,g),\end{aligned}$$

because

$$(-1)^{f(x)+g(x)} = \begin{cases}1, & \text{if } f(x)=g(x),\\ -1 & \text{else.}\end{cases}$$

Therefore $d(f,g)$ summands are $=-1$, and $2^n - d(f,g)$ summands are $=1$. We have shown the following generalization of corollary 2 in 2.1:

**Proposition 2** *The Hamming distance of two Boolean functions $f, g$ on $\mathbb{F}_2^n$ is*

$$d(f,g) = 2^{n-1} - \frac{1}{2}\,\chi_f * \chi_g(0).$$

Another way to express this result is in terms of the **correlation**,

$$\begin{aligned}\kappa(f,g) &:= \frac{1}{2^n}\left[\#\{x \mid f(x)=g(x)\} - \#\{x \mid f(x)\neq g(x)\}\right]\\ &= \frac{1}{2^{n-1}}\left[\#\{x \mid f(x)=g(x)\}\right] - 1.\end{aligned}$$

**Corollary 1** *The correlation of the functions $f$ and $g$ is*

$$\kappa(f,g) = \frac{1}{2^n}\cdot \chi_f * \chi_g(0).$$

**Exercise** Show: The correlation $\kappa$ is a scalar product on the real function space $\mathcal{C}_n$. The set $\{\chi_f \mid f \in \mathcal{L}_n\}$ of the character forms of the linear forms on $\mathbb{F}_2^n$ is an orthonormal basis of $\mathcal{C}_n$. The Walsh transformation of a function $f \in \mathcal{C}_n$ is its representation in this basis.

**Definition 4** The **autocorrelation** of a Boolean function $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ with respect to the shift $x \in \mathbb{F}_2^n$ is

$$\kappa_f(x) := \frac{1}{2^n}\left[\#\{u\in\mathbb{F}_2^n \mid f(u+x)=f(u)\} - \#\{u\in\mathbb{F}_2^n \mid f(u+x)\neq f(u)\}\right].$$

Therefore we have

$$\kappa_f(x) = \frac{1}{2^n}\cdot\sum_{u\in\mathbb{F}_2^n}(-1)^{f(u+x)+f(u)} = \frac{1}{2^n}\cdot\sum_{u\in\mathbb{F}_2^n}\chi_f(u+x)\chi_f(u),$$

hence

**Lemma 2** *The autocorrelation of $f$ is*

$$\kappa_f = \frac{1}{2^n} \cdot \chi_f * \chi_f.$$

Let's calculate the Walsh transform of a convolution:

$$\begin{aligned}
\widehat{\varphi * \psi}(u) &= \sum_{w \in \mathbb{F}_2^n} (\varphi * \psi)(w)(-1)^{u \cdot w} \\
&= \sum_{w \in \mathbb{F}_2^n} \sum_{x \in \mathbb{F}_2^n} \varphi(x)\psi(w+x)(-1)^{u \cdot w} \\
&= \sum_{x \in \mathbb{F}_2^n} \varphi(x) \left[ \sum_{w \in \mathbb{F}_2^n} \psi(w+x)(-1)^{u \cdot w} \right] \\
&= \sum_{x \in \mathbb{F}_2^n} \varphi(x) \sum_{v \in \mathbb{F}_2^n} \psi(v)(-1)^{u \cdot (v+x)} \\
&= \sum_{x \in \mathbb{F}_2^n} \varphi(x) \left[ \sum_{v \in \mathbb{F}_2^n} \psi(v)(-1)^{u \cdot v} \right] (-1)^{u \cdot x} \\
&= \left[ \sum_{x \in \mathbb{F}_2^n} \varphi(x)(-1)^{u \cdot x} \right] \hat{\psi}(u) \\
&= \hat{\varphi}(u)\hat{\psi}(u).
\end{aligned}$$

**Proposition 3 (Convolution theorem)** *For $\varphi, \psi : \mathbb{F}_2^n \longrightarrow \mathbb{R}$ we have $\widehat{\varphi * \psi} = \hat{\varphi}\hat{\psi}$.*

**Corollary 2** *$\mathcal{C}_n$, with $*$ as multiplication is a $\mathbb{R}$-algebra $\mathcal{C}_n^*$; in particular $*$ is commutative and associative, and $\Phi : \mathcal{C}_n \longrightarrow \mathcal{C}_n^*$ is a homomorphism of $\mathbb{R}$-algebras.*

Since $\Phi^{-1} = \frac{1}{2^n}\,\Phi$, up to the factor $2^n$ also $\Phi$ is a homomorphism $\mathcal{C}_n^* \longrightarrow \mathcal{C}_n$, in other words:

**Corollary 3** *For $\varphi, \psi \colon \mathbb{F}_2^n \longrightarrow \mathbb{R}$ we have $\widehat{\varphi\psi} = \frac{1}{2^n} \cdot \hat{\varphi} * \hat{\psi}$.*

**Corollary 4** *For $f, g \colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ we have*

$$\begin{aligned}
\widehat{\chi_{f+g}} &= \widehat{\chi_f \chi_g} = \frac{1}{2^n}\hat{\chi}_f * \hat{\chi}_g, \\
2\widehat{\chi_{fg}} &= \Phi(1 + \chi_f + \chi_g - \chi_f\chi_g) = \hat{1} + \hat{\chi}_f + \hat{\chi}_g - \frac{1}{2^n}\hat{\chi}_f * \hat{\chi}_g.
\end{aligned}$$

**Corollary 5** *The Walsh transform of the autocorrelation $\kappa_f$ is given by $\hat{\kappa}_f = \frac{1}{2^n}\hat{\chi}_f^2$.*

There are two ways to calculate the value of a convolution product at 0; first:

$$\varphi * \psi(0) = \sum_{x \in \mathbb{F}_2^n} \varphi(x)\psi(x).$$

Secondly, by the corollary 1 of the inversion formula (proposition 1):

$$\varphi * \psi(0) = \frac{1}{2^n} \sum_{u \in \mathbb{F}_2^n} \widehat{\varphi * \psi}(u) = \frac{1}{2^n} \sum_{u \in \mathbb{F}_2^n} \hat{\varphi}(u)\hat{\psi}(u).$$

We have shown:

**Proposition 4 (Parseval's equation)** *For $\varphi, \psi \colon \mathbb{F}_2^n \longrightarrow \mathbb{R}$*

$$\sum_{u \in \mathbb{F}_2^n} \hat{\varphi}(u)\hat{\psi}(u) = 2^n \sum_{x \in \mathbb{F}_2^n} \varphi(x)\psi(x).$$

## 2.4 Bent functions

Parseval's equation, applied to the character form of a Boolean function $f \colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ yields:

$$\sum_{u \in \mathbb{F}_2^n} \hat{\chi}_f(u)^2 = 2^n \cdot \sum_{x \in \mathbb{F}_2^n} \chi_f(x)^2 = 2^{2n},$$

because in the last sum all summands are $= 1$. Therefore in the first sum there must be at least one of the $2^n$ summands $\hat{\chi}_f(u)^2 \geq 2^n$. Hence:

**Proposition 5** *For every Boolean function $f \colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ we have*

$$max|\hat{\chi}_f| \geq 2^{n/2},$$

*with equality, if and only if $\hat{\chi}_f^2 = 2^n$ constant.*

These functions are well-known in combinatorics since many years:

**Definition 5** (Rothaus, ca 1965, published in 1976) A Boolean function $f \colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ is called bent, if $(\hat{\chi}_f)^2 = 2^n$ constant.

In particular the spektrum $\hat{\chi}_f$ of a bent function can only assume the values $\pm 2^{n/2}$; these must be integers:

$$\hat{\chi}_f(u) = \sum_{x \in \mathbb{F}_2^n} \chi_f(x)(-1)^{u \cdot x} \in \mathbb{Z}.$$

**Corollary 1** *If a bent function $f \colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ exists, then $n$ must be even.*

**Remarks**

1. The correlation of a Boolean function $f$ with the linear form $\alpha$ that corresponds to $u \in \mathbb{F}_2^n$ is

$$\kappa(f, \alpha) = \frac{1}{2^n} \cdot \hat{\chi}_f(u).$$

While constructing stream ciphers (or pseudorandom generators) by combining linear shift registers one tries to avoid correlations with linear functions. But because the sum of squares over all such correlations is constant = 1, the correlation 0 is possible only, if there are higher correlations with other linear forms. It's better to minimize all these correlationen in a uniform way, that means to minimize $\max |\hat{\chi}_f|$. That's what bent functions fulfil.

**Exercise 1** Find a bent function of 2 or 4 variables.

**Exercise 2** Show that for every bent function $f$ there is a bent function $g$ such that $\hat{\chi}_f = 2^{n/2}\chi_g$. (Duality of bent functions.)

**Exercise 3** Let $n = 2m$ be even. Let $\pi : \mathbb{F}_2^m \longrightarrow \mathbb{F}_2^m$ be bijective, and $g : \mathbb{F}_2^m \longrightarrow \mathbb{F}_2$ be any Boolean function. Let $f : \mathbb{F}_2^m \times \mathbb{F}_2^m \longrightarrow \mathbb{F}_2$ be defined by $f(x, y) = \pi(x) \cdot y + g(x)$. Show that $f$ is bent. (Maiorana-McFarland construction.)

## 2.5 An algorithm for the Walsh transformation

Let $\varphi : \mathbb{F}_2^n \longrightarrow \mathbb{R}$ be a function. We assume that $\varphi$ is given by it's value table—that is, all the values $\varphi(x)$ are known. We want to calculate the value table of the transformed function $\hat{\varphi}$. To this end we construct an algorithm via binary recursion that strongly resembles the algorithm in section 1.4. We start from the observation: For $v \in \mathbb{F}_2^j$, $w \in \mathbb{F}_2^{n-j}$ and $0 \leq j \leq n$

$$\hat{\varphi}(v, w) = \sum_{y \in \mathbb{F}_2^j} (-1)^{v \cdot y} \left[ \sum_{z \in \mathbb{F}_2^{n-j}} (-1)^{w \cdot z} \varphi(y, z) \right].$$

We define

$$\varphi^{(j)}(y, w) := \sum_{z \in \mathbb{F}_2^{n-j}} (-1)^{w \cdot z} \varphi(y, z) \quad \text{for } y \in \mathbb{F}_2^j \text{ and } w \in \mathbb{F}_2^{n-j}$$

(**partial Walsh transformation**). Then

$$\begin{aligned} \varphi^{(0)}(w) &= \hat{\varphi}(w) \quad \text{for } w \in \mathbb{F}_2^n, \\ \varphi^{(n)}(y) &= \varphi(y) \quad \text{for } y \in \mathbb{F}_2^n, \end{aligned}$$

and we have:

**Lemma 3** *For all* $v \in \mathbb{F}_2^j$ *and* $w \in \mathbb{F}_2^{n-j}$

$$\hat{\varphi}(v,w) = \sum_{y \in \mathbb{F}_2^j} (-1)^{v \cdot y} \varphi^{(j)}(y,w).$$

This gives a recursion: For $y \in \mathbb{F}_2^{j-1}$, $\eta \in \mathbb{F}_2$, $w \in \mathbb{F}_2^{n-j}$

$$\varphi^{(j-1)}(y,\eta,w) = \sum_{\zeta \in \mathbb{F}_2} \sum_{z \in \mathbb{F}_2^{(n-j)}} (-1)^{\eta\zeta + w \cdot z} \varphi(y,\zeta,z) = \sum_{\zeta \in \mathbb{F}_2} (-1)^{\eta\zeta} \varphi^{(j)}(y,\zeta,w).$$

Therefore:

**Proposition 6 (Recursion for the partial Walsh transformation)** *For* $y \in \mathbb{F}_2^{j-1}$ *and* $w \in \mathbb{F}_2^{n-j}$

$$\begin{aligned} \varphi^{(j-1)}(y,0,w) &= \varphi^{(j)}(y,0,w) + \varphi^{(j)}(y,1,w), \\ \varphi^{(j-1)}(y,1,w) &= \varphi^{(j)}(y,0,w) - \varphi^{(j)}(y,1,w). \end{aligned}$$

In order to get an iterative procedure for the Walsh transformation from this formula, we take $i := n - j$. The initial vector $x^{(0)} = (x_u)_{u \in \mathbb{F}_2^n}$ consists of the value table $x_u = \varphi(u)$ of $\varphi$. Via the intermediate vectors $x^{(i)}$, $i = 1, \ldots, n-1$, we get the final reasult $x^{(n)}$, the value table of the Walsh transform $\hat{\varphi}$. For the step from $x^{(i)}$ to $x^{(i+1)}$ we decompose the $n$-bit index as $u\xi v$ with $n-i-1$ bits $u$, 1 bit $\xi$, and $i$ bits $v$; then by proposition 6 we have:

$$\begin{aligned} x^{(i+1)}_{u0v} &= x^{(i)}_{u0v} + x^{(i)}_{u1v} \\ x^{(i+1)}_{u1v} &= x^{(i)}_{u0v} - x^{(i)}_{u1v} \end{aligned}$$

To implement this procedure in a common programming language, as before we interpret the indices as natural numbers $k = \sum k_{n-i} 2^i$ in the integer interval $[0 \ldots 2^n - 1]$ as in table 1. Then in the above equations we have $u1v = u0v + 2^i$ and in analogy with 1.4 get the formula for the iteration:

$$x_k^{(i+1)} = \begin{cases} x_k^{(i)} + x_{k+2^i}^{(i)}, & \text{if } k_{n-i} = 0, \\ x_{k-2^i}^{(i)} - x_k^{(i)}, & \text{if } k_{n-i} = 1, \end{cases}$$

for $k = 0, \ldots, 2^n - 1$. The entire algorithm reads as follows:

**Procedure [WT]**

**Input and output parameters:** A vector $x$ of length $2^n$, $x[0], \ldots, x[2^n - 1]$.

**Local variables:** A vector $y$ of length $2^n$, $y[0], \ldots, y[2^n - 1]$. Loop counters $i = 0, \ldots, n-1$, and $k = 0, \ldots, 2^n - 1$.

**Instructions:**

For $i = 0, \ldots, n-1$:
  For $k = 0, \ldots, 2^n - 1$:
    If $(((k \gg i) \bmod 2) = 1)$ then $y[k] := x[k - 2^i] - x[k]$
      else $y[k] := x[k] + x[k + 2^i]$
  For $k = 0, \ldots, 2^n - 1$:
    $x[k] := y[k]$

Of course this procedure makes sense only with exact arithmetic, say with integer vectors. One has to take care of errors by overflow.

Note that, if $\varphi$ takes values only in a subring of $\mathbb{R}$ (say $\mathbb{Z}$ or $\mathbb{Q}$), then the entire procedure works in this subring.

The expense as function of the input size $N = 2^n$ is—as in 1.4—almost linear: $3N \cdot {}^2\log N$ (as usual for the fast Fourier transform). We need roughly $2N$ memory cells for elements of the base ring (with exact arithmetic).

The corresponding C procedure in the sources is called `wt`.

## 2.6 An algorithm for the convolution

The naive application of definition 2 requires $2^{2n}$ products of (complex or integer, depending on the context) numbers: multiply each value of $\varphi$ with each value of $\psi$. The expense is quadratic in the input size $N = 2^n$.

Using the convolution theorem we reduce the expense to $N \log N$: Let's denote the intermediate result by $g := \widehat{\varphi * \psi} = \hat{\varphi}\hat{\psi}$. Then $\hat{g} = 2^n \varphi\psi$. Therefore we may use the following algorithm:

1. a) Calculate $\hat{\varphi}$,
   b) Calculate $\hat{\psi}$,
2. Multiply $g = \hat{\varphi}\hat{\psi}$ (for each argument),
3. Transform back $\varphi * \psi = \frac{1}{2^n}\hat{g}$.

The effort essentially consists of 3 Walsh transformations, each with $3n \cdot 2^n$ elementary operations; plus additionally $2^n$ multiplications in step 2. Togehter we asymptotically need some $9N \cdot {}^2\log N$ elementary operations. For this we essentially need $3N$ memory units.

**Note.** An analoguous procedure performs the efficient multiplication of polynomials via the fast Fourier transformation.

# 3 Approximation by Linear Relations

In this section we approach hidden linearity of a Boolean map by looking for linear combinations of the output bits that linearly depend on a linear combination of the input bits, at least for some arguments.

## 3.1 Transformation of indicator functions

**Definition 1** For $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ the function $\vartheta_f : \mathbb{F}_2^n \times \mathbb{F}_2^q \longrightarrow \mathbb{R}$,

$$\vartheta_f(x, y) := \begin{cases} 1, & \text{if } y = f(x), \\ 0 & \text{else,} \end{cases}$$

is called the **indicator function** of $f$.

Let's calculate the Walsh transform of an indicator function; we'll encounter the set

$$L_f(u, v) := \{x \in \mathbb{F}_2^n \mid u \cdot x = v \cdot f(x)\},$$

where the function $v \cdot f$ coincides with the linear form corresponding to $u$. The bigger $L_f(u, v)$, the closer is the linear approximation of $f$ by $(u, v)$.

$$\begin{aligned} \hat{\vartheta}_f(u, v) &= \sum_{x \in \mathbb{F}_2^n} \sum_{y \in \mathbb{F}_2^q} \vartheta_f(x, y)(-1)^{u \cdot x + v \cdot y} \\ &= \sum_{x \in \mathbb{F}_2^n} (-1)^{u \cdot x + v \cdot f(x)} \\ &= \#L_f(u, v) - (2^n - \#L_f(u, v)). \end{aligned}$$

We have shown:

**Proposition 1** *For a Boolean map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ the Walsh transform of the indicator function is $\hat{\vartheta}_f : \mathbb{F}_2^n \times \mathbb{F}_2^q \longrightarrow \mathbb{R}$,*

$$\hat{\vartheta}_f(u, v) = \sum_{x \in \mathbb{F}_2^n} (-1)^{u \cdot x + v \cdot f(x)} = 2 \cdot \#L_f(u, v) - 2^n.$$

*In particular $-2^n \leq \hat{\vartheta}_f \leq 2^n$, and all the values of $\hat{\vartheta}_f$ are even.*

The derivation of this proposition gives as an intermediate result:

**Corollary 1** *Let $\beta : \mathbb{F}_2^q \longrightarrow \mathbb{F}_2$ the linear form corresponding to $v$. Then*

$$\hat{\vartheta}_f(u, v) = \hat{\chi}_{\beta \circ f}(u).$$

**Definition 2** For a Boolean map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ the transformed function $\hat{\vartheta_f} : \mathbb{F}_2^n \times \mathbb{F}_2^q \longrightarrow \mathbb{R}$ of the indicator function $\vartheta_f$ is called the **(Walsh) spectrum** of $f$.

Imagine the spectrum $\hat{\vartheta}_f$ of $f$ as a $2^n \times 2^q$ matrix, whose rows are indexed by $u \in \mathbb{F}_2^n$ and whose columns are indexed by $v \in \mathbb{F}_2^q$, in the canonical order. By corollary 1 the columns are just the spectra of the Boolean functions $\beta \circ f$ for all the linear forms $\beta \in \mathcal{L}_q$.

**Corollary 2 (Column sums of the spectrum)** *Let $v \in \mathbb{F}_2^q$. Then*

$$\begin{aligned} \sum_{u \in \mathbb{F}_2^n} \hat{\vartheta}_f(u,v) &= \begin{cases} 2^n, & \text{if } v \cdot f(0) = 0, \\ -2^n & \text{else,} \end{cases} \\ \sum_{u \in \mathbb{F}_2^n} \hat{\vartheta}_f(u,v)^2 &= 2^{2n}. \end{aligned}$$

*Proof.* This follows from corollary 2 of the inversion formula in 2.2 and from corollary 1 together with Parseval's equation (proposition 4 in 2.3). $\diamond$

By proposition 5 in 2.4 we furthermore conclude:

$$\max |\hat{\vartheta}_f(\bullet, v)| = \max |\hat{\chi}_{\beta \circ f}| \geq 2^{n/2} \qquad \text{for each vector } v \in \mathbb{F}_2^q,$$

where equality holds, if and only if $\beta \circ f$ is bent. Hence:

**Corollary 3** *Let $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ be a Boolean map. Then*

$$\max_{\mathbb{F}_2^n \times (\mathbb{F}_2^q - \{0\})} |\hat{\vartheta}_f| \geq 2^{n/2}.$$

*Equality holds, if and only if $\beta \circ f$ is bent for each linear form $\beta : \mathbb{F}_2^q \longrightarrow \mathbb{F}_2$, $\beta \neq 0$.*

**Definition 3** (Nyberg, Eurocrypt 91) A Boolean map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ is called **bent**, if for every linear form $\beta : \mathbb{F}_2^q \longrightarrow \mathbb{F}_2$, $\beta \neq 0$, the function $\beta \circ f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ is bent.

**Remarks**

1. Because $L_f(0,0) = \mathbb{F}_2^n$ we have $\hat{\vartheta}_f(0,0) = 2^n$. Therefore every $f$ attains the upper bound in proposition 1; only some $f$ attain the lower bound.

2. If $u \neq 0$, we have

$$\hat{\vartheta}_f(u,0) = \sum_{x \in \mathbb{F}_2^n} (-1)^{u \cdot x} = 0.$$

Therefore the first column of the spectrum, "column 0", is $(2^n, 0, \ldots, 0)^t$.

3. By the corollaries of proposition 1 a Boolean map is bent, if and only if
$$\hat{\vartheta}_f(u,v) = \pm 2^{n/2} \qquad \text{for all } u \in \mathbb{F}_2^n, v \in \mathbb{F}_2^q - \{0\},$$
i. e., if the spectrum (outside of column 0) takes the values $\pm 2^{n/2}$ only.

4. If a bent map $\mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ exists, $n$ is even by corollary 1 of proposition 5 in section 2.4.

**Exercise 1** Let $f\colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^n$ be bijective. Show that the spectrum $\hat{\vartheta}_{f^{-1}}$ is given by the transposed matrix of $\hat{\vartheta}_f$.

**Exercise 2** Compare the spectrum in the case $q = 1$ with the spectrum of a Boolean function in the sense of section 2.

**Note** Nyberg, EUROCRYPT 91, has shown: A bent map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ exists, if and only if $n$ even and $\geq 2q$. The proof is slightly outside this tutorial. (It's contained in the german version.)

Let $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ be affine, $f(x) = Ax + b$ where $A \in M_{n,q}(\mathbb{F}_2)$ and $b \in \mathbb{F}_2^q$. Then

$$L_f(u,v) = \{x \in \mathbb{F}_2^n | u^t x = v^t A x + v^t b\} = \{x \in \mathbb{F}_2^n | (u^t - v^t A)x = v^t b\}.$$

This is the kernel of the linear form $u^t - v^t A$, if $v^t b = 0$. It is a parallel hyperplane, if $v^t b = 1$. We distinguish the cases

$$\#L_f(u,v) = \begin{cases} 2^n, & \text{if } v^t A = u^t \text{ and } v^t b = 0, \\ 0, & \text{if } v^t A = u^t \text{ and } v^t b = 1, \\ 2^{n-1}, & \text{if } v^t A \neq u^t. \end{cases}$$

Hence

$$\hat{\vartheta}_f(u,v) = 2 \cdot \#L_f(u,v) - 2^n = \begin{cases} 2^n, & \text{if } v^t A = u^t \text{ and } v^t b = 0, \\ -2^n, & \text{if } v^t A = u^t \text{ and } v^t b = 1, \\ 0, & \text{if } v^t A \neq u^t. \end{cases}$$

Therefore the spectrum contains exactly one entry $\pm 2^n$ in each column (i. e. for constant $v$), and only zeroes else.

If vice versa the spectrum of $f$ looks like this, then $\beta \circ f$ is affine for all linear forms $\beta\colon \mathbb{F}_2^q \longrightarrow \mathbb{F}_2$, hence $f$ is affine. We have shown:

**Proposition 2** *The map $f\colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ is affine, if and only if each column of the spectrum $\hat{\vartheta}_f$ of $f$ has exactly one entry $\neq 0$.*

**Exercise 3** Calculate the spectrum of the "half adder" $f\colon \mathbb{F}_2^2 \longrightarrow \mathbb{F}_2^2$, given by the component ANFs $f_1 = T_1T_2$ and $f_2 = T_1 + T_2$. Do the same for the "full adder" $f : \mathbb{F}_2^3 \longrightarrow \mathbb{F}_2^2$, given by the component ANFs $f_1 = T_1T_2 + T_1T_3 + T_2T_3$ and $f_2 = T_1 + T_2 + T_3$.

**Exercise 4** How does the spectrum of a Boolean map from $\mathbb{F}_2^n$ to $\mathbb{F}_2^q$ behave under affine transformations of its domain and range?

### 3.2 Balanced maps and the preimage counter

From the last section we know the first column of the spectrum. Now let's look at the first row. We'll meet the **preimage counter**

$$\nu_f(y) := \#f^{-1}(y) = \#\{x \in \mathbb{F}_2^n \mid f(x) = y\} = \sum_{x\in\mathbb{F}_2^n} \vartheta_f(x, y),$$

We have

$$\begin{aligned}\hat{\vartheta}_f(0, v) &= \sum_{x\in\mathbb{F}_2^n}\sum_{y\in\mathbb{F}_2^q} \vartheta_f(x, y)(-1)^{v\cdot y}\\ &= \sum_{y\in\mathbb{F}_2^q} \nu_f(y)(-1)^{v\cdot y}\\ &= \hat{\nu_f}(v).\end{aligned}$$

Summing up we get

$$\sum_{v\in\mathbb{F}_2^q} \hat{\vartheta}_f(0, v) = \sum_{v\in\mathbb{F}_2^q} \hat{\nu}_f(v) = 2^q \cdot \nu_f(0)$$

by 2.2. Note that $\nu_f(0)$ is the number of zeroes of $f$. We have shown:

**Lemma 1** *Let $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ be a Boolean map. Then*

$$\begin{aligned}\hat{\vartheta}_f(0, v) &= \hat{\nu}_f(v),\\ \sum_{v\in\mathbb{F}_2^q-\{0\}} \hat{\vartheta}_f(0, v) &= 2^q \cdot \nu_f(0) - 2^n.\end{aligned}$$

**Exercise 1** Let $V(f) = \{x \in \mathbb{F}_2^n \mid f(x) = 0\}$ be the zero set of $f$. Show that

$$\sum_{v\in\mathbb{F}_2^q} \hat{\vartheta}_f(u, v) = 2^q \cdot \sum_{x\in V(f)} (-1)^{u\cdot x}$$

for each $u \in \mathbb{F}_2^n$. (Row sums of the spectrum.)

For cryptology one of the most important properties of Boolean functions is balancedness (that however has nothing to do with nonlinearity). Unbalanced maps give a nonuniform distribution of their output and facilitate statistical attacks.

**Definition 4** A map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ is called **balanced**, if all its fibers $f^{-1}(y)$ for $y \in \mathbb{F}_2^q$ have the same size.

**Remarks**

1. $f$ is balanced, if and only if the preimage counter $\nu_f$ is constant.
2. If $f$ is balanced, then $f$ is surjective, in particular $n \geq q$, and the constant value of the preimage counter is $\nu_f = 2^{n-q}$; if $n = q$, then exactly the bijective maps are balanced.
3. By remark 3 in section 2.1 and remark 2 above, $f$ is balanced, if and only if $\hat{\nu}_f(0) = 2^n$ and $\hat{\nu}_f(v) = 0$ for $v \neq 0$. By lemma 1 this happens, if and only if
$$\hat{\vartheta}_f(0, v) = \begin{cases} 2^n & \text{for } v = 0, \\ 0 & \text{else.} \end{cases}$$
In this way the balancedness is tied to the first row ("row 0") of the spectrum.
4. A Boolean *function* $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ is balanced, if it takes the values 0 und 1 each exactly $2^{n-1}$ times; in other words, if its truth table contains exactly $2^{n-1}$ zeroes, or if $d(f, 0) = 2^{n-1}$. Corollary 2 in section 2.1, applied to the linear form 0, yields that $f$ is balanced, if and only if $\hat{\chi}_f(0) = 0$.
5. Because the total number of all preimages is $2^n$, we have
$$\sum_{y \in \mathbb{F}_2^q} \nu_f(y) = 2^n.$$

**Exercise 2** Show that an affine map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ is balanced, if and only if it is surjective.

**Exercise 3** Let $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ be any Boolean function, and $\breve{f} : \mathbb{F}_2^{n+1} \longrightarrow \mathbb{F}_2$ defined by $\breve{f}(x_0, x_1, \ldots x_n) = x_0 + f(x_1, \ldots x_n)$. Show that $\breve{f}$ is balanced.

**Proposition 3** (Seberry/Zhang/Zheng, Eurocrypt 94) *A Boolean map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ is balanced, if and only if for each linear form $\beta : \mathbb{F}_2^q \longrightarrow \mathbb{F}_2$, $\beta \neq 0$, the linear form $\beta \circ f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ is balanced.*

*Proof.* If $f$ is balanced, then obviously each component function $f_1, \ldots f_q : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ is balanced. An arbitrary linear form $\beta \neq 0$ can be transformed to the first coordinate function by a linear automorphism of $\mathbb{F}_2^q$; therefore $\beta \circ f$ is balanced too.

For the opposite direction we have to show, that the preimage counter is constant, $\nu_f = 2^{n-q}$. By corollary 1 in section 3.1 we have $\hat{\vartheta}_f(0,v) = \hat{\chi}_{v\cdot f}(0) = 0$ for every $v \in \mathbb{F}_2^q - \{0\}$. Moreover $\hat{\vartheta}_f(0,0) = 2^n$. Therefore the assertion follows from remark 3. $\diamond$

We also can express the balancedness by the convolution square of the preimage counter $\nu_f$:

**Proposition 4** *Let $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ be a Boolean map. Then the following statements are equivalent:*

(i) *$f$ is balanced.*

(ii) *$\nu_f * \nu_f = 2^{2n-q}$ constant.*

(iii) *$\nu_f * \nu_f(0) = 2^{2n-q}$.*

*Proof.* "(i) $\Longrightarrow$ (ii)" is almost trivial:

$$\nu_f * \nu_f(v) = \sum_{y \in \mathbb{F}_2^q} \nu_f(y)\nu_f(v+y) = 2^q \cdot 2^{n-q} \cdot 2^{n-q} = 2^{2n-q}.$$

"(ii) $\Longrightarrow$ (iii)" is the reduction to a special case.
"(iii) $\Longrightarrow$ (i)": We have

$$\begin{aligned} 2^{2n-q} = \nu_f * \nu_f(0) &= \sum_{y \in \mathbb{F}_2^q} \nu_f(y)^2, \\ 2^n &= \sum_{y \in \mathbb{F}_2^q} \nu_f(y). \end{aligned}$$

The Cauchy-Schwarz inequality yields

$$2^{2n} = \left[ \sum_{y \in \mathbb{F}_2^q} 1 \cdot \nu_f(y) \right]^2 \leq \sum_{y \in \mathbb{F}_2^q} 1^2 \cdot \sum_{y \in \mathbb{F}_2^q} \nu_f(y)^2 = 2^q \cdot 2^{2n-q}.$$

Since we have equality, $\nu_f(y)$ is a constant multiple of 1. $\diamond$

### 3.3 The linear profile

Let $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ be a Boolean map. In section 3.1 we introduced the sets $L_f(u,v)$ for $u \in \mathbb{F}_2^n$ and $v \in \mathbb{F}_2^q$. By proposition 1 we have

$$\#L_f(u,v) = 2^n - d(\alpha, \beta \circ f) = 2^{n-1} + \frac{1}{2}\,\hat{\vartheta}_f(u,v),$$

if $\alpha$ and $\beta$ are the linear forms corresponding to $u$ and $v$. We use the notation:

$$\begin{aligned} p_f(u,v) &:= \frac{\#L_f(u,v)}{2^n} = 1 - \frac{d(\alpha, \beta \circ f)}{2^n} = \frac{1}{2} + \frac{\hat{\vartheta}_f(u,v)}{2^{n+1}}, \\ \lambda_f(u,v) &:= (2p_f(u,v)-1)^2 = \frac{1}{2^{2n}} \cdot \hat{\vartheta}_f(u,v)^2. \end{aligned}$$

**Definition 5** The function

$$\lambda_f : \mathbb{F}_2^n \times \mathbb{F}_2^q \longrightarrow \mathbb{R}$$

is called the **linear profile** of $f$. The quantities $p_f(u,v)$ and $\lambda_f(u,v)$ are called the **probability** and the **potential** of the linear relation $(u,v) \in \mathbb{F}_2^n \times \mathbb{F}_2^q$ for $f$.

**Note** The use of the square in the definition of the linear profile follows a proposal of Matsui 1999. Unusual, but, as we shall see, useful too, is the normalization by the coefficient $\frac{1}{2^{2n}}$.

**Remarks**

1. We have
$$0 \le \lambda_f(u,v) \le 1,$$
$$p_f(u,v) = \frac{1 \pm \sqrt{\lambda_f(u,v)}}{2},$$
and by proposition 1 in 3.1 all values of $\lambda_f$ are integer multiples of $\frac{1}{2^{2n-2}}$.

2. Several properties of the linear profile immediately follow from the corresponding statements for the spectrum. The column 0 of the linear profile is
$$\lambda_f(u,0) = \begin{cases} 1, & \text{if } u = 0, \\ 0 & \text{else.} \end{cases}$$
All column sums of the linear profile are 1:
$$\sum_{u \in \mathbb{F}_2^n} \lambda_f(u,v) = 1.$$
In particular for each $v \in \mathbb{F}_2^q$ there is a $u \in \mathbb{F}_2^n$ such that $\lambda_f(u,v) \ge \frac{1}{2^n}$. Furthermore $f$ is balanced, if and only if row 0 of the linear profile is $10\ldots0$, and $f$ is bent, if and only if all columns except column 0 are constant $= \frac{1}{2^n}$.

**Exercise 1** Write down the linear profile for all the maps where you formerly determined the spectrum.

The quantity

$$\Lambda_f := \max\{\lambda_f(u,v) \mid u \in \mathbb{F}_2^n, v \in \mathbb{F}_2^q, (u,v) \neq 0\}$$

denotes the maximal potential of a non trivial linear relation. The bigger $\Lambda_f$, the "closer" to linearity is $f$. Linear cryptanalysis uses $\Lambda_f$ as its measure of linearity.

**Definition 6** For a Boolean map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ the quantity $\Lambda_f$ is called the **linear potential** of $f$.

**Remarks**

1. Always $0 \leq \Lambda_f \leq 1$. If $f$ is affine, then $\Lambda_f = 1$.
2. We have
$$\Lambda_f = \frac{1}{2^{2n}} \cdot \max_{(\mathbb{F}_2^n \times \mathbb{F}_2^q) - \{(0,0)\}} \hat{\vartheta}_f^2.$$
3. In the case $q = 1$ we have
$$\Lambda_f = \frac{1}{2^{2n}} \cdot \max \hat{\chi}_f^2.$$
4. More generally for $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$
$$\Lambda_f = \frac{1}{2^{2n}} \cdot \max_{\beta \in \mathcal{L}_q - \{0\}} \hat{\chi}_{\beta \circ f}^2 = \max_{\beta \in \mathcal{L}_q - \{0\}} \Lambda_{\beta \circ f}.$$

**Exercise 2** Show that $\Lambda_f$ is invariant under affine transformations of the range and domain of $f$.

**Exercise 3** Show that $\Lambda_f = \Lambda_{f^{-1}}$ if $f$ is bijective.

From corollary 2 of proposition 1 we have:

**Proposition 5** (Chabaud/Vaudenay, Eurocrypt 94) *Let $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ be a Boolean map. Then*

$$\Lambda_f \geq \frac{1}{2^n};$$

*equality holds, if and only if $f$ is bent.*

### 3.4 The nonlinearity of Boolean maps

**Definition 7** (i) (Pieprzyk/Finkelstein 1988) The **nonlinearity** of a Boolean function $f \in \mathcal{F}_n$ is the Hamming distance

$$\sigma_f := d(f, \mathcal{A}_n)$$

between $f$ and the subspace of affine functions.

(ii) (Nyberg 1992) For a Boolean map $f : \mathbb{F}_2^n \longrightarrow F_2^q$ the **nonlinearity** is

$$\sigma_f := \min\{\sigma_{\beta\circ f} \mid \beta : \mathbb{F}_2^q \longrightarrow \mathbb{F}_2 \text{ affine}, \beta \neq 0\}.$$

**Lemma 2** *The nonlinearity of a Boolean function $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ is*

$$\sigma_f = 2^{n-1} - \frac{1}{2}\max|\hat{\chi}_f|.$$

*Proof.* Let $\alpha$ be the linear form, $\bar{\alpha}$ the nonlinear affine function corresponding to $u \in \mathbb{F}_2^n$. Then by corollary 2 in 2.1

$$\begin{aligned} d(f,\alpha) &= 2^{n-1} - \frac{1}{2}\hat{\chi}_f(u), \\ d(f,\bar{\alpha}) &= 1 - d(f,\alpha) = 2^{n-1} + \frac{1}{2}\hat{\chi}_f(u), \\ d(f,\{\alpha,\bar{\alpha}\}) &= 2^{n-1} - \frac{1}{2}|\hat{\chi}_f(u)|. \end{aligned}$$

The assertion follows. $\diamond$

**Proposition 6** *The nonlinearity of a Boolean map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ is*

$$\sigma_f = 2^{n-1} - \frac{1}{2}\max|\hat{\vartheta}_f|,$$

*where the maximum is taken over $\mathbb{F}_2^n \times \mathbb{F}_2^q - \{(0,0)\}$.*

*Proof.* This follows from lemma 2 and the corollary 1 in 3.1. (The points $(u,0)$ don't affect the maximum). $\diamond$

Since the linear profile is $\lambda_f = \frac{1}{2^{2n}}\hat{\vartheta}_f^2$, for the linear potential we conclude:

**Corollary 1** (i) $\sigma_f = 2^{n-1}\cdot(1-\sqrt{\Lambda_f}), \quad \Lambda_f = \left(1 - \frac{1}{2^{n-1}}\sigma_f\right)^2$.

(ii) (Meier/Staffelbach, EUROCRYPT 89, for $q = 1$)

$$\sigma_f \leq 2^{n-1} - 2^{\frac{n}{2}-1},$$

*where the equality holds, if and only if $f$ is bent.*

In particular the nonlinearity and the linear potential are equivalent measures.

Since $\sigma_f$ is integer valued, we get better bounds for small $n$:

| $n$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| $\sigma_f \leq$ | 0 | 1 | 2 | 6 | 13 | 28 | 58 | 120 | 244 |

For $n = 3$ this gives the improved lower bound $\Lambda_f \geq \frac{1}{4}$. For $n = 5, 7, \ldots$ the analogously improved bounds $\frac{9}{256}, \frac{9}{1024}, \ldots$ become more and more uninteresting.

Because $\chi_f(u) = \pm 2^{n/2}$ for a bent function, from corollary 2 in 2.1 follows:

**Corollary 2** *If $f$ is a bent function, and $\alpha$ affine, then*

$$d(f, \alpha) = 2^{n-1} \pm 2^{\frac{n}{2}-1}.$$

**Corollary 3** *If $f$ is a bent function, then $f$ has exactly $2^{n-1} \pm 2^{\frac{n}{2}-1}$ zeroes; in particular $f$ is not balanced.*

*Proof.* $d(f, 0) = 2^{n-1} \pm 2^{\frac{n}{2}-1} \neq 2^{n-1}$. $\diamond$

**Exercise 1** Assuming the existence of a bent function, show that if $n$ is even, then there exists a balanced function $f \in \mathcal{F}_n$ whose nonlinearity is $\sigma_f = 2^{n-1} - 2^{\frac{n}{2}}$, and whose linear potential is $\Lambda_f = \frac{1}{2^{n-2}}$.

**Exercise 2** Let $f \in \mathcal{F}_n$, and let $\breve{f}$ as in exercise 3 of section 3.2. Express the linear profile, the linear potential, and the nonlinearity of $\breve{f}$ in terms of the corresponding quantities of $f$. Assuming the existence of a bent function for even $n$, show that for odd $n$ there exists a balanced function $f$ with $\sigma_f = 2^{n-1} - 2^{\frac{n-1}{2}}$, $\Lambda_f = \frac{1}{2^{n-1}}$.

# 4 Approximation by Linear Structures

The second main approach to hidden linearity is via linear structures. These are detected by difference calculus.

## 4.1 Linear structures of a Boolean map

**Definition 1** Let $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ be a Boolean map, and $u \in \mathbb{F}_2^n$. Then the **difference map** is defined by $\Delta_u f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ is

$$\Delta_u f(x) := f(x+u) - f(x) \qquad \text{for all } x \in \mathbb{F}_2^n.$$

**Lemma 1** *Let $f, g \colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ and $u \in \mathbb{F}_2^n$. Then:*

(i) $\Delta_u(f+g) = \Delta_u f + \Delta_u g$,

(ii) $\operatorname{Deg} \Delta_u f \leq \operatorname{Deg} f - 1$.

*Proof.* (i) is trivial.

(ii) Assume without loss of generality: $q = 1$, $f = T^I$ is a monomial, and finally $f = T_1 \cdots T_r$. Then

$$\Delta_u f(x) = (x_1 + u_1) \cdots (x_r + u_r) - x_1 \cdots x_r$$

obviously hs degree $\leq r - 1$. $\diamond$

**Corollary 1** *If $f$ is constant, then $\Delta_u f = 0$ for all $u \in \mathbb{F}_2^n$.*

**Corollary 2** *If $f$ is affine, then $\Delta_u f$ constant for all $u \in \mathbb{F}_2^n$.*

**Definition 2** (Evertse, EUROCRYPT 87) A vector $u \in \mathbb{F}_2^n$ is called **linear structure** of $f \colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$, if $\Delta_u f$ is constant.

**Remarks**

1. $\Delta_{u+v} f(x) = f(x+u+v) - f(x) = f(x+u+v) - f(x+v) + f(x+v) - f(x) = \Delta_u f(x+v) + \Delta_v f(x)$.
2. If $f$ is affine, then every vector is a linear structure of $f$.
3. 0 always is a linear structure of $f$.
4. If $u$ and $v$ are linear structures, then so is $u+v$ by remark 1. Therefore the linear structures of $f$ form a vector subspace of $\mathbb{F}_2^n$. On this subspace $f$ is affine. We conclude that the converse of remark 2 is also true.

5. If $g : \mathbb{F}_2^q \longrightarrow \mathbb{F}_2^r$ is linear, then $\Delta_u(g \circ f) = g \circ \Delta_u f$.

**Definition 3** For a Boolean map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ the vector space of its linear structures is called the **radical** $\mathrm{Rad}_f$, its dimension, **linearity dimension** of $f$, and its codimension, **rank** of $f$, $\mathrm{Rank}\, f$.

## 4.2 The differential profile

For a Boolean map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ and $u \in \mathbb{F}_2^n$, $v \in \mathbb{F}_2^q$ let

$$\begin{aligned} D_f(u,v) &:= \{x \in \mathbb{F}_2^n \mid \Delta_u f(x) = v\}, \\ \delta_f(u,v) &:= \frac{1}{2^n} \# D_f(u,v). \end{aligned}$$

**Definition 4** (Chabaud/Vaudenay, EUROCRYPT 94) The function

$$\delta_f : \mathbb{F}_2^n \times \mathbb{F}_2^q \longrightarrow \mathbb{R}$$

is called the **differential profile** of $f$.

(The normalization with the coefficient $\frac{1}{2^n}$ is useful. In the literature the matrix $\# D_f(u,v)$ is called difference table.)

**Remarks**

1. If $f$ is affine, $f(x) = Ax + b$, then $\Delta_u f(x) = Au$, hence

$$\begin{aligned} D_f(u,v) &= \{x \in \mathbb{F}_2^n | Au = v\} = \begin{cases} \mathbb{F}_2^n, & \text{if } Au = v, \\ \emptyset & \text{else,} \end{cases} \\ \delta_f(u,v) &= \begin{cases} 1, & \text{if } Au = v, \\ 0 & \text{else.} \end{cases} \end{aligned}$$

   Each row of the differential profile contains exactly one 1, and 0 else.

2. The following statements are equivalent:

$$\begin{aligned} u \text{ is a linear structure of } f &\iff D_f(u,v) = \begin{cases} \mathbb{F}_2^n & \text{for one } v, \\ \emptyset & \text{else} \end{cases} \\ &\iff \delta_f(u,v) = \begin{cases} 1 & \text{for one } v, \\ 0 & \text{else.} \end{cases} \end{aligned}$$

   The "row $u$" of the differential profile is 0 except exactly one entry 1.

3. For arbitrary $f$, and $u = 0$, we have

$$\delta_f(0,v) = \begin{cases} 1, & \text{if } v = 0, \\ 0 & \text{else} \end{cases}$$

   (row 0 of the differential profile).

4. $\sum_{v \in \mathbb{F}_2^q} \delta_f(u,v) = 1$ (row sums of the differential profile). In particular for each vector $u \in \mathbb{F}_2^n$ there is a $v \in \mathbb{F}_2^q$ such that $\delta_f(u,v) \geq \frac{1}{2^q}$.

We have shown:

**Proposition 1** *For a Boolean map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ the following statements are equivalent:*

(i) *$f$ is affine.*

(ii) *Each vector $u \in \mathbb{F}_2^n$ is linear structure of $f$.*

(iii) *Each row of the differential profile contains exactly one entry $\neq 0$.*

**Remarks**

5. $x \in D_f(u,v) \Leftrightarrow x + u \in D_f(u,v)$.

6. All values $\#D_f(u,v)$ are even: For $u = 0$ this follows from remark 3, else from remark 5. Therefore all $\delta_f(u,v)$ are integer multiples of $\frac{1}{2^{n-1}}$.

7. In the case $q = 1$ the autocorrelation, by its definition, can be expressed as
$$\kappa_f(x) = \delta_f(x,0) - \delta_f(x,1).$$

**Exercise 1** How does the differential profile behave under affine transformations of the argument or image space?

**Exercise 2** Show that for bijective $f$ always $\delta_{f^{-1}}(v,u) = \delta_f(u,v)$.

### 4.3 Efficient calculation of the differential profile

The following lemma is the basis for the efficient calculation of differential profiles:

**Lemma 2** *For every Boolean map $f \colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$*
$$\delta_f = \frac{1}{2^n} \vartheta_f * \vartheta_f.$$

*Proof.*
$$\begin{aligned} \vartheta_f * \vartheta_f(u,v) &= \sum_{x \in \mathbb{F}_2^n} \sum_{y \in \mathbb{F}_2^q} \vartheta_f(x,y)\vartheta_f(x+u,y+v) \\ &= \sum_{x \in \mathbb{F}_2^n} \vartheta_f(x+u, f(x)+v) \\ &= \#\{x \in \mathbb{F}_2^n \mid f(x+u) = f(x) + v\}. \diamond \end{aligned}$$

The convolution theorem yields

$$\hat{\delta}_f = \frac{1}{2^n}\hat{\vartheta}_f^2 = 2^n \lambda_f,$$

and we have shown:

**Theorem 1** *The differential profile is, up to a constant factor, the Walsh transform of the linear profile:*

$$\lambda_f = \frac{1}{2^n}\hat{\delta}_f, \qquad \delta_f = \frac{1}{2^q}\hat{\lambda}_f.$$

Parseval's equation immediately gives:

**Corollary 1** *For every Boolean map* $f\colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$

$$2^n \cdot \sum_{u\in\mathbb{F}_2^n}\sum_{v\in\mathbb{F}_2^q} \lambda_f(u,v)^2 = 2^q \cdot \sum_{x\in\mathbb{F}_2^n}\sum_{y\in\mathbb{F}_2^q} \delta_f(x,y)^2.$$

**Corollary 2** *Two Boolean maps* $\mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ *have the same linear profile, if and only if they have the same differential profile.*

Therefore we can efficiently calculate the differential profile of a map $f\colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ by the following algorithm, that yields the linear profile as an intermediate result:

1. Calculate the spectrum $\hat{\vartheta}_f$.
2. Take the squares $\omega := \hat{\vartheta}_f^2$ and normalize $\lambda_f = \frac{1}{2^{2n}} \cdot \omega$.
3. Transform back to $\delta_f = \frac{1}{2^q}\hat{\lambda}_f = \frac{1}{2^{2n+q}}\hat{\omega}$.

The effort, after having calculated $\hat{\lambda}_f$, consists of additional $3N \cdot {}^2\log(N)$ "elementary operations". All in all this makes $6N \cdot {}^2\log(N)$ such operations plus $N$ squarings, where $N = 2^{n+q}$ is the input size.

This entire procedure is in the sources as executable program `bma` ('Boolean Map Analysis').

**Exercise** Let $f\colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ be a Boolean map. Show that

$$\sum_{u\in\mathbb{F}_2^n} \delta_f(u,v) = \frac{1}{2^n}\nu_f * \nu_f(v)$$

for all $v \in \mathbb{F}_2^q$. (Remember that $\nu_f$ is the preimage counter.)

Deduce that the following statements are equivalent (Zhang/Zheng, SAC '96):

(i) $f$ is balanced.

(ii) $\sum_{u \in \mathbb{F}_2^n} \delta_f(u,v) = 2^{n-q}$ for all $v \in \mathbb{F}_2^q$ (all column sums of the differential profile).

(iii) $\sum_{u \in \mathbb{F}_2^n} \delta_f(u,0) = 2^{n-q}$ (first column sum of the differential profile).

## 4.4 The differential potential

**Definition 5** (Nyberg, EUROCRYPT 93) For a Boolean map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ the quantity

$$\Omega_f := \max\{\delta_f(u,v) \mid u \in \mathbb{F}_2^n, v \in \mathbb{F}_2^q, (u,v) \neq 0\}$$

is called **differential potential** of $f$.

**Note**: Nyberg denotes the maximum entry of the *difference table* (except at $(0,0)$) by "differential uniformity". Here I prefer a uniform treatment of the linear and the differential profiles and potentials.

**Remarks**

1. By remark 4 in 4.2 we have the bounds

$$\frac{1}{2^q} \leq \Omega_f \leq 1.$$

2. $\Omega_f$ takes the lower bound $2^{-q}$, if and only if all $\delta_f(u,v) = 2^{-q}$ for $u \neq 0$ , i. e., if all the difference maps $\Delta_u f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ are balanced. (The "row $u$" of the differential profile is constant.)

3. Since for $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ all values of the differential profile $\delta_f$ are multiples of $\frac{1}{2^{n-1}}$, the differential potential $\Omega_f \geq \frac{1}{2^{n-1}}$.

4. If $f$ has a linear structure $\neq 0$, i. e., if $\mathrm{Rad}_f \neq 0$, then $\Omega_f = 1$.

**Exercise 1** Show that $\Omega_f$ is invariant under affine transformations of $\mathbb{F}_2^n$ and $\mathbb{F}_2^q$.

**Exercise 2** Show that if $f$ is bijective, then $\Omega_{f^{-1}} = \Omega_f$.

**Definition 6** (Nyberg, EUROCRYPT 93) A Boolean map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ is called **perfectly nonlinear**, if its differential potential has the (minimally possible) value $\Omega_f = 2^{-q}$.

**Remarks**

5. By remark 5 in 4.1 and proposition 3 in 3.2 this holds, if and only if $\beta \circ f$ is perfectly nonlinear for each linear form $\beta : \mathbb{F}_2^q \longrightarrow \mathbb{F}_2$, $\beta \neq 0$.

6. A perfectly nonlinear map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ cannot have any linear structure $u \neq 0$.

7. If a perfectly nonlinear map exists, then $q \leq n - 1$ by remark 3.

From remark 2 we conclude:

**Proposition 2** *$f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ is perfectly nonlinear, if and only if the differential profile $\delta_f$ is constant $= 2^{-q}$ on $(\mathbb{F}_2^n - \{0\}) \times \mathbb{F}_2^q$.*

## 4.5 Good diffusion

**Definition 7** A Boolean map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ has **good diffusion** with respect to $u \in \mathbb{F}_2^n$, if the difference function $\Delta_u f$ is balanced.

**Remarks**

1. For $q = 1$ this means $f(x + u) - f(x) = 0$ or 1 each for exactly $2^{n-1}$ vectors $x \in \mathbb{F}_2^n$. Let's denote the number of zeroes of the difference function by

$$\eta_f(u) := \#\{x \in \mathbb{F}_2^n \mid \Delta_u f(x) = 0\} = 2^n \delta_f(u, 0),$$

   then good diffusion with respect to $u$ is equivalent with $\eta_f(u) = 2^{n-1}$.

2. For general $q$ good diffusion means, that $\#D_f(u, v) = 2^{n-q}$ and $\delta_f(u, v) = \frac{1}{2^q}$ for all $v \in \mathbb{F}_2^q$—i. e. the "row $u$" of the differential profile is constant.

3. With respect to 0 no map has good diffusion.

4. Affine maps don't have good diffusion with respect to any vector $u$.

5. A Boolean map $f$ is perfectly nonlinear, if and only if it has good diffusion with respect to *all* vectors $u \in \mathbb{F}_2^n - \{0\}$.

**Definition 8** (Webster/Tavares, CRYPTO 85) A Boolean *function* $f$ fulfils the strict avalanche criterion (SAC), if $f$ has good diffusion with respect to all canonical base vectors.

This means: Flipping one input bit changes exactly half of the values of $f$.

**Remarks**

6. Every perfectly nonlinear function fulfils the SAC.

We can express good diffusion of a Boolean function $f$ by the convolution of the character form $\chi_f$ with itself:

$$\chi_f * \chi_f(u) = 2^n \kappa_f(u) = 2^n[\delta_f(u,0) - \delta_f(u,1)] = 2\eta_f(u) - 2^n,$$

where $\kappa_f$ is the autocorrelation. Hence:

**Lemma 3** *A Boolean function $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ has good diffusion with respect to $u$, if and only if*

$$\chi_f * \chi_f(u) = 0 \qquad \textit{or in other words} \quad \kappa_f(u) = 0.$$

*Moreover $u$ is a linear structure of $f$, if and only if*

$$\chi_f * \chi_f(u) = \pm 2^n \qquad \textit{or in other words} \quad \kappa_f(u) = \pm 1.$$

Setting $u = 0$ we conclude

$$\chi_f * \chi_f(0) = 2^n,$$

since $\eta_f(0) = 2^n$. Therefore $f$ is perfectly nonlinear, if and only if $\chi_f * \chi_f = \hat{1}$, the point mass in 0, or if $(\hat{\chi}_f)^2 = \widehat{\chi_f * \chi_f} = 2^n$ constant. This was just the definition of a bent function. Thus we have shown:

**Corollary 1 (Dillon 1974)** *A Boolean function $f$ is perfectly nonlinear, if and only if it is bent.*

**Corollary 2 (Nyberg,** EUROCRYPT **91)** *A Boolean map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ is perfectly nonlinear, if and only if it is bent.*

*Proof.* Each of these properties is equivalent analogous statement for all functions $\beta \circ f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ where $\beta : \mathbb{F}_2^q \longrightarrow \mathbb{F}_2$ an arbitrary linear form $\neq 0$. $\diamond$

An expression for a globally "as good as possible" diffusion of a Boolean function is the **global autocorrelation**

$$\tau_f := \sum_{x \in \mathbb{F}_2^n} \kappa_f(x)^2 = \frac{1}{2^n} \sum_{u \in \mathbb{F}_2^n} \hat{\kappa}_f(u)^2 = \frac{1}{2^n} \sum_{u \in \mathbb{F}_2^n} \hat{\chi}_f(u)^4;$$

we have used Parseval's equation and the corollary 5 of the convolution theorem in 2.3. In particular $\tau_f \geq \kappa_f(0)^2 = 1$, and we know already, that $f$ is perfectly nonlinear, if and only if $\tau_f = 1$. Furthermore

$$\tau_f = \frac{1}{2^n} \sum_{u \in \mathbb{F}_2^n} \hat{\chi}_f(u)^4 \quad \leq \quad \frac{1}{2^n} \left[ \sum_{u \in \mathbb{F}_2^n} \hat{\chi}_f(u)^2 \right]^2,$$

because all summands are $\geq 0$; equality holds, if and only if at most one summand is $> 0$. Therefore $\tau_f \leq 2^n$, and equality holds, if and only if at most one $\hat{\chi}_f(u)^2 > 0$. This one term then must equal the total sum of squares $2^{2n}$, hence $\hat{\chi}_f(u) = \pm 2^n$, hence $L_f(u) = \emptyset$ or $\mathbb{F}_2^n$, hence $f(x) = u \cdot x + 1$ or $f(x) = u \cdot x$ for all $x$. We have shown:

**Proposition 3** *Let $\tau_f$ be the global autocorrelation of a Boolean function $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$. Then:*

(i) $1 \leq \tau_f \leq 2^n$.

(ii) $\tau_f = 1 \Longleftrightarrow f$ *perfectly nonlinear.*

(iii) $\tau_f = 2^n \Longleftrightarrow f$ *affine.*

## 4.6 The linearity distance

Let

$$\mathcal{LS}_n := \{f : \mathbb{F}_2^n \to \mathbb{F}_2 \mid f \text{ has a linear structure } \neq 0\}.$$

This is the union of the vector subspaces for a fixed linear structure, but it is in general not a vector subspace.

**Definition 9** (Meier/Staffelbach, EUROCRYPT 89) For a Boolean function $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ the Hamming distance

$$\rho_f := d(f, \mathcal{LS}_n)$$

is called the **linearity distance** of $f$.

**Remarks**

1. $\rho_f = 0 \Leftrightarrow f$ has a linear structure $\neq 0$.
2. Because $\mathcal{A}_n \subseteq \mathcal{LS}_n$, we have $\rho_f \leq \sigma_f$, the nonlinearity.

How large is $\rho_f$ else? To find an answer, we count: For a fixed vector $u \in \mathbb{F}_2^n$ we decompose $\mathbb{F}_2^n$ into two subsets

$$\begin{aligned} D_f(u,0) &= \{x \in \mathbb{F}_2^n \mid \Delta_u f(x) = 0\}, \\ D_f(u,1) &= \{x \in \mathbb{F}_2^n \mid \Delta_u f(x) = 1\} \end{aligned}$$

of sizes $n_0 = \eta_f(u) = 2^n \delta_f(u,0)$ and $n_1 = 2^n - \eta_f(u) = 2^n \delta_f(u,1)$.

First assume $n_0 \geq n_1$. To convert $f$ to a function that has $u$ as a linear structure, we have to change at least $\frac{n_1}{2}$ values, and that suffices: To see this

let $D_f(u,1) = M_1' \cup M_1''$ be decomposed into any two subsets of the same size, where $x \in M_1' \Leftrightarrow x+u \in M_1''$, $\#M_1' = \#M_1'' = \frac{n_1}{2}$; then the function

$$f'(x) := \begin{cases} f(x)+1 & \text{for } x \in M_1', \\ f(x) & \text{else,} \end{cases}$$

has $u$ as a linear structure:

$$\Delta_u f'(x) = f'(x+u) + f'(x) = \begin{cases} f(x+u)+f(x) & = 0 \quad \text{for } x \in M_0, \\ f(x+u)+f(x)+1 & = 0 \quad \text{for } x \in M_1', \\ f(x+u)+1+f(x) & = 0 \quad \text{for } x \in M_1'', \end{cases}$$

and this cannot be got with less changes.

If $n_0 < n_1$, in the same way we need $\frac{n_0}{2}$ changes. Therefore the distance of $f$ to any function $g$, that has $u$ as a linear structure, is

$$d(f,g) \geq n_f(u) := \min\{\frac{n_0}{2}, \frac{n_1}{2}\} = 2^{n-1} \cdot \min\{\delta_f(u,0), \delta_f(u,1)\},$$

and this value is assumed by a suitable $g$. We conclude

$$\rho_f = \min\{n_f(u) \mid u \in \mathbb{F}_2^n - \{0\}\}.$$

Since always $n_0 + n_1 = 2^n$, we have $n_f(u) \leq 2^{n-2}$. We have shown the first statement of:

**Proposition 4 (Meier/Staffelbach,** EUROCRYPT 89**)** *The linearity distance of a Boolean function $f\colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ is*

$$\rho_f \leq 2^{n-2}.$$

*Equality holds, if and only if $f$ is perfectly nonlinear.*

*Proof.* We have to show the second statement: In the count above for each vector $u \in \mathbb{F}_2^n - \{0\}$ we have $n_0 = \delta_f(u,0) = n_1 = \delta_f(u,1) = 2^{n-1}$. $\diamond$

Furthermore

$$\rho_f = 2^{n-1} \cdot \min\{\delta_f(u,v) \mid u \in \mathbb{F}_2^n - \{0\}, v \in \mathbb{F}_2\}.$$

Let this minimum be taken in $(u_0, v_0)$, i. e. $\rho_f = 2^{n-1} \cdot \delta_f(u_0, v_0)$, then $\delta_f(u_0, v_0+1) = 1 - \delta_f(u_0, v_0)$ is maximum, whence $= \Omega_f$. We conclude:

**Proposition 5** *The linearity distance $\rho_f$ of a Boolean function $f$ is tied to the differential potential $\Omega_f$ by the formula:*

$$\rho_f = 2^{n-1} \cdot (1 - \Omega_f).$$

# 5 Characterization of Bent Maps

In these section we summarize the properties of bent functions and maps proven in the former sections.

**Theorem 1** *For a Boolean function $f\colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ the following statements are equivalent:*

(i) *$f$ is bent, i. e., $\hat{\chi}_f^2 = 2^n$ constant.*

(ii) *$f$ is perfectly nonlinear, i. e., the difference function $\Delta_u f$ is balanced for all $u \in \mathbb{F}_2^n - \{0\}$.*

(iii) *The linear potential of $f$ has the (smallest possible) value $\Lambda_f = 2^{-n}$.*

(iv) *The nonlinearity of $f$ has the (largest possible) value $\sigma_f = 2^{n-1} - 2^{\frac{n}{2}-1}$.*

(v) *The differential potential of $f$ has the (smallest possible) value $\Omega_f = \frac{1}{2}$.*

(vi) *The linearity distance of $f$ has the (largest possible) value $\rho_f = 2^{n-2}$.*

**Corollary 1** *If $f$ is bent, then:*

(i) *$n$ is even.*

(ii) *$f$ doesn't have any linear structures $\neq 0$.*

(iii) *$f$ has exactly $2^{n-1} \pm 2^{\frac{n}{2}-1}$ zeroes and is not balanced.*

(iv) *$f$ fulfils the strict avalanche criterion.*

**Theorem 2** *For a Boolean map $f\colon \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ the following statements are equivalent:*

(i) *$f$ is bent, i. e., for all linear forms $\beta \neq 0$ on $\mathbb{F}_2^q$ the function $\beta \circ f$ is bent.*

(ii) $\max_{(\mathbb{F}_2^n \times \mathbb{F}_2^q) - \{(0,0)\}} |\hat{\vartheta}_f| = 2^{n/2}$.

(iii) *$\hat{\vartheta}_f^2$ is constant $= 2^n$ on $\mathbb{F}_2^n \times (\mathbb{F}_2^q - \{0\})$.*

(iv) *The linear potential of $f$ has the (smallest possible) value $\Lambda_f = 2^{-n}$.*

(v) *The nonlinearity of $f$ has the (largest possible) value $\sigma_f = 2^{n-1} - 2^{\frac{n}{2}-1}$.*

(vi) *$f$ is perfectly nonlinear, i. e., the differential potential has the (smallest possible) value $\Omega_f = 2^{-q}$.*

(vii) *The differential profile $\delta_f$ is constant $= 2^{-q}$ on $(\mathbb{F}_2^n - \{0\}) \times \mathbb{F}_2^q$.*

**Corollary 2** *If $f$ is bent, then:*

(i) *$n$ is even.*

(ii) *$f$ doesn't have any linear structures $\neq 0$.*

(iii) *$f$ is not balanced.*

(iv) *Each coordinate function of $f$ fulfils the strict avalanche criterion.*

[Extension of (i) without proof: ... *and* $n \geq 2q$; see the note in section 3.1.]

**Corollary 3** *A balanced map $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^q$ is not bent, in particular its differential potential $\Omega_f > 2^{-q}$, and its linear potential $\Lambda_f > 2^{-n}$.*