A New Family of Boolean Functions with Good Cryptographic Properties

Guillermo Sosa-Gómez 1,*, Octavio Paez-Osuna 2, Omar Rojas 1 and Evaristo José Madarro-Capó 3

1 Facultad de Ciencias Económicas y Empresariales, Universidad Panamericana, Álvaro del Portillo 49, Zapopan, Jalisco 45010, Mexico; orojas@up.edu.mx
2 Ronin Institute, Montclair, NJ 07043, USA; opaezosu@gmail.com
3 Institute of Cryptography, University of Havana, Havana 10400, Cuba; ejmcapo@gmail.com

* Correspondence: gsosag@up.edu.mx; Tel.: +52-3313682200

Abstract: In 2005, Philippe Guillot presented a new construction of Boolean functions using linear codes as an extension of the Maiorana–McFarland’s (MM) construction of bent functions. In this paper, we study a new family of Boolean functions with cryptographically strong properties, such as non-linearity, propagation criterion, resiliency, and balance. The construction of cryptographically strong Boolean functions is a daunting task, and there is currently a wide range of algebraic techniques and heuristics for constructing such functions; however, these methods can be complex, computationally difficult to implement, and not always produce a sufficient variety of functions. We present in this paper a construction of Boolean functions using algebraic codes following Guillot’s work.

Keywords: resilient; boolean function; Hadamard; cryptography; non-linearity

MSC: 06E30; 94C10

1. Introduction

In today’s society, there is a dependency-based communications security for financial transactions [1], telematic services, and telephony networks. Each day, new challenges arise for the protection of sensitive information from unauthorized entities. Cryptosystems are divided into two main types: symmetric and asymmetric ciphers [2]. Symmetric cryptosystems are classified in two families: block ciphers and stream ciphers. A fundamental piece of many symmetric ciphers are Boolean functions [3] used, for example, in internal state S-boxes. For example, the well known DES block cipher uses 6 input bits and 4 output bits [4] and stream ciphers, like Mickey and Sosemanuk, in the eSTREAM final round portfolio [5].

Symmetric ciphers are classified, generally, in block cipher and stream cipher and a fundamental piece in the design of both type of ciphers are Boolean functions [3]. Block ciphers include Boolean functions as component function of internal state non-linear Vectorial function, called S-boxes [4], while stream ciphers that are feedback shift register based used them as non-linear combination, non-linear filtering, or irregular clock-control [5].

In a symmetric cipher system, one seeks cryptographically strong Boolean functions that can be of practical use and resistant to different kinds of attacks that have appeared over the years (differential attack, linear attack, correlation attack, algebraic attack, etc.).

Boolean functions appear in various scientific disciplines, including Coding Theory, Combinatorics, Complexity Theory, Cryptography, and Graphic Theory, among others. In Cryptography, the design and analysis of Boolean functions that possess specific properties has often been the focus of attention. A productive field of research for most of these cryptographic properties is the Walsh-Hadamard spectrum, one of the most widely used representations of the Boolean function.
The construction of cryptographically strong Boolean functions is a daunting task, and there is currently a wide range of algebraic techniques and heuristics for constructing such functions; however, these methods can be complex, computationally difficult to implement, and not always produce a sufficient variety of functions. We present in this paper a construction of Boolean functions using algebraic codes following Guillot’s work.

2. Preliminaries

Here, we follow Reference [6]. Let $\mathbb{F}_2^n$ be the binary vector space of dimension $n$ over the Galois field of two elements $\mathbb{F}_2$. Given two vectors $a, b \in \mathbb{F}_2^n$, we define the scalar product as

$$a \cdot b = (a_1 b_1 + \ldots + a_n b_n),$$

and the sum as

$$a \oplus b = (a_1 \oplus b_1, \ldots, a_n \oplus b_n),$$

where the product and sum $\oplus$ (also called XOR) are over $\mathbb{F}_2$.

An $n$-variable Boolean function $f$ is a mapping

$$f : \mathbb{F}_2^n \rightarrow \mathbb{F}_2.$$ 

We will denote by $\mathcal{B}_n$ the set of all Boolean functions of $n$ variables. The set $\mathcal{B}_n$ is a vector space over $\mathbb{F}_2$ with the addition $\oplus$ defined by

$$(f \oplus g)(x) = f(x) \oplus g(x),$$

for any $f, g \in \mathcal{B}_n$ and any $x \in \mathbb{F}_2^n$. The polar form $\hat{f} : \mathbb{F}_2^n \rightarrow \mathbb{R}$, or sign function, of a Boolean function $f \in \mathcal{B}_n$, is defined by

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

The truth table of a Boolean function $f$ is the vector, indexed by the elements of $\mathbb{F}_2^n$ (in lexicographical order),

$$(f(\overline{0}), f(\overline{1}), \ldots, f(2^n - 1)),$$

where $\overline{0} = (0, \ldots, 0, 0)$, $\overline{1} = (0, \ldots, 0, 1)$, $\ldots$, $2^n - 1 = (1, \ldots, 1, 1)$. The polar truth table of $f$ is the $(1, -1)$ sequence defined by

$$((-1)^{f(\overline{0})}, \ldots, (-1)^{f(2^n - 1)}).$$

The support $f$, denoted by $\text{Supp}(f)$, is the set of vectors in $\mathbb{F}_2^n$ in which the image under $f$ is 1, i.e.,

$$\text{Supp}(f) = \{ x \in \mathbb{F}_2^n \mid f(x) = 1 \}.$$ 

The weight of a Boolean function $f \in \mathcal{B}_n$, denoted by $w(f)$, is the cardinality of its support, i.e., $w(f) = |\text{Supp}(f)|$. We will say that a function $f \in \mathcal{B}_n$ is balanced if $w(f) = 2^{n-1}$, i.e., if the truth table of $f$ contains the same number of 0 and 1. This property is desirable in a Boolean function to resist differential attacks, such as those introduced by Shamir against the DES algorithm [7]. A Boolean function $f \in \mathcal{B}_n$ is called affine if we can write it as

$$f(x) = (a, x) \oplus b,$$

for some $a \in \mathbb{F}_2^n$ and $b \in \mathbb{F}_2$. If $b = 0$, we say that $f$ is a linear function. The set of affine functions will be denoted by $\mathcal{A}_n$. Let $f, g \in \mathcal{B}_n$. The distance, $d(f, g)$, between $f$ and $g$, is the weight of the function $f \oplus g$, i.e.,

$$d(f, g) = w(f \oplus g).$$
The non-linearity of a Boolean function \( f \in \mathcal{B}_n \), denoted by \( N_f \), is the minimum distance between \( f \) and the set of affine functions \( \mathcal{A}_n \), i.e.,

\[
N_f = \min \{ d(f, \varphi) \mid \varphi \in \mathcal{A}_n \}.
\]

A high non-linearity is desired to reduce the effect of linear cryptanalysis attacks [8].

A Boolean function in \( \mathbb{F}_2^n \) can be expressed uniquely as a polynomial in \( \mathbb{F}_2 \left[ x_1, \ldots, x_n \right] \) through its Algebraic Normal Form (ANF)

\[
f(x) = \sum_{a \in \mathbb{F}_2^n} c_a x_1^{a_1} \cdots x_n^{a_n},
\]

where \( c_a \in \mathbb{F}_2 \), and \( a = (a_1, \ldots, a_n) \), with \( c_a = \sum_{x \leq a} f(x) \), where \( x \leq a \) means that \( x_i \leq a_i \) for all \( 1 \leq i \leq n \), i.e., \( c_a = g(a_1, \ldots, a_n) \), and \( g \) is a function in \( \mathcal{B}_n \) called the Möbius Transform of \( f \), denoted by \( g = \mu(f) \). The Algebraic Degree of a Boolean function \( f \) is the degree of its ANF. It follows that the algebraic degree of \( f \in \mathcal{B}_n \) does not exceed \( n \), that is, is the number of variables in the highest order term with non-zero coefficient.

The Walsh-Hadamard Transform of a function \( f \) in \( \mathbb{F}_2^n \) is the mapping \( H(f) : \mathbb{F}_2^n \to \mathbb{R} \), defined by

\[
H(f)(h) = \sum_{x \in \mathbb{F}_2^n} f(x)(-1)^{h \cdot x}.
\]

Let \( f \in \mathcal{B}_n \) be a Boolean function, and let \( S \) be an arbitrary subspace of \( \mathbb{F}_2^n \) and \( S^\perp \) the dual (annihilator) of \( S \), i.e.,

\[
S^\perp = \{ x \in \mathbb{F}_2^n : x \cdot s = 0, \forall s \in S \},
\]

then

\[
\sum_{u \in S} H(f)(u) = 2^{\dim S} \sum_{u \in S^\perp} f(u).
\]

From the definition of the Walsh-Hadamard Transform, it follows that \( H(\hat{f})(u) \) equals the number of zeros minus the number of ones in the binary vector \( f \oplus l_u(l_u \in \mathcal{A}_n \) and \( l_u(v) = \sum_{i=1}^n u_i v_i \) such that

\[
H(\hat{f})(u) = 2^n - 2d(f, \sum_{i=1}^n u_i v_i),
\]

\[
d(f, \sum_{i=1}^n u_i v_i) = \frac{1}{2}(2^n - H(\hat{f})(u)),
\]

\[
d(f, 1 \oplus \sum_{i=1}^n u_i v_i) = \frac{1}{2}(2^n + H(\hat{f})(u)).
\]

We summarize these results in the following

**Theorem 1.** The non-linearity \( f \) is determined by the Walsh-Hadamard Transform of \( f \), i.e.,

\[
N_f = 2^{n-1} - \frac{1}{2} \max_{u \in \mathbb{F}_2^n} |H(\hat{f})(u)|.
\]

**Proof.** See proof in Reference [9]. \( \Box \)
In what follows, we summarize some factors which are important in the design of Boolean functions with good cryptographic properties [10]. An $n$-variable Boolean function is said to have **correlation immunity** of order $m$ if and only if $H(f)(u) = 0$, with $1 \leq w(u) \leq m$. A Boolean function with correlation immunity of order $m$ and balanced is called $m$-resilient. The fundamental relationship between the number of variables $n$, the algebraic degree $d$, and the order of correlation immunity $m$ of a Boolean function is $m + d \leq n$; see Reference [11].

The **autocorrelation function** $r_f(s)$ of a Boolean function $f$ is defined from its polar representation as

$$r_f(s) = \sum_{x \in \mathbb{F}_2^n} \tilde{f}(x)\tilde{f}(x \oplus s).$$

This value is proportional to the imbalance of all the first-order derivatives of the Boolean function. Small autocorrelation values are desirable, while Boolean functions having larger values are considered weak.

We say that a Boolean function has **propagation criteria** of order $l$, denoted by $PC(l)$ if $f(x) \oplus f(x \oplus u)$ is balanced for all $u$ with $1 \leq w(u) \leq l$.

The **Strict Avalanche Criterion (SAC)** [12], refers to the effect of changing all input bits. A Boolean function $f$ is said to satisfy SAC if $f(x) \oplus f(x \oplus u)$ is balanced for all $u$ with $w(u) = 1$.

Let $q = 2^m$, and let $\mathbb{F}_q$ be the finite field with $q$ elements. An $\mathbb{F}_q$-linear **error correcting code** $C$ of length $n$ is an $\mathbb{F}_q$-linear subspace of $\mathbb{F}_q^n$. The elements of $C$ are called words. The weight $wt(x)$ of a word $x$ in $C$ is the number of its non-zero coordinates. The minimum weight $d$ of the code $C$ is defined as the minimum of the weights among all non-zero words occurring in $C$. For $x, y \in C$, we define the Hamming distance $d(x, y)$ between $x$ and $y$ as $wt(x - y)$. The **minimum distance** of a code $C$ is defined as

$$d = \min \{d(x, y)|x, y \in C, x \neq y\}.$$ 

If $k$ is the dimension of $C$ as a vector space over $\mathbb{F}_q$, then we say that $C$ is a $[n, k, d]_q$ error correcting code. The **Singleton bound** states that the parameters of a code $C$ must satisfy

$$n + 1 \geq k + d.$$ 

A code satisfying the previous inequality with equality is called a maximum distance separable code, or simply a maximum distance separable (MDS)-code [13].

For $q \geq 2, h \geq 1$. Let $Q = q^h$. Consider two codes which we call outer code and inner code. Let $C$ be outer code with parameters $[N, K, D]_Q$, and let $I$ be inner code with parameters $[n, h, d]_q$. The concatenation method [14] constructs a code $F$ over $\mathbb{F}_q$ out of a code over $\mathbb{F}_Q$. The first step is to fix any isomorphism $\varphi: \mathbb{F}_Q \rightarrow I \subseteq \mathbb{F}_q^n$. Then,

$$F := \{(\varphi(x_1), \ldots, \varphi(x_N))|(x_1, \ldots, x_N) \in C\}.$$ 

The code $F$ has parameters $[N \cdot n, K \cdot h, D \cdot d]_q$.

### 3. Maiorana-McFarland-Guillot’s Construction

The Maiorana–McFarland (MM) construction was originally designed to obtain bent functions [15]. It has been extended to construct resilient functions [16].

For $n \in \mathbb{N}$, $n \geq 2$ and $\mathbb{F}_2^q = E \oplus F$, a decomposition into two complementary subspaces $E$ of dimension $p$ and $F$ of dimension $q = n - p$. For any application $\pi: E \rightarrow \mathbb{F}_2^q$ and any application $h: E \rightarrow \mathbb{F}_2$, the MM construction defines a Boolean function $f$ as follows:

$$f: E \oplus F \rightarrow \mathbb{F}_2$$

$$x + y \mapsto \pi(x) \cdot y + h(x).$$
The application $\pi$ is defined on $\mathbb{F}_2^n$, but, since $\pi(x)$ is wrapped by an internal product with an element of $F$, the value of $f$ is invariant when $\pi(x)$ is moved by a vector of $F^\perp$. So, $\pi$ can be considered to be defined over the space $\mathbb{F}_2^n / F^\perp \cong E^\perp$, so that $\pi : E \to E^\perp$.

One of the properties we are interested in from a Boolean function is the Propagation Criteria. In Reference [16], it was shown that for a Boolean function to have Propagation Criteria of order $k$ it is enough that the coset $x_0 + F$, with $x_0 \in E$, has $w(x_0 + F) > k$. Therefore, to find a Boolean function with $PC(k - 1)$, it is enough to select an appropriate $x_0$ in the complement of $F$, such that the lateral class $x_0 + F$ has weight $\geq k$.

4. Construction of $\pi$ and $h$

4.1. $\pi$ One-to-One

Following Guillot’s ideas, we note that we need to have all the values of $\pi(E)$. For $u \in E^\perp$, we will construct the lateral class $u + F^\perp$ and calculate the minimum weight of this class for each $u$. Let us save these weights in the set $\mathcal{U}(l) = \{u \in E^\perp : w(u + F^\perp) \geq l + 1\}$. In order to build $\pi$ one by one, we must have $q > p$ because we want our Boolean function to be balanced; this implies that $H(\hat{f})(0) = 0$, i.e., $0 \notin \pi(E)$. To build a Boolean function with a high resiliency order $(l)$, we must take care that the cardinality of $\mathcal{U}(l) \geq 2^p$. To build the image of $\pi(E)$, we take $2^p$ non-zero elements of $\mathcal{U}(l)$, and we assign it randomly. In the same way, we randomly generate the values of $h(E)$ using any pseudo-random generator.

4.2. $\pi$ Two-to-One

Let $u \in E^\perp$. The linear functional $v \mapsto v \cdot x_0$ partitions the lateral class $u + F^\perp$ into the sets

$$E_0 = \{v \in u + F^\perp \mid v \cdot x_0 = 0\},$$

and

$$E_1 = \{v \in u + F^\perp \mid v \cdot x_0 = 1\}.$$

Let $w_0$ and $w_1$ be the minimum weights of $E_0$ and $E_1$, respectively. In order to construct an $l$–resilient function, we need that

$$\max\{w_0, w_1\} \geq l + 1.$$

We will store the pairs $(u,i)$ such that $w_i = \max\{w_0, w_1\}$ in the set $\mathcal{U}$, i.e.,

$$\mathcal{U} = \{(u,i)\mid u \in E^\perp, w_i = \max\{w_0, w_1\}\}.$$

We proceed to construct the image $\pi(E)$ and the image $h(E)$. To do so, we first notice that translating by $x_0$ partitions $E$ into two parts of size $2^{p-1}$: a $p - 1$ dimensional subspace $S_0 \leq E$ not containing $x_0$ and its translate $S_1 = x_0 + S_0$. We construct the image of $\pi$ in such a way that $\pi(x) = \pi(x + x_0)$ for all $x \in E$. There are many possible ways of achieving this. For the image of $h$, we choose the value of $h(E)$ at random for $x \in S_0$, and we define $h(x + x_0) = h(x) + i$ such that $(\pi(x), i) \in \mathcal{U}$.

5. Construction of $f$

Recall that the Boolean function $f$ is expressed as

$$f(z) = \pi(x) \cdot y + h(x),$$

with $z = x + y; x \in E, y \in F$. We have all the ingredients in place to compute $f$. For example, we may proceed as follows: without loss of generality, we may assume that the information coordinates for the code $F$ are the first $q$ coordinates. Then, given $z = (z_1, z_2, \ldots, z_q, z_{q+1}, \ldots, z_n) \in \mathbb{F}_2^n$, we may compute $y$ using any systematic generator matrix $G_F$ for $F$ as
and then we obtain $x$ by computing $x = y + z$. As we know the images of $\pi$ an $h$ for all $x \in E$ (see previous section), we may obtain the value of $f(z)$.

6. Reed-Solomon Codes

The class of Reed-Solomon Codes [17] is considered of great importance in coding theory. They are members of the family of algebraic codes. Recall one of the standard descriptions of an extended Reed-Solomon code over $\mathbb{F}_q$ [18]. Let $\mathbb{F}_q = \{0, 1, a, a^2, \ldots, a^{q-2}\}$. Consider the set

$$L = \{ f(x) \in \mathbb{F}_q[x] \mid \deg(f(x)) < r \}.$$ 

The Reed-Solomon code $RS(r, q)$ of length $n = q$ is defined by

$$RS(r, q) := \{ c = (f(0), f(1), f(a), f(a^2), \ldots, f(a^{q-2})) \mid f(x) \in L \}.$$ 

Because a polynomial of degree $l$ has at most $l$ zeros in $\mathbb{F}_q$, we see that $RS(r, q)$ has minimum distance $d = q - r + 1$, which is the best possible, i.e., $RS(r, q)$ is a maximum distance separable (MDS) code [18]. The code $RS(r, q)$ has parameters

$$[q, r, q - r + 1]_q.$$ 

In this paper, we will assume that $q = 2^m$; then, $RS(r, 2^m)$ has parameters

$$[2^m, r, 2^m - r + 1]_{2^m}.$$ 

7. Boolean Functions from $RS(r, 2^m)$

For our construction of Boolean functions, we will use a concatenated Reed-Solomon code. Let $C = RS(r, 2^m)$; this is our outer code. Let $I$ be the all even-weight codewords, then with parameters $[m + 1, m, 2]_2$. After concatenation, we obtain a code $F$ with parameters

$$[(m + 1)2^m, m \cdot r, 2(2^m - r + 1)]_{2^m}.$$ 

We will use our code $F$ as the main ingredient to the MM construction, obtaining a new family of Boolean functions, in $n = (m + 1)2^m$ variables. The dimension of the complementary vector space $E$ is, therefore, $(m + 1)2^m - m \cdot r$, and $\mathbb{F}_q^E = E \oplus F$.

We focus now in the lateral class $x_0 + F$. As $F$ is constructed by evaluating all polynomials of degree less than $r$ over $\mathbb{F}_{2^{2m}}[x]$, we can assume that $x_0$ is also constructed by evaluating a polynomial $L(x)$ over $\mathbb{F}_{2^{2m}}[x]$. A polynomial $L(x)$ can be obtained using Lagrange interpolation in which the evaluation produces a suitable concatenated $x_0$. Let $a_1, \ldots, a_r$ be a set of information coordinates for the code $RS(r, 2^m)$, by Lagrange interpolation, we can obtain a polynomial $L(x)$ of degree $r$ such that $L(a_i) = 0$ for $i = 1, \ldots, r$ and $L(a_{i+1}) = s, s \in \mathbb{F}_{2^m} - \{0\}$. The vector $ev(L)$ is a vector in the complement of $RS(r, 2^m)$ as a vector space over $\mathbb{F}_{2^{2m}}$, and the lateral class $ev(L) + RS(r, 2^m)$ has minimum weight $\geq 2^m - r$. Let $x_0$ be the image of $ev(L)$ under concatenation; it follows that $x_0$ is a vector in the complement of $F$ as a binary vector space, and, by construction, the minimum weight of the lateral class $x_0 + F$ is $\geq 2^{2m - r}$. Thus, by using our proposed $F$ and $x_0$ in Guillot’s construction, we obtain a Boolean function satisfying $PC(2^{m+1} - 2r - 1)$. 

\[ y = (z_1, z_2, \ldots, z_q) \cdot G_f, \]
8. On the Number of Distinct Boolean Functions

The cryptographic properties of the functions obtained in our construction depend solely on the properties of the code $F$ and the image $\pi(E)$. Once $\pi(E)$ and $F$ are fixed, we may construct $2^p! \times 2^q!$ (see Reference [16]) distinct Boolean functions with identical cryptographic properties. In our construction, we have incorporated extra choices in the concatenation step; namely, we have $2^{2k} - 1$ choices for the isomorphism:

$$\phi : \mathbb{F}_{2^{2k}} \longrightarrow \mathbb{F}_2^{2k+1},$$

bringing the total number of distinct Boolean functions up to

$$2^p! \times 2^q! \times (2^{2k} - 1).$$

9. Examples

9.1. Example #1. $\pi$ One-to-One

For this example, we will take $n = 16$; so, the $\dim(F) = 9$ and the $\dim(E) = 7$ was the one that was selected. For this, we select a code $RS(3, 4)$ over $\mathbb{F}_3$ with primitive polynomial $a^3 + a^2 + 1$ and a concatenation code of even weight $I = \{ 4, 3, 2 \}$. Now,

$$G_F = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & a^1 & a^2 & a^1 & a^2 & 1 & + a^1 & 0 & 0 \\ 0 & a^2 & a^1 & a^2 & 1 & + a^2 & 0 & 0 & 1 \end{pmatrix}. \quad \mathbb{F}_3$$

$$G_F = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$G_F = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$G_F = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

We build all cosets of the form $u + F^\perp$, where $u \in E^\perp$, and we calculate the minimum weights of each class.

For $n = 16$, the maximum resistance we can aspire to is $l = 3$; so, since $E$ has 128 elements, we choose exactly from $E^\perp$, and we determine.

$U(3) = \{ D600, B900, 6B00, 5B00, EF00, DF00, D280, 3980, DB80, D200, B200, 7200, CA00, AA00, 6A00, 9A00, 5A00, 3A00, C600, A600, 6600, 9600, 5600, 3600, F600, 8E00, 2E00, EE00, DE00, BE00, D100, B100, 7100, A900, 6900, 9900, 5900, 3900, D800, F900, C500, A500, 6500, 9500, 5500, 3500, F500, 8D00, 2D00, ED00, DD00, BD00, C300, A300, 6300, 9300, 5300, F300, 8B00, 4B00, B800, EB00, 1B00, 7800, DB00, B800, 7B00, 4700, 2700, E700, 1700, D700, 7700, 0F00, CF00, AF00, 6F00, D400, 9F00, 5F00, 7400, 3F00, FF00, D080, B080, 7080, C880, A880, 6880, 5880, 3880, F880, 5480, 3480, F480, 2C80, EC80, DC80, BC80, C280, A280, 9280, 5280, CC00, 3280, F280, 8A80, 4A80, 2A80, EA80, 1A80, DA80, BA80, 4680, 2680, E680, 1680, D680, 0E80, CE80, AE80, C180, A180, 6180, 9180, 3180, F180 \}$ in hexadecimal but with $u \neq 0$ to ensure that the function is balanced and in turn these vectors will be the images of $\pi$. 
Now, we get the \( h(E) \) values randomly using any binary pseudo-random generator.

In summary, our function will have the following properties: balanced, \( N_f = 32,512 \), 3-resilient, and \( \deg(f) = 8 \).

9.2. Example #2. \( \pi \) Two to One

Suppose we want to build a 12-variable Boolean function. As the main ingredient, we use the Reed-Solomon code \( C = RS(3, 4) \) over \( F_4 \) with parameters \([4, 3, 2]\). A generator matrix for \( C \) is

\[
G = \begin{pmatrix}
1 & 1 & 1 & 1 \\
0 & \alpha & \alpha + 1 & 1 \\
0 & \alpha + 1 & \alpha & 1
\end{pmatrix},
\]

where \( \alpha^2 + \alpha + 1 = 0 \). We now obtain a binary code from \( C \) by concatenation with the even weight code \( I = \{000, 101, 011, 110\} \) with parameters \([3, 2, 2]\). Any other 2-dimensional binary code will serve as an inner code. The next step is to choose any homomorphism \( \nu \) between \( F_4 \) and \( I \) as vector spaces over \( F_2 \). For our example, we choose

\[
0 \mapsto 000, \quad 1 \mapsto 101, \quad \alpha \mapsto 011, \quad \alpha + 1 \mapsto 110.
\]

After concatenation, we obtain a binary code \( F \) with parameters \([12, 6, 4]\). A systematic generator matrix for \( F \) is given by

\[
G_F = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1
\end{pmatrix}.
\]

The row span of \( G_F \) is the binary vector space \( E \) in the MM construction. As \( G_F \) is systematic, i.e., the first 6 columns are the information coordinates of code \( F \), we may easily describe the complementary space \( E \) with generator matrix

\[
G_E = \begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}.
\]

In this example, we have \( n = 12, p = 6, q = 6 \), so we will build a two to one function \( \pi \). The next step is to build \( x_0 \in E \) by concatenation of the evaluation vector of \( L(x) = x^2 + x \). We obtain \( x_0 = \{0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0\} \in E \). For each lateral class \( u + F^\perp \) with \( u \in E^\perp \), we construct the sets \( E_0 = \{v \in u + F^\perp : v \cdot x_0 = 0\} \) and \( E_1 = \{v \in u + F^\perp : v \cdot x_0 = 1\} \). Let \( d_0 = d(E_0), d_1(E_1) \) be the minimum distances of \( E_0 \) and \( E_1 \), respectively, and let \( d_j = \max\{d_0, d_1\} \). Next, we store the pairs \((u, j)\) in an array. In this example, the array is given by
As might be noticed, all $u$ in the previous arrays have weight $\geq 3$, as expected from Guillot’s results; so, the Boolean function we will construct will have resilience order 2. For $x \in E$, we define $\pi(x) = \pi(x + x_0) \in \mathbb{F}_2$ at random, and we define $h(x) = h_u$ and $h(x + x_0) = h_u + h_t$, where $h_t$ is a random value in $\mathbb{F}_2$.

Using $\pi$ and $h$ defined above in the MM construction, the following cryptographic parameters for the Boolean function $f$ were checked using sage: Balanced, non-linearity of 1984, propagation criteria of order 3, and resilience of order 2.

10. Conclusions

In this work, a complete description is made of how the images of $\pi$ and $h$ should be chosen to build the well-desired Boolean functions. The desirable cryptographic properties were carefully reviewed given the chosen construction, and we described the advantages and disadvantages of the aforementioned construction. We set which are going to do the optimal parameters to find the Boolean functions. Thus, it will be possible to know when the functions will be balanced and what order of propagation criterion they will have; thus, it will be possible to know the non-linearity and resiliency order. It is also important to note that a non-trivial factor is obtained in terms of the number of different functions obtained with the same properties. All of the above can be known beforehand by the characteristics and properties of the Reed-Solomon code used, according to the function that wants to be built from $n$-variables.
Author Contributions: Conceptualization, E.J.M.C., G.S.-G. and O.P.-O.; methodology, E.J.M.-C., G.S.-G. and O.R.; software, E.J.M.-C., G.S.-G. and O.P.-O.; validation, E.J.M.-C. and O.R.; formal analysis, E.J.M.-C., G.S.-G., O.R. and O.P.-O.; investigation, E.J.M.-C., G.S.-G., O.R. and O.P.-O.; writing—original draft preparation, E.J.M.-C., G.S.-G., O.R. and O.P.-O.; writing—review and editing, G.S.-G., O.R. and O.P.-O.; supervision, O.R. and O.P.-O. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Conflicts of Interest: The authors declare no conflict of interest.

References
1. Lachowicz, P. Walsh–Hadamard Transform and Tests for Randomness of Financial Return-Series. Presented at Quant at Risk (Online), 7 April 2015. Available online: https://quantatrisk.com/2015/04/07/walsh-hadamard-transform-python-tests-for-randomness-of-financial-return-series/ (accessed on 15 March 2021).
2. Menezes, A.J.; Van Oorschot, P.C.; Vanstone, S.A. Handbook of Applied Cryptography; CRC Press: Boca Raton, FL, USA, 2018.
3. Pasalic, E. On Boolean Functions in Symmetric-Key Ciphers; Lund University: Lund, Sweden, 2003.
4. Zeebarea, S.R. DES encryption and decryption algorithm implementation based on FPGA. Indones. J. Electr. Eng. Comput. Sci. 2020, 18, 774–781. [CrossRef]
5. Jiao, L.; Hao, Y.; Feng, D. Stream cipher designs: A review. Sci. China Inf. Sci. 2020, 63, 131101. [CrossRef]
6. Cusick, T.W.; Stanica, P. Cryptographic Boolean Functions and Applications; Academic Press: Cambridge, MA, USA, 2017.
7. Biham, E.; Shamir, A. Differential cryptanalysis of DES-like cryptosystems. J. Cryptol. 1991, 4, 3–72. [CrossRef]
8. Langford, S.K.; Hellman, M.E. Differential-linear cryptanalysis. In Annual International Cryptology Conference; Springer: Berlin/Heidelberg, Germany, 1994; pp. 17–25.
9. Chuan-Kun, W.; Dengguo, F. Boolean Functions and Their Applications in Cryptography; Springer: Berlin/Heidelberg, Germany, 2016. [CrossRef]
10. Henríquez, F.R. De la búsqueda de funciones booleanas con buenas propiedades criptográficas. Cinvestav 2007, 26, 50–65.
11. Chee, S.; Lee, S.; Lee, D.; Sung, S.H. On the correlation immune functions and their nonlinearity. In International Conference on the Theory and Application of Cryptology and Information Security; AsiaCrypt; Springer: Berlin/Heidelberg, Germany, 1996; pp. 232–243. [CrossRef]
12. Forrié, R. The strict avalanche criterion: Spectral properties of Boolean functions and an extended definition. In Conference on the Theory and Application of Cryptography; Springer: Berlin/Heidelberg, Germany, 1988; pp. 450–468. [CrossRef]
13. Stichtenoth, H. A Note on Hermitian Codes Over GF(q2). IEEE Trans. Inf. Theory 1988, 34, 1345–1348. [CrossRef]
14. Forney, G.D. Concatenated Codes; Citeseer: Princeton, NJ, USA, 1966; Volume 11.
15. Carlet, C. A Larger Class of Cryptographic Boolean Functions via a Study of the Maiorana-McFarland Construction. In Advances in Cryptology; Springer: Berlin/Heidelberg, Germany, 2002; Volume 2442, pp. 549–564. [CrossRef]
16. Guillot, P. Cryptographical boolean functions construction from linear codes. Boolean Funct. Cryptogr. Appl. 2005, 387, 141.
17. Wicker, S.B.; Bhargava, V.K. Reed-Solomon Codes and Their Applications; John Wiley & Sons: Hoboken, NJ, USA, 1999.
18. Van Lint, J.H. Introduction to Coding Theory; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2012; Volume 86.