COVERING SEQUENCES AND *T*,*K*-BENTNESS CRITERIA FOR BOOLEAN FUNCTIONS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

GÜZİN KURNAZ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
ELECTRIC AND ELECTRONICS ENGINEERING

MARCH 2009

Approval of the thesis:

COVERING SEQUENCES AND *T,K*-BENTNESS CRITERIA FOR BOOLEAN FUNCTIONS

submitted by **GÜZİN KURNAZ** in partial fulfillment of the requirements for the degree of **DOCTOR of PHILOSOPHY in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen ________________
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. S. İsmet Erkmen ________________
Head of Department, **Electrical and Electronics Engineering**

Assoc. Prof. Dr. Melek Diker Yücel ________________
Supervisor, **Electrical and Electronics Engineering Dept., METU**

Examining Committee Members:

Prof. Dr. Yalçın Tanık ________________
Electrical and Electronics Engineering Dept., METU

Assoc. Prof. Dr. Ali Özgür Yılmaz ________________
Electrical and Electronics Engineering Dept., METU

Prof. Dr. Ferruh Özbudak ________________
Mathematics Dept., METU

Assist. Prof. Dr. Ali Aydın Selçuk ________________
Computer Engineering Dept., Bilkent University

Date: 05/03/2009

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name : Güzin KURNAZ

Signature :

ABSTRACT

COVERING SEQUENCES AND T, K-BENTNESS CRITERIA
FOR BOOLEAN FUNCTIONS

Kurnaz, Güzin

Ph.D., Department of Electrical and Electronics Engineering

Supervisor: Assoc. Prof. Dr. Melek Diker Yücel

March 2009, 126 pages

This dissertation deals with some crucial building blocks of cryptosystems in symmetric cryptography; namely the Boolean functions that produce a single-bit result for each possible value of the *m*-bit input vector, where m>1. Objectives in this study are two-fold; the first objective is to develop relations between cryptographic properties of Boolean functions, and the second one is to form new concepts that associate coding theory with cryptology.

For the first objective, we concentrate on the cryptographic properties of Boolean functions such as balancedness, correlation immunity, nonlinearity, resiliency and propagation characteristics; many of which are depending on the Walsh spectrum that gives components of the Boolean function along the direction of linear functions. Another efficient tool to study Boolean functions is the subject of covering sequences introduced by Carlet and Tarannikov in 2000. Covering sequences are defined in terms of the derivatives of the Boolean function. Carlet and Tarannikov relate the correlation immunity and balancedness properties of the Boolean function to its covering sequences. We find further relations between the covering sequence and the Walsh spectrum, and present two theorems for the calculation of covering sequences associated with each null frequency of the Walsh spectrum.

As for the second objective of this thesis, we have studied linear codes over the rings Z4 and Z8 and their binary images in the Galois field GF(2). We have investigated the best-known examples of nonlinear binary error-correcting codes such as Kerdock, Preperata and Nordstrom-Robinson, which are Z_4-linear codes. We have then reviewed Tokareva's studies on Z4-linear codes and extended them to Z8-linear codes. We have defined a new classes of bent functions. Next, we have shown that the newly defined classes of bent, namely Tokareva's k-bent and our t,k-bent functions are affine equivalent to the well-known Maiorana McFarland class of bent functions. As a cryptological application, we have described the method of cubic cryptanalysis, as a generalization of the linear cryptanalysis given by Matsui in 1993. We conjecture that the newly introduced *t,k*-bent functions are also strong against cubic cryptanalysis, because they are as far as possible to *t,k*-bent functions.

Keywords: Boolean functions, nonlinearity, Walsh-Hadamard transformation, covering sequence, affine equivalence, bent functions, k-bent functions.

ÖZ

BOOLE İŞLEVLERİ İÇİN

KAPSAYAN DİZİNLER VE T, K-BÜKÜKLÜK ÖLÇÜTLERİ

KURNAZ, Güzin

Doktora, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Yöneticisi: Doç. Dr. Melek Diker Yücel

Mart 2009, 126 sayfa

Bu tez, simetrik kriptografideki kripto-sistemlerinin önemli yapısal bloklarından olan ve *m*-ikiliden oluşan (m>1) her girdiye karşılık bir tek ikili çıktı üreten Boole fonksiyonlarına değinmektedir. Bu çalışmanın iki ana amacı vardır; ilk amaç Boole fonksiyonlarının kriptolojik özellikleri arasında ilişkiler geliştirmek; ikincisi ise kodlama teorisi ve kriptoloji arasında yeni bir geçiş oluşturan kavramlar üretmektir.

İlk amaç doğrultusunda, dengelilik, ilinti (korelasyon) bağışıklığı, doğrusal olmama, esneklik ve yayılma gibi Boole fonksiyonu özellikleri üzerine yoğunlaşılmıştır; ki bu özelliklerin çoğu, fonksiyonun doğrusal işlevler yönündeki

bileşenlerini veren Walsh görüngesine bağlıdır. Boole fonksiyonlarını çalışmak için etkili bir diğer yöntem ise, 2000 yılında Carlet ve Tarannikov, tarafından sunulan kapsayan dizin konusudur. Kapsayan dizinler, Boole fonksiyonlarının türevlerine bağlı olarak tanımlanmaktadır. Carlet ve Tarannikov, dengelilik ve ilinti bağışıklığının kapsayan dizinlerle ilişkilerini kurmuşlardır. Bizim çalışmalarımızda ise Walsh görüngesi ve kapsayan dizinler arasında yeni bağlantılar kurularak, Walsh görüngesinin her sıfır frekansına bağlı kapsayan dizinin hesaplanması üzerine iki teorem sunulmaktadır.

Tezin ikinci amacı için, Z_4 ve Z_8 halkalarındaki doğrusal kodlar ve bu kodların GF(2) sonlu cismine eşlenmiş görüntüleri üzerinde çalıştık. Z4-doğrusal kodlarının görüntüsü olan ve bilinen en iyi doğrusal olmayan ikili hata-düzeltme kodlarını, Kerdock, Preperata ve Nordstrom-Robinson'u inceledik. Tokareva'nın Z4-doğrusal kodlar üzerindeki çalışmalarını Z8-doğrusal kodlara genişlettik. Yeni t,k-doğrusalımsı ve *t,k*-bükük fonksiyonlar tanımlayarak, Tokareva'nın k-bükük ve bizim *t,k*-bükük fonksiyonlarımızın, yaygın olarak bilinen Maiorana McFarland sınıfı bükük fonksiyonlarla doğrusal denkliğini gösterdik. Ayrıca, kriptolojik bir uygulama olarak, 1993 yılında Matsui tarafından tanımlanan doğrusal kriptanalizi genelledik ve kübik kriptanalizi tanımladık. Önerdiğimiz *t,k*-bükük fonksiyonlar tüm birinci, ikinci ve üçüncü derece fonksiyonlardan olabildiğince uzakta olduğu için, kübik kriptanalize karşı da dirençli oldukları kanısındayız.

Anahtar Sözcükler: Boole işlevleri, doğrusal olmama, Walsh-Hadamard dönüşümü, kapsayan dizin, doğrusal denklik, bükük işlevler, k-bükük işlevler.

To my whole family; my parents, my sisters, my husband Faruk and my sons Hasan and Cafer.

ACKNOWLEDGMENTS

My first acknowledgment must go to my supervisor Assoc. Prof. Melek Diker Yücel for patiently guiding, and motivating me throughout this study. In every sense, none of this work would have been possible without her.

I would like to thank also to my family for their love and support during not only the thesis but also my whole educational life.

I would like to thank to my friends Özgül Salor, Selva Muratoğlu, Seval Özaydın and Süheyda Küçükpetek for their encouregements.

I wish to thank to TÜBİTAK-SAGE, the Scientific and Technical Research Council of Turkey, for supporting my work through a doctoral thesis.

TABLE OF CONTENTS

LIST OF TABLES

Notation

Related to fields and rings,

Z	Ring of integers
$GF(2)$	Galois Field with two elements
$GF(2)^m$	Galois Field with 2^m elements

Related to vectors

$\mathbf{x}$	m-bit row vector
x_i	i^{th} bit of the vector $\mathbf{x}$.
$wt(\mathbf{x})$	Hamming weight of the vector $\mathbf{x}$.
$d(\mathbf{x}, \mathbf{y})$	Hamming distance between vectors $\mathbf{x}$ and $\mathbf{y}$.

Related to Boolean functions

$W_f(\mathbf{w})$	Walsh transform of the function *f* at frequency $\mathbf{w}$.
r_f	Autocorrelation function of *f*.
$\mathbf{D_a f}$	Derivative vector of the function *f* for a input shift vector $\mathbf{a}$.
δ	Kronecker delta function

Related to codes

(n,k,d) code Linear code with length *n*, dimension *k* and minimum distance *d*.

$RM(r,m)$ Reed Muller code of order *r* and length 2^m.

Related to matrices

$\mathbf{A_n}$	Matrix of order *n* associated with the Möbius transform
$\mathbf{H_n}$	Hadamard matrix of order *n*.
$\mathbf{I_n}$	$n \times n$ identity matrix.
$\mathbf{J_n}$	$n \times n$ matrix of all ones.

Related to operators

$\oplus$	Addition modulo 2.
$+, \sum$	Integer addition or addition on rings depending on context.
$< .,. >$	dot or scalar product of two vectors

CHAPTER 1

INTRODUCTION

In this thesis, we focus on the study of Boolean functions, which are among the main building blocks of symmetric cryptosystems. Symmetric cryptography is used in GSM mobile phones, WLAN and Internet connections, banking transactions, credit cards and many other places as an effective means of privacy and authentication [2].

There are various and comprehensive studies in the literature for the usage of Boolean functions inside cryptography. A vector Boolean function or an S-box [61, 66, 70, 84] maps m input bits to n output bits; for $n > 1$ and $m > 1$. If $n = 1$, corresponding function is simply called an m-variable Boolean function. A Boolean function f can be uniquely represented both by its truth table, which is a vector that contains the function values of f and its Walsh transform, which is a kind of discrete Fourier transform. The most desirable Boolean function properties are those, which strengthen the related cryptosystem against well known statistical attacks such as differential, linear and algebraic cryptanalysis. We refer to [4, 10, 11, 19, 22, 42, 79] for linear and differential cryptanalysis and [3, 11, 16, 17, 34, 36] for algebraic cryptanalysis. A Boolean function must have good autocorrelation properties [37, 57, 58, 79, 82, 88, 95, 97, 102, 14, 116, 117] in order to be safe against differential cryptanalysis. Moreover, a Boolean function must be highly nonlinear, i.e., it must be as far as possible to all affine functions [49-54] to be strong against linear cryptanalysis. In other words, the magnitude Walsh spectrum of a cryptographically strong Boolean function should be as flat as possible, to yield maximum achievable nonlinearity [51-53, 77, 78, 83]. Bent functions [33, 43, 44, 49, 50, 68, 94, 118] are the Boolean functions that reach this maximum nonlinearity. They were first studied by Dillon [49] and Rothaus [94] and Rothaus used the word "bent" in the literature in 1970. Maiorana McFarland class of bent functions [41, 87, 113] are one of the main families of bent functions. This class can be constructed by concatenating affine functions and it achieves good cryptographic properties.

Correlation Immunity [6, 37, 63, 66, 73, 79, 89, 91] of a Boolean function measures the correlation of its input variables to its output value. A Boolean function is said to be correlation immune of order r if every subset of r or fewer input variables are statistically independent with the output. A Boolean function with lower order correlation immunity is more succeptible to correlation attacks [16, 17, 21, 34, 36] than a Boolean function with higher order correlation immunity. It is well known that the correlation immunity order of a Boolean function can be directly found from zeros of its Walsh transform spectrum. In 2000, Carlet and Tarannikov [40] introduced the notion of covering sequences, which are connected to the function via its derivatives as an efficient tool to study Boolean functions. Then they showed that correlation immunity order and covering sequences [39, 40, 101] of a Boolean function are related.

Classification of Boolean functions is another subject in cryptology. Affine equivalent Boolean functions [1, 18, 20, 24, 25, 48, 56, 60, 69, 100, 109] have similar cryptographic properties. This makes affine classification meaningful in the sense that the number of representatives is much less than the number of all Boolean functions. Such perspective allows the Boolean space to be considered as a structure in which all Boolean functions are grouped into affine equivalence classes and only one function from each class is sufficient for analysis.

Relations between error correcting codes and Boolean functions are studied extensively in the literature [12-16, 26-31, 45, 46, 55, 59, 64, 67, 71-76, 86, 93, 96, 98-108, 110-112]. Some of the best-known examples of nonlinear binary error-correcting codes that are better than any linear code are the Nordstrom-Robinson [55, 59, 86, 98], Kerdock and Preparata codes [29, 46, 81, 86]. Calderbank et'al [29] showed that, when properly defined, Kerdock and Preparata codes are linear over the ring Z_4; and as Z_4-codes, they are the duals of each other. All these codes are in fact just extended cyclic codes [46, 81]. Since 1990's, coding theory researchers intensively study nonlinear codes [13, 76] that can be transformed into linear codes [26, 67, 74, 103, 104] in other metric spaces via appropriate mappings. Tokareva [104-108] used Krotov matrices [72, 73] to generate Z_4-linear codes [12, 14, 15, 45, 59, 71, 93, 99, 112] and from these codes she introduced *k*-affine binary functions, which are affine in an alternative sense. From *k*-affine functions, she then defined *k*-bent functions and a special form of the dot-product denoted as the *k*-dot product.

In this thesis, we firstly find a relation between two important tools for Boolean functions; Walsh transform null frequencies and covering sequences. Correlation immunity order, nonlinearity, resiliency and propagation characteristics of Boolean functions depend on the Walsh transform, which is related to the covering sequence of the function. Secondly, we derive new classes of affine and bent functions using linear codes over the ring Z_8. We then suggest cubic cryptanalysis, as an extended version of linear and quadratic cryptanalyses. We claim that the newly introduced class of *t*,*k*-bent functions are strong against cubic cryptanalysis, since they are as far as possible to affine, quadratic and cubic functions. Finally we examine the affine equivalence of *t*,*k*-bent functions and Maiorana McFarland class of bent functions.

The main background on properties and definitions of Boolean functions are introduced in Chapter 2.

In Chapter 3, we show that the Walsh transform null frequencies of Boolean functions are related to their covering sequences. We prove that each nonzero null frequency of the Walsh transform defines a covering sequence; however, in general the number of covering sequences is more than the number of Walsh transform nulls. We then present a lower bound for the number of covering sequences. We also show that the set of covering sequences given in our theorems 3.3 and 3.4 and those can be found from Proposition 3.2 given by Carlet and Tarannikov [40] are distinct. Then we study the covering sequences of affine equivalent Boolean functions.

Chapter 4 studies the Z_4 and Z_8-linear codes and the relation of these codes to newly introduced affine Boolean functions. We start by giving the origins of the the *k*-dot product and *k*-affine functions introduced by Tokareva [104-108]. Then we show that Krotov matrices [72, 73] have the lexicographically ordered codewords of the Z_4-linear (2^m, m) code C, as columns. Later we describe the rules that quadratic parts of *k*-dot products must obey. We then extend Tokareva's definitions to a larger ring, Z_8. We drive a new class of affine functions and a new *t,k*-dot product using linear codes over the ring Z_8. The new class of *t,k*-affine functions contain affine functions, quadratic functions and cubic functions. Examples of these functions are given at the end of Chapter 4.

In Chapter 5, we study bent functions including *k*-bent functions in detail. Then we suggest a new class, the *t,k*-bent functions depending on the *t,k*-dot product definition given in Chapter 4. The new class of bent functions are at maximum distance from the newly introduced affine functions, i.e., from affine functions, quadratic functions and cubic functions. Next we analyse the affine equivalence of *k*-bent and *t,k*-bent functions with the well known Maiorana McFarland class of bent functions. For the application to cryptology, we introduce the method of cubic cryptanalysis for block ciphers. It is a generalization of the well-known method of linear cryptanalysis given in 1993 by M. Matsui [79]. In our method we approximate Boolean functions by *t,k*- affine functions . The newly introduced *t,k*-bent functions are claimed to be strong against cubic cryptanalysis, since they are as far as possible to affine, quadratic and cubic functions.

Finally, we give our conclusions in Chapter 6.

CHAPTER 2

BOOLEAN FUNCTIONS; DEFINITIONS AND AFFINE EQUIVALENCY CLASSES

The aim of this chapter is to present a compact overview on the most essential aspects of Boolean functions related to cryptography. We describe two different ways of representing Boolean functions, the truth table and the algebraic normal form, in section 2.2. Next, we present two important tools to define cryptographic properties of Boolean functions, the Walsh and autocorrelation spectra in section 2.3. Remark 2.1 gives the relation between the Walsh transform and the Fourier transform, both are being widely used in cryptography. Section 2.4 gives necessary definitions and notations that will be used throughout the thesis. Remark 2.2 interprets the bentness criterion in terms of the White Gaussian Noise, which is a well-known subject in the telecomunications branch of electrical engineering. Then in section 2.5 a review of the affine equivalence classes is made.

2.1. Introduction

After Shannon's theory which proposes confusion and diffusion in secrecy systems [96] and the popularity of the subsequent Data Encryption Standard [11], S-boxes are studied widely in the literature [61, 66, 70, 84]. It has then been clearly demonstrated that differential and linear cryptanalysis [4, 9, 11, 19, 22, 42] can be resisted by the selection of nearly optimal Boolean functions as components of the S-boxes.

A Boolean function [61, 66, 70, 84, 96] produces a single-bit result $f(\mathbf{x}) \in GF(2)$ for each possible value of the m-bit vector, $\mathbf{x} \in GF(2)^m$. Boolean functions are used in cryptographic applications such as block ciphers, stream ciphers and hash functions. There are many criteria used to judge the suitability of a Boolean function for use in an encryption algorithm. The most desirable Boolean function properties are those, which strengthen the related cryptosystem against well known statistical attacks such as differential, linear cryptanalysis [4, 9, 11, 19, 22, 42] and algebraic attacks [3, 11, 16, 17, 34, 36]. Different criteria for Boolean functions such as balancedness, correlation-immunity [37, 63, 66, 73, 79, 89, 91, 118], resiliency, nonlinearity [51-53, 77, 78, 83] and algebraic degree [51-53, 77, 78, 83] are studied extensively in many works. It is known that some criteria cannot be satisfied simultaneously. So the problem is to find a trade-off between these criteria.

The classification of Boolean functions is meaningful in the sense that the number of representatives is much less than the number of all Boolean functions. Such perspectives allow the Boolean space to be considered as a structure in which all Boolean functions are grouped into equivalence classes and thus only one function from each class is enough for analysis.

2.2 Boolean Function Representations

We now present two representations of Boolean functions that we will use throughout the thesis; truth table (TT) and algebraic normal form (ANF). Other representations such as the numerical normal form representation and trace representation [25] also exist in the literature.

Let f be a Boolean function that produces a single-bit result for each possible combination of *m* Boolean variables; that is,

$$f(\mathbf{x}): GF(2)^m \rightarrow GF(2) \quad (2.1)$$

Here GF denotes the Galois Field consisting of binary numbers {0,1}, with modulo 2 addition (XOR operation shown by ⊕) and multiplication (AND operation shown by a dot or nothing).

2.2.1 Truth Table Representation of Boolean Functions

A Boolean function *f* can be uniquely represented by its truth table which is a vector that contains the function values of *f*, ordered lexicographically. In other words, the $1x2^m$ dimensional vector

$$\mathbf{f} = (f(0...00), f(0...01),... f(1...11) \quad (2.2)$$

is defined as the truth table of f, where the input vector $\mathbf{x}$ is ordered lexicographically. We mean by the weight and support of a function, the weight and support of the corresponding truth table. Analogously, the distance between two functions is computed by considering the distance between the corresponding truth tables.

2.2.2. Algebraic Normal Form Representation of Boolean Functions

Another way of uniquely representing a Boolean function *f* is by means of a polynomial in $GF(2)$ and is defined as the algebraic normal form. The corresponding transformation is called the algebraic normal transform:

$$ANF_f = \oplus_{(a_{m-1}\cdots a_0)\in GF(2)^m} h(a_{m-1}\cdots a_0) x_{m-1}^{a_{m-1}} \cdots x_0^{a_0} = \oplus_{\mathbf{a}} h(\mathbf{a})\mathbf{x}^{\mathbf{a}} \qquad (2.3)$$

where h is also a Boolean function on $GF(2)^m$. As the algebraic normal transform is a linear transformation, one can also use a matrix representation. Denoting the column matrix containing the coefficients $h(\mathbf{a})$ as $\mathbf{h_f}$, then with $\mathbf{f}$ representing the truth table of f,

$$\mathbf{h_f} = \mathbf{A_m f} \mod 2 \qquad (2.4)$$

where $\mathbf{A_m}$ is recursively determined by

$$\mathbf{A_0} = 1,\ \mathbf{A_1} = \begin{bmatrix} 1\,0 \\ 1\,1 \end{bmatrix} \text{ and } \mathbf{A_m} = \begin{bmatrix} 1\,0 \\ 1\,1 \end{bmatrix} \otimes \mathbf{A_{m-1}} \qquad (2.5)$$

The algebraic degree of f, denoted by deg(f) or shortly d, is defined as the maximum number of variables of the terms $x_{m-1}^{a_{m-1}} \cdots x_0^{a_0}$ in the ANF of f. Functions with algebraic degree less than or equal to 1 are called affine. If $f(0)= 0$ then the function is called linear.

2.3 Basic Tools Used to Define Cryptographic Properties of Boolean Functions

Two basic and important tools, Walsh and autocorrelation spectrum are defined in this section.

2.3.1. Walsh Transform of Boolean Functions

A Boolean function f can be uniquely represented by its Walsh transform. The Walsh transform of a Boolean function f is defined as

$$W_f(\mathbf{w}) = \sum_{\mathbf{x}\in GF(2)^m} (-1)^{f(\mathbf{x})}(-1)^{<\mathbf{w},\mathbf{x}>} \tag{2.6}$$

where $\mathbf{w}\in GF(2)^m$, $<\mathbf{w},\mathbf{x}>$ is the inner product of the vectors **w** and **x**. The $1\text{x}2^m$ dimensional vector

$$\mathbf{W_f} = (W_f(0...00), W_f(0...01),...W_f(1...11)) \tag{2.7}$$

is called the Walsh spectrum of f, where the input vector **w** is ordered lexicographically.

Remark 2.1: Sometimes, the Fourier transform $\hat{f}(\mathbf{w})$ is used instead of theWalsh transform. The Fourier transform of the function f at frequency **w** is defined as

$$\hat{f}(\mathbf{w}) = \sum_{\mathbf{a}\in GF(2)^m} f(\mathbf{a})(-1)^{<\mathbf{w},\mathbf{a}>}\ . \tag{2.8}$$

Walsh and Fourier transforms of a function f at frequency **w** are related by,

$$W_f(\mathbf{w}) = -2\hat{f}(\mathbf{w}) + 2^m\delta(\mathbf{w})\ . \tag{2.9}$$

where $\delta(\mathbf{w}) = \left\{\begin{matrix} 1 & \text{if } \mathbf{w}=0 \\ 0 & \text{else} \end{matrix}\right\}$ is the Kronecker delta function.

Definition 2.1: The support of the Walsh transform of f is defined as

$$Sup\{W_f\} = \{\mathbf{w}\in GF(2)^m \mid W_f(\mathbf{w}) \neq 0\}. \tag{2.10}$$

Notice that the support of the Walsh transform and the set of frequencies at which Fourier transform is nonzero are equal. Only one exception can occur if $W_f(0) = 0$.

2.3.2. Autocorrelation of Boolean Functions

The autocorrelation of a Boolean function is a real-valued function. To define the autocorrelation, we will first define the derivative of f with respect to the input difference vector $\mathbf{a} \in GF(2)^m$.

$$D_{\mathbf{a}} f(\mathbf{x}) = f(\mathbf{x}) \oplus f(\mathbf{x} \oplus \mathbf{a}) \tag{2.11}$$

The derivative vector is arrranged by ordering the index **x** lexicographically,

$$\mathbf{D_a f} = (D_a f(0...0), ..., D_a f(1...1)). \tag{2.12}$$

The autocorrelation of f corresponding to the shift vector **a** is denoted by

$$r_f(\mathbf{a}) = \sum_{\mathbf{x} \in GF(2)^m} (-1)^{f(\mathbf{x})} (-1)^{f(\mathbf{x} \oplus \mathbf{a})} = \sum_{\mathbf{x} \in GF(2)^m} (-1)^{\mathbf{D_a} f(\mathbf{x})} \tag{2.13}$$

All values of the autocorrelation can be collected in a $1 \mathrm{x} 2^m$ dimensional vector called the autocorrelation spectrum

$$\mathbf{r_f} = (r_f(0...00), r_f(0...01), ..., r_f(1...11)), \tag{2.14}$$

by ordering the index vector **a** lexicographically. Note that the autocorrelation spectrum does not uniquely determine the function in contrast to the previous transformations like ANF, truth table and the Walsh transform.

2.4 Basic Notations and Definitions

This section is intended as a summary of the minimum mathematical knowledge required throughout the thesis.

Definition 2.1: An *m*-variable Boolean function f is *balanced* if its output is equally distributed, i.e., its weight is equal to 2^{m-1}. This translates in $W_f(0)=0$ for the Walsh spectrum.

Definition 2.2: f is called r^{th} *order correlation immune* (*r*-CI) if [37]

$$W_f(\mathbf{w})=0, \left\{\forall \mathbf{w}\in GF(2)^m \middle| 1\le wt(\mathbf{w})\le r\right\}. \tag{2.15}$$

Definition 2.3: The combination of correlation immunity of order *r* and the property of balancedness results in the property of resiliency of order *r*.

Definition 2.4: *Nonlinearity* of f is defined as the minimum distance from the set of affine functions and one can show that it is related to the maximum magnitude in the Walsh spectrum of *f* as follows

$$NL_f = 2^{m-1} - \tfrac{1}{2}\max_{\mathbf{w}}\left| W_f(\mathbf{w})\right|. \tag{2.16}$$

Definition 2.5: An *m*-variable function *f*, with *m* even is called a *bent function* if its Walsh spectrum is flat, i.e., $W_f(\mathbf{w})=\pm 2^{m/2}$ or $W_f^2(\mathbf{w})=2^m$ for $\forall \mathbf{w}\in GF(2)^m$. Then the function has maximum nonlinearity, i.e., $NL_f = 2^{m-1}-2^{(m/2)-1}$.

Remark 2.2: Using (2.9) in Remark 2.1, it can be observed that $|W_f(\mathbf{w})|=2^{m/2}$ is true if and only if the magnitude of the Fourier spectrum is also flat except at **w**=0. This corresponds to White Gaussian Noise (WGN) spectrum (except for **w**=0). Hence a bent Boolean function has the Walsh and Fourier spectra similar to the power spectrum of WGN. The autocorrelation spectra of bent functions and WGN are also similar.

Definition 2.6: An mxn S-box is a mapping from m binary inputs to n binary outputs, i.e., $F(\mathbf{x}):GF(2)^m \to GF(2)^n$. The output vector of the S-box, $F(\mathbf{x})=(f_1(\mathbf{x}),\cdots,f_n(\mathbf{x}))$, can be decomposed into n component functions as, $f_i(\mathbf{x}):GF(2)^m \to GF(2), i=1,\cdots,n$.

Definition 2.7: The extended output of an $m\times n$ S-box can be obtained from its output vector by including all linear combinations of output bits. Thus the extended output vector G is composed of the functions

$$g_j(\mathbf{x})=\oplus_{i=1}^{n}\ j_i f_i = <\mathbf{j},\mathbf{F}>$$

where $\mathbf{j}=(j_1,\ j_2,...j_n)\in GF(2)^n$.

Definition 2.8: The set

$$R(r,m)=\{f(\mathbf{x})\,|\,\deg(f)\le r\} \tag{2.17}$$

denotes the r^{th} order Reed-Muller code of codeword length 2^m. The term $R(r,m)/R(s,m)$, where $s<r\le m$, defines the set of cosets of $R(r,m)$ with respect to $R(s,m)$ [8].

Definition 2.9 [40] A *covering sequence* of a function f is any sequence

$\boldsymbol{\lambda}=(\lambda_{00..0},\lambda_{0....01},.....\lambda_{11....1})=(\lambda_{\mathbf{a}})_{\mathbf{a}\in GF(2)^m}$ such that the derivative $\mathbf{D_a f}$ defined by (2.12) satisfies

$$\sum_{\mathbf{a}\in GF(2)^m}\lambda_{\mathbf{a}}\mathbf{D_a f}=(\rho\ \rho\ \ \ ...\rho)=\boldsymbol{\rho}\ \ , \tag{2.18}$$

where $\boldsymbol{\rho}$ is a vector with identical elements. The value of ρ is called the level of this sequence. If $\rho \neq 0$, then the covering sequence is said to be nontrivial [40].

Definition 2.10: Hadamard matrix $\mathbf{H_m}$ is an *mxm* matrix with entries +1 or -1, such that all rows and all columns are orthogonal, i.e., $\mathbf{H_m}\mathbf{H_m}^T = m\mathbf{I_m}$ where $\mathbf{H_m}^T$ is the transpose of the Hadamard matrix and $\mathbf{I_m}$ is the identity matrix of order *m*. A special kind of Hadamard matrix, called the Sylvester-Hadamard matrix of order 2^m denoted by $\mathbf{H_m}$ is generated by the following recursive relation

$$\mathbf{H_0} = 1, \qquad \mathbf{H_m} = \begin{bmatrix} \mathbf{H_{m-1}} & \mathbf{H_{m-1}} \\ \mathbf{H_{m-1}} & -\mathbf{H_{m-1}} \end{bmatrix} \tag{2.19}$$

It can be shown that each row (or column) of $\mathbf{H_m}$ is a linear sequence of length 2^m, i.e., it corresponds to the sequence of the linear function

$$l_{\mathbf{w}}(\mathbf{x}) = < \mathbf{w}, \mathbf{x} > . \tag{2.20}$$

Walsh transform of a function can easily be transformed into a matrix equation as,

$$\mathbf{W_f} = \left[W_f(0....0)\, W_f(0....1) \cdots W_f(1....1) \right] = \mathbf{H_m} \left[(-1)^{f(0...0)} \cdots (-1)^{f(1...1)} \right] \tag{2.21}$$

Remark 2.3: The product of the matrix $\mathbf{A_m}$ from the ANF transform and the Hadamard matrix $\mathbf{H_m}$ satisfies the following recursive relation for $m \geq 1$,

$$\mathbf{A_m}\mathbf{H_m} = \begin{bmatrix} 1\,1 \\ 2\,0 \end{bmatrix}^{[m]} \tag{2.22}$$

Definition 2.11: The maximum absolute value of the autocorrelation function of *f*(**x**) is given by

$$AI_f = \max_{\alpha \neq 0} \left| r_f(\boldsymbol{\alpha}) \right| \tag{2.23}$$

and is known as the *absolute indicator* [117].

The overall absolute indicator for the autocorrelation of an S-box [32-35] is defined in terms of the absolute indicators of the component functions (fi's)

$$AI_S = \max_i AI_{fi}. \tag{2.24}$$

Definition 2.12: For an mxn S-box as in Definition 2.6, the XOR table is a $2^m \times 2^n$ matrix with the (i,j)'th entry

$$k_{ij} = \#\{ \mathbf{x} \mid F(\mathbf{x}) \oplus F(\mathbf{x} \oplus \mathbf{i}) = \mathbf{j} \} \tag{2.25}$$

where $i=0,\ldots,2^m-1$ and $j=0,\ldots,2^n-1$ and the $1\times m$ vector $\mathbf{i}$ and $1\times n$ vector $\mathbf{j}$ are the corresponding binary representations respectively [4].

Definition 2.13: The largest entry in XOR table not including the (0,0)'th element gives the differential uniformity [17].

2.5 Affine Equivalence of Boolean Functions

We will give the definition of equivalence which then leads to affine equivalence of two m-variable Boolean functions.

Definition 2.14: [60] Two functions $f(\mathbf{x}), g(\mathbf{x}) \in R(r,m)$ are called equivalent with respect to $R(s,m)$, if there exists a nonsingular binary mxm matrix $\mathbf{A}$ and 1xm vector $\mathbf{b}$ such that

$$f(\mathbf{x}) = g(\mathbf{xA} \oplus \mathbf{b}) \bmod R(s,m). \tag{2.26}$$

In this case, due to the modulo operation,

$$f(\mathbf{x}) \oplus g(\mathbf{xA} \oplus \mathbf{b}) \in R(s,m). \tag{2.27}$$

$$f(\mathbf{x}) = g(\mathbf{xA} \oplus \mathbf{b}) \oplus v_s \tag{2.28}$$

where $v_s \in R(s,m)$.

Definition 2.15: [60] If one chooses $v_s \in R(1,m)$ then this equivalence equation becomes,

$$g(\mathbf{x}) = f(\mathbf{xA} \oplus \mathbf{b}) \oplus < \mathbf{x}, \mathbf{c} > \oplus d \tag{2.29}$$

where $\mathbf{c} \in GF(2)^m$ and $d \in GF(2)$. (2.29) is called the affine equivalence relation.

Proposition 2.1: [91] Let $f(\mathbf{x}), g(\mathbf{x})$ be two functions satisfying (2.29). Then for any $\mathbf{w} \in GF(2)^m$,

$$W_g(\mathbf{w}) = (-1)^{d \oplus <\mathbf{bA}^{-1}, (\mathbf{c} \oplus \mathbf{w})>} W_f(< (\mathbf{c} \oplus \mathbf{w}), \mathbf{A}^{-1} >) \tag{2.30}$$

Corollary 2.1: [100] The Walsh spectrum of $f(\mathbf{x})$ at $\mathbf{i}$ is equal to the Walsh spectrum of $g(\mathbf{x})$ at $\mathbf{j}$, where $\mathbf{j} = \mathbf{c} + \mathbf{iA^T}$. Therefore the distribution of the absolute values of the Walsh spectrum of $f(\mathbf{x})$ is same as that of $g(\mathbf{x})$.

Proposition 2.2: [91] Let $f(\mathbf{x})$ and $g(\mathbf{x})$ be two functions such that $g(\mathbf{x}) = f(\mathbf{xA} \oplus \mathbf{b}) \oplus < \mathbf{c}, \mathbf{x} >$. Then for any given $\mathbf{s} \in GF(2)^m$, $r_g(\mathbf{s}) = (-1)^{<\mathbf{c},\mathbf{s}>} r_f(\mathbf{sA})$.

Corollary 2.2: [100] The autocorrelation function of $f(\mathbf{x})$ at $\mathbf{j}$ is equal to the autocorrelation function of $g(\mathbf{x})$ at $\mathbf{i}$; where $\mathbf{j} = \mathbf{iA}$: Therefore the ranks of vectors with the same absolute autocorrelation function value are same between two equivalent functions. Hence, the distribution of the absolute values of the autocorrelation function of $f(\mathbf{x})$ is same as that of $g(\mathbf{x})$.

Proposition 2.3: [100] For any Boolean function $f(\mathbf{x}) \in R(r,m)$, derivative is

$$D_a(f \circ B) = D_{aA}(f) \circ B(\mathbf{x}) \tag{2.31}$$

where $B(\mathbf{x}) = \mathbf{xA} \oplus \mathbf{b}$. Here “o“ denotes function combination operation.

CHAPTER 3

RELATION BETWEEN COVERING SEQUENCE AND WALSH TRANSFORM NULL FREQUENCIES

In this chapter, we show that the Walsh transform null frequencies of Boolean functions are related to their covering sequences. We show that some covering sequences of a Boolean function can be obtained using the Walsh transform nulls. We prove that each nonzero null frequency of the Walsh transform defines one covering sequence; and if the Boolean function is balanced, each null is associated with two covering sequences. We present a lower bound for the number of covering sequences and confirm that the set of covering sequences that we find from Walsh transform nulls are distinct from those given by Carlet and Tarannikov.

We then present a lower bound for the number of covering sequences. We also show that the set of covering sequences given in our theorems 3.3 and 3.4 and those can be found from Proposition 3.2 given by Carlet and Tarannikov are distinct. Then, we study the covering sequences of affine equivalent Boolean functions.

3.1. Introduction

Covering sequences are introduced in 2000 by Carlet and Tarannikov [40] as an efficient tool to study Boolean functions. These are binary-valued sequences $\boldsymbol{\lambda} \in GF(2)^{2^m}$ that are related to the function via its derivatives $D_{\mathbf{a}} f(\mathbf{x}) = f(\mathbf{x}) \oplus f(\mathbf{x} \oplus \mathbf{a})$. Carlet and Tarannikov show for any Boolean function that, balancedness and admitting a nontrivial covering sequence are equivalent. They also obtain a characterization of correlation-immune and resilient functions by means of covering sequences.

In this chapter, we show that,

i) in Theorem 3.3, for any *m*-variable Boolean function *f*, each nomzero Walsh transform null $\mathbf{w} \in GF(2)^m$ defines a covering sequence $\boldsymbol{\lambda} \in GF(2)^{2^m}$ with elements $\lambda_{\mathbf{a}} = (-1)^{<\mathbf{w},\mathbf{a}>}$ and for each covering sequence $\boldsymbol{\lambda}$ which can be represented as $\lambda_{\mathbf{a}} = (-1)^{<\mathbf{w},\mathbf{a}>}$, there exists a nonzero Walsh transform null **w**.

ii) in Theorem 3.4, for a balanced *n*-variable Boolean function *f*, each nonzero Walsh transform null $\mathbf{w} \in GF(2)^m$ defines a covering sequence $\boldsymbol{\lambda} \in GF(2)^{2^m}$ with elements $\lambda_{\mathbf{a}} =< \mathbf{w}, \mathbf{a} >$ and for each covering sequence $\boldsymbol{\lambda}$ which can be represented as $\lambda_{\mathbf{a}} =< \mathbf{w}, \mathbf{a} >$, there exists a nonzero Walsh transform null **w,** and

We also show that all the covering sequences calculated from Theorem 3.4 are linearly independent and none of them can be an indicator of a subspace. Therefore, the set of covering sequences which can be calculated from Proposition 3.2 given by Carlet and Mesnager [39] and our theorems 3.3 and 3.4 are proven to be distinct.

3.2. Already Known Facts on Covering Sequences

For a Boolean function, Carlet and Tarannikov has shown the equivalence between its balancedness and the fact it admits a covering sequence. They also obtain a characterization of correlation-immune and resilient functions by means of covering sequences. Carlet and Tarannikov results are given as therorems and propositions 3.1 and 3.2. In section 3.4, we give the relation between covering sequence and Walsh transform null frequencies. Correlation immunity order can be found from Walsh transform nulls. Thus results of Carlet and Tarannikov are related to our findings. At first, the definition of the covering sequence of a Bolean functions is given.

Definition 3.1: [40] The covering sequence of an *m*-variable function f is any sequence

$$\boldsymbol{\lambda} = \left(\lambda_{00..0}, \lambda_{0....01}, ...\lambda_{11....1}\right) = (\lambda_{\mathbf{a}})_{\mathbf{a} \in GF(2)^m}$$

(where the index vector **a** is ordered lexicographically) such that

$$\sum_{\mathbf{a} \in GF(2)^m} \lambda_{\mathbf{a}} \mathbf{D_a f} = (\rho\ \rho\ \ldots\ \rho) = \boldsymbol{\rho} \tag{3.1}$$

is a vector with identical elements and the derivative $\mathbf{D_a f}$ is defined by (2.11). The value of ρ is called the level of this sequence. If $\rho \neq 0$, then the covering sequence is said to be nontrivial.

Proposition 3.1 [40]**:** Let f be a Boolean function on $GF(2)^m$. Assume that there exist numbers $\lambda_{\mathbf{a}} \in Z$, $\mathbf{a} \in GF(2)^m$ and a nonzero number ρ such that $\sum_{\mathbf{a} \in GF(2)^n} \lambda_{\mathbf{a}} \mathbf{D_a f}$ is equal to the constant function ρ. Then f is balanced. Conversely, assume that f is balanced, then the integer valued function $\sum_{\mathbf{a}\in GF(2)^m} \mathbf{D_a f}$ is constant and equal to 2^{m-1}.

Theorem 3.1: [40] Let f be any Boolean function on $GF(2)^m$ and $\boldsymbol{\lambda} = (\boldsymbol{\lambda}_\mathbf{a})_{\mathbf{a}\in GF(2)^m}$ be any sequence. f admits λ as covering sequence if and only if its Fourier transform $\hat{\lambda}(\mathbf{w})$ takes constant value on the support of the Walsh transform W_f, i.e., for all frequencies $\{\mathbf{w} \in GF(2)^m \mid W_f(\mathbf{w}) \neq 0\}$. Let r be this constant value, then the level of this covering sequence is the number $\frac{1}{2}\left[\left(\sum_{\mathbf{a}\in GF(2)^m} \lambda_\mathbf{a}\right) - r\right]$.

Theorem 3.2: [40] Let f be any Boolean function on $GF(2)^m$.

1- If f admits a covering sequence $\boldsymbol{\lambda} = (\boldsymbol{\lambda}_\mathbf{a})_{\mathbf{a}\in GF(2)^m}$ with level ρ (resp. with level $\rho \neq 0$), then f is k^{th} order correlation-immune (resp. k-resilient), where (k +1) is the minimum Hamming weight of nonzero $\mathbf{b} \in GF(2)^m$ such that $\hat{\lambda}(\mathbf{b}) = r$, and $r = \hat{\lambda}(0) - 2\rho$.

2- Conversely if f is k^{th} order CI and it is not $(k+1)^{\text{th}}$ order CI then there exists one trivial covering sequence $\boldsymbol{\lambda} = (\boldsymbol{\lambda}_\mathbf{a})_{\mathbf{a}\in GF(2)^m}$ with level ρ such that k+1 is the minimum Hamming weight of nonzero $\mathbf{b} \in GF(2)^m$ satisfying

$$\hat{\lambda}(\mathbf{b}) = \hat{\lambda}(0) - 2\rho. \tag{3.2}$$

The proof of Theorem 3.2 is given in [40]. The following proposition requires the definition of the indicator for a given set, A, of vectors.

Definition 3.2: The indicator I_A is a binary 2^m–dimensional vector, each element $I_A(\mathbf{x})$ of which is indicating the existence or nonexistence of (lexicographically ordered) $GF(2)^m$ elements within the set A, i. e.,

$$\mathbf{I}_A(\mathbf{x}) = \begin{cases} 1 \text{ if } \mathbf{x} \in A \\ 0 \text{ if } \mathbf{x} \notin A \end{cases} \tag{3.3}$$

Hence, the Hamming weight of I_A is equal to |A|, the number of elements in A.

Proposition 3.2: [39] Let E be any vector subspace of $GF(2)^m$ and $(\mathbf{u}\oplus E)$ be any of its cosets. Let f be a Boolean function on $GF(2)^m$. Assume it admits no derivative $\mathbf{D_a f}$ equal to the constant function 1. Then f admits the indicator of $(\mathbf{u}\oplus E)$ as a nontrivial covering sequence if and only if the support of $W_f(\mathbf{w})$ is disjoint from $E^{\perp} = \{\mathbf{x} \in GF(2)^m \mid \mathbf{v}^T \mathbf{x} = 0, \forall \mathbf{v} \in E\}$. This is equivalent to the fact that the restriction of f to any coset of E is balanced. The level of this covering sequence is then equal to $|E|/2$ and the indicator of every coset of E is also a covering sequence of f with the same level. More generally, any sequence $\boldsymbol{\lambda}$ such that for every $\mathbf{a} \in E$ and every $\mathbf{u} \in GF(2)^m$, $\lambda_{\mathbf{a}+\mathbf{u}} = \lambda_{\mathbf{u}}$ is also a covering sequence of f.

3.3. Relation Between Covering Sequences and Walsh Transform Null Frequencies

Our aim in this section is to find relations between covering sequences and Walsh transform null frequencies of a Boolean function.

Theorem 3.3: Let f be any Boolean function on $GF(2)^m$ and $W_f(\mathbf{w})$ be its Walsh transform at frequency **w**.

1- For all nonzero Walsh transform nulls **w**, there exists a (-1,+1)-valued covering sequence $\boldsymbol{\lambda} = (\boldsymbol{\lambda}_{\mathbf{a}})_{\mathbf{a} \in GF(2)^m}$ with elements $\lambda_{\mathbf{a}} = (-1)^{<\mathbf{w},\mathbf{a}>}$. (3.4)

2- For all covering sequences which can be represented as $\boldsymbol{\lambda} = (\boldsymbol{\lambda}_{\mathbf{a}})_{\mathbf{a} \in GF(2)^m}$ with elements $\lambda_{\mathbf{a}} = (-1)^{<\mathbf{w},\mathbf{a}>}$, there is a nonzero Walsh transform null **w**.

Proof:

1- A Walsh transform null frequency **w** satisfies

$$W_f(\mathbf{w}) = \sum_{\mathbf{x} \in GF(2)^m} (-1)^{f(\mathbf{x})} (-1)^{<\mathbf{w},\mathbf{x}>} = \sum_{\mathbf{x} \in GF(2)^m} (-1)^{f(\mathbf{x})} (-1)^{<\mathbf{w},\mathbf{x}>} = 0. \qquad (3.5)$$

Hence, $f(\mathbf{x}) \oplus < \mathbf{w},\mathbf{x} >$ is balanced for all Walsh transfrom null frequencies **w**. Using Proposition 3.3,

$$\sum_{\mathbf{a} \in GF(2)^m} \mathbf{D}_{\mathbf{a}}(f(\mathbf{x}) \oplus < \mathbf{w},\mathbf{x} >) = \left(2^{m-1} \ldots \ 2^{m-1}\right). \qquad (3.6)$$

Using the definition of derivative of a vector from (2.8) we have,

$$\mathbf{D}_{\mathbf{a}}(f(\mathbf{x}) \oplus < \mathbf{w},\mathbf{x} >) = f(\mathbf{x}) \oplus < \mathbf{w},\mathbf{x} > \oplus f(\mathbf{x} \oplus \mathbf{a}) \oplus < \mathbf{w},(\mathbf{x} \oplus \mathbf{a}) >= \mathbf{D}_{\mathbf{a}}\mathbf{f} \oplus < \mathbf{w},\mathbf{a} > \qquad (3.7)$$

$$\mathbf{D_a}(f(\mathbf{x}) \oplus <\mathbf{w},\mathbf{x}>) = \begin{Bmatrix} \mathbf{D_a f}, & \text{if } <\mathbf{w},\mathbf{a}>=0 \\ \mathbf{D_a f} \oplus \mathbf{1}, & \text{if } <\mathbf{w},\mathbf{a}>=1 \end{Bmatrix}, \tag{3.8}$$

$$\mathbf{D_a}(f(\mathbf{x}) \oplus <\mathbf{w},\mathbf{x}>) = \begin{Bmatrix} \mathbf{D_a f}, & \text{if } <\mathbf{w},\mathbf{a}>=0 \\ \mathbf{1} - \mathbf{D_a f}, & \text{if } <\mathbf{w},\mathbf{a}>=1 \end{Bmatrix}, \tag{3.9}$$

Using (3.9), the binary '⊕' addition in (3.8), becomes an integer '+' addition in (3.10).

$$\mathbf{D_a}(f(\mathbf{x}) \oplus <\mathbf{w},\mathbf{x}>) = (-1)^{<\mathbf{w},\mathbf{a}>}\mathbf{D_a f} + \begin{Bmatrix} \mathbf{0}, & \text{if } <\mathbf{w},\mathbf{a}>=0 \\ \mathbf{1}, & \text{if } <\mathbf{w},\mathbf{a}>=1 \end{Bmatrix}. \tag{3.10}$$

For 2^m possible $\mathbf{a}$ vectors, in $2^{m\text{-}1}$ cases $<\mathbf{w},\mathbf{a}>=0$ and in $2^{m\text{-}1}$ cases $<\mathbf{w},\mathbf{a}>=1$. Then

$$(2^{m-1}...2^{m-1}) + \sum_{\mathbf{a}\in GF(2)^m} (-1)^{<\mathbf{w},\mathbf{a}>}\mathbf{D_a f} = (2^{m-1}...2^{m-1}). \tag{3.11}$$

Therefore

$$\sum_{\mathbf{a}\in GF(2)^m} (-1)^{<\mathbf{w},\mathbf{a}>}\mathbf{D_a f} = (0...0). \tag{3.12}$$

Recall the covering sequence relation

$$\sum_{\mathbf{a}\in GF(2)^m} \lambda_{\mathbf{a}}\mathbf{D_a f} = (\rho \quad \rho...\rho). \tag{3.13}$$

Comparing (3.12) and (3.13), one can find the covering sequence in (3.12) as

$$\lambda_{\mathbf{a}} = (-1)^{<\mathbf{w},\mathbf{a}>}. \tag{3.14}$$

excluding **w** = **0**. Notice that for **w** = **0**, we have $(-1)^{<\mathbf{w},\mathbf{a}>} = 1$ and (3.12) can be satisfied for only constant functions. Thus (3.14) is valid except for **w** = **0,** which is a Walsh transform null of only balanced functions.

Hence, $\boldsymbol{\lambda} = (\boldsymbol{\lambda}_{\mathbf{a}})_{\mathbf{a} \in GF(2)^m}$ with elements $\lambda_{\mathbf{a}} = (-1)^{<\mathbf{w},\mathbf{a}>}$ is a (+1,-1) valued trivial covering sequence.

2- Since the proof steps (3.4) to (3.14) are equalities that can be repeated in the reverse direction, the second statement of Theorem 3.3 is also proved simultaneously.

Theorem 3.4: Let f be a balanced Boolean function on $\mathbf{w} \in GF(2)^m$ and $W_f(\mathbf{w})$ be its Walsh transform at frequency **w**.

1- For all nonzero Walsh transform nulls **w**, there exists a (0,1)-valued covering sequence $\boldsymbol{\lambda} = (\boldsymbol{\lambda}_{\mathbf{a}})_{\mathbf{a} \in GF(2)^m}$ with elements $\lambda_{\mathbf{a}} =< \mathbf{w}, \mathbf{a} >$. (3.15)

2- For all covering sequences which can be represented as $\boldsymbol{\lambda} = (\boldsymbol{\lambda}_{\mathbf{a}})_{\mathbf{a} \in GF(2)^m}$ with elements $\lambda_{\mathbf{a}} =< \mathbf{w}, \mathbf{a} >$, there is a nonzero Walsh transform null **w**.

Proof:

1- Starting from (3.12), and since $(-1)^{<\mathbf{w},\mathbf{a}>} = 1 - 2 < \mathbf{w}, \mathbf{a} >$,

$$\sum_{\mathbf{a} \in GF(2)^m} (1 - 2 < \mathbf{w}, \mathbf{a} >) \mathbf{D}_{\mathbf{a}} \mathbf{f} = (0,...,0). \tag{3.16}$$

(3.16) can also be written as

$$2 \sum_{\mathbf{a} \in GF(2)^m} (< \mathbf{w}, \mathbf{a} >) \mathbf{D}_{\mathbf{a}} \mathbf{f} + \sum_{\mathbf{a} \in GF(2)^m} \mathbf{D}_{\mathbf{a}} \mathbf{f} = (0,...,0). \tag{3.17}$$

It is easy to see that

$$\sum_{\mathbf{a}\in GF(2)^m}(<\mathbf{w},\mathbf{a}>)\mathbf{D_a f} = \tfrac{1}{2}\left(\sum_{\mathbf{a}\in GF(2)^m}\mathbf{D_a f}\right). \tag{3.18}$$

For a balanced function *f*, $\sum_{\mathbf{a}\in GF(2)^m}\mathbf{D_a f} = (2^{m-1}\ 2^{m-1}\ \ldots\ 2^{m-1})$ by Proposition 3.1.

Hence,

$$\sum_{\mathbf{a}\in GF(2)^m}(<\mathbf{w},\mathbf{a}>)\mathbf{D_a f} = (2^{m-2}\ 2^{m-2}\ \ldots\ 2^{m-2}) \tag{3.19}$$

Comparing the covering sequence equation $\sum_{\mathbf{a}\in GF(2)^m}\lambda_{\mathbf{a}}\mathbf{D_a f} = (\rho\quad \rho\ldots\rho)$ and (3.19) one gets $\lambda_{\mathbf{a}} =<\mathbf{w},\mathbf{a}>$, excluding **w** = **0**. Notice that for **w** = **0** , we have $<\mathbf{w},\mathbf{a}>=0$ and (3.19) can not be satisfied. Thus $\boldsymbol{\lambda} = (\lambda_{\mathbf{a}})_{\mathbf{a}\in GF(2)^m}$ with elements $\lambda_{\mathbf{a}} =<\mathbf{w},\mathbf{a}>$ is a covering sequence of the balanced function *f* with level $\rho = 2^{m-2}$ except for **w**=**0** which is a Walsh transform null for all balanced functions.

2- Since the proof steps (3.4) to (3.19) are equalities that can be repeated in the reverse direction, the second statement of Theorem 3.4 is also proved simultaneously.

We now give corollaries 3.1 and 3.2 for theorems 3.3 and 3.4.

Corollary 3.1: (i) Let $\mathbf{w} = (w_m,\ldots, w_2, w_1)$ be the nonzero Walsh transform null frequency of a Boolean function *f*, and $\boldsymbol{\lambda}$ be the corresponding (-1,+1)-valued covering sequence with elements $\lambda_{\mathbf{a}} = (-1)^{<\mathbf{w},\mathbf{a}>}$. Then, $\lambda_{\mathbf{0}}$ =1 and for any two indices **a** and **b**, the element $\lambda_{\mathbf{a}\oplus\mathbf{b}} = \lambda_{\mathbf{a}}\lambda_{\mathbf{b}}$.

(ii) Similarly, any (-1,+1)-valued covering sequence **λ,** with elements satisfying the property $\lambda_{\mathbf{a}\oplus\mathbf{b}} = \lambda_{\mathbf{a}}\lambda_{\mathbf{b}}$ and $\lambda_{\mathbf{0}}=1$ implies a nonzero Walsh transform null frequency**,** $\mathbf{w}=(w_m,..., w_2, w_1)$, which is equal to $(1,1,\dots 1)-\frac{(\lambda_{10\dots0},\dots\lambda_{0\dots10},\lambda_{0\dots01})}{2}$. Each element of the vector **w** can be found using $w_a = 1-\frac{\lambda_{\mathbf{a}}}{2}$ for all $\mathbf{a} \mid wt(\mathbf{a})=1$.

Proof: (i) $\lambda_{\mathbf{0}} = \lambda_{00..0} = (-1)^{<(00..0),\mathbf{w}>} = (-1)^0 = 1$ and

$$\lambda_{\mathbf{a}\oplus\mathbf{b}} = (-1)^{<\mathbf{w},\mathbf{a}\oplus\mathbf{b}>} = (-1)^{<\mathbf{w},\mathbf{a}>\oplus<\mathbf{w},\mathbf{b}>} = (-1)^{<\mathbf{w},\mathbf{a}>}(-1)^{<\mathbf{w},\mathbf{b}>} = \lambda_{\mathbf{a}}\lambda_{\mathbf{b}} \quad (3.20)$$

(ii) Using $\lambda_{\mathbf{a}\oplus\mathbf{b}} = \lambda_{\mathbf{a}}\lambda_{\mathbf{b}}$ with $\lambda_{\mathbf{0}}=1$ and the fact that the covering sequence is (-1,+1)-valued, its elements can be represented as $\lambda_{\mathbf{a}}=(-1)^{<\mathbf{k},\mathbf{a}>}$. For all weight 1 indexed terms this becomes $\lambda_{\mathbf{a}}=(-1)^{k_a}$, k_a being the a^{th} bit of vector **k**. Since all binary vectors can be represented as a sum of vectors of weight 1 knowledge of covering sequence elements with weight 1 is sufficient to calculate other elements. This can be shown by (3.21) as,

$$\boldsymbol{\lambda} = (\lambda_{0..0},\ ...,\lambda_{1..1}) = (0,\ (-1)^{k_1},\ (-1)^{k_2},\ (-1)^{(k_1\oplus k_2)},\ \dots\ ,\ (-1)^{(k_1\oplus\dots\oplus k_m)}) \quad (3.21)$$

Since one can express all weight-1 indexed terms as

$$\lambda_{\mathbf{a}} = (-1)^{<\mathbf{k},\mathbf{a}>} = (-1)^{k_a} = 1-2\mathbf{k}_{\mathbf{a}},\ k_a = 1-\frac{\lambda_{\mathbf{a}}}{2} \quad (3.22)$$

The corresponding vector $\mathbf{k}=(k_m,...,k_2,k_1)$ is a Walsh transform null by Theorem 3.3. Denoting **k** by **w**, from (3.22)

$$\mathbf{w} = (w_m, ..., w_{2,} w_1) = (1, 1, ... 1) - \frac{(\lambda_{10...0}, ...\lambda_{0...10}, \lambda_{0...01})}{2}, \text{ using } \lambda_{\mathbf{a}} = (-1)^{<\mathbf{w},\mathbf{a}>}$$

for $\forall \mathbf{a} \mid wt(\mathbf{a}) = 1$. (3.23)

We now give Corollary 3.2 for Theorem 3.4.

Corollary 3.2: i) Let $\mathbf{w} = (w_m, ..., w_2, w_1)$ be the nonzeroWalsh transform null frequency of a balanced Boolean function *f*, and $\boldsymbol{\lambda}$ be the corresponding (0,1)-valued covering sequence with elements $\lambda_{\mathbf{a}} =< \mathbf{w}, \mathbf{a} >$. Then, $\lambda_{\mathbf{0}}$ =0 and for any two indices **a** and **b**, the element $\lambda_{\mathbf{a} \oplus \mathbf{b}} = \lambda_{\mathbf{a}} \oplus \lambda_{\mathbf{b}}$.

ii) Similarly, any covering sequence $\boldsymbol{\lambda}$**,** with elements satisfying the property $\lambda_{\mathbf{a} \oplus \mathbf{b}} = \lambda_{\mathbf{a}} \oplus \lambda_{\mathbf{b}}$ and $\lambda_{\mathbf{0}}$ =0 implies a nonzero Walsh transform null frequency **w,** $\mathbf{w} = (w_m, ..., w_{2,} w_1) = (\lambda_{10...0}, ...\lambda_{0...10}, \lambda_{0...01})$. Each element of the vector **w** can be found using $\lambda_{\mathbf{a}} =< \mathbf{w}, \mathbf{a} >$ for $\forall \mathbf{a} \mid wt(\mathbf{a}) = 1$.

Proof: (i) $\lambda_{\mathbf{0}} = \lambda_{00..0} =< (00...0), \mathbf{w} >= 0$ and

$$\lambda_{\mathbf{a} \oplus b} =< \mathbf{w}, \mathbf{a} \oplus \mathbf{b} >=< \mathbf{w}, \mathbf{a} > \oplus < \mathbf{w}, \mathbf{b} >= \lambda_{\mathbf{a}} \oplus \lambda_b \quad (3.24)$$

(ii) Using $\lambda_{\mathbf{a} \oplus \mathbf{b}} = \lambda_{\mathbf{a}} \oplus \lambda_{\mathbf{b}}$ with $\lambda_{\mathbf{0}}$ =0 and the fact that the covering sequence is (0,1)-valued, its elements can be represented as $\lambda_{\mathbf{a}} =< \mathbf{k}, \mathbf{a} >$. For all weight 1 indexed terms this becomes $\lambda_{\mathbf{a}} = k_a$, the a^{th} bit of vector **k**. Since all binary vectors can be represented as a sum of vectors of weight 1 knowledge of covering sequence elements with weight 1 is sufficient to calculate other elements. This can be shown by (3.25) as,

$$\boldsymbol{\lambda} = (\lambda_{0..0}, ..., \lambda_{1..1}) = (0, \; k_1, \; k_2, \; (k_1 \oplus k_2), \; ... \; , \; (k_1 \oplus ... \oplus k_m)). \quad (3.25)$$

The corresponding vector $\mathbf{k} = (k_m,..., k_2, k_1)$ is a Walsh transform null by Theorem 3. Call now **k** as **w**. From (3.25)

$\mathbf{w} = (w_m,..., w_{2,} w_1) = (\lambda_{10...0},...\lambda_{0...10}, \lambda_{0...01})$, using $\lambda_{\mathbf{a}} = <\mathbf{w}, \mathbf{a}>$ for $\forall \mathbf{a} \mid wt(\mathbf{a}) = 1$.

Corollary 3.3: For any Boolean function, number of covering sequences is greater than or equal to the number of Walsh transform nulls, i.e.,

(# of covering sequences) ≥ (# of Walsh transform null frequencies). (3.26)

Proof: Because of the relations (3.14) and (3.15), each Walsh transform null defines a covering sequence; hence, the minimum number of covering sequences is equal to the number of Walsh zeros. Inequality occurs either when *f* is balanced or there are other covering sequences that cannot be represented as $\lambda_{\mathbf{a}} = <\mathbf{w}, \mathbf{a}>$ or $\lambda_{\mathbf{a}} = (-1)^{<\mathbf{w},\mathbf{a}>}$.

Corollary 3.4: Hamming weight of the covering sequence of a balanced function calculated from any nonzero Walsh transform null frequency through equation (3.15) is 2^{m-1}.

Proof: Let **w** be a nonzero Walsh transform null; $W_f(\mathbf{w}) = 0$ and let $\boldsymbol{\lambda} = (\boldsymbol{\lambda}_{\mathbf{a}})_{\mathbf{a} \in GF(2)^m}$ be the corresponding covering sequence. The Hamming weight of $\boldsymbol{\lambda}$ is,

$$wt(\boldsymbol{\lambda}) = \sum_{\mathbf{a} \in GF(2)^m} \lambda_{\mathbf{a}} = \sum_{\mathbf{a} \in GF(2)^m} <\mathbf{w}, \mathbf{a}> \tag{3.27}$$

If $\mathbf{w} = (0 \ ... \ 0)$ then $wt(\boldsymbol{\lambda}) = 0$. Assuming $\mathbf{w} \neq (0 \ ... \ 0)$ and using (3.12),

$$\sum_{\mathbf{a}\in GF(2)^m} <\mathbf{w},\mathbf{a}> = \frac{1}{2}\sum_{\mathbf{a}\in GF(2)^m} 1-(-1)^{<\mathbf{w},\mathbf{a}>} = 2^{n-1} - \frac{1}{2}\sum_{\mathbf{a}\in GF(2)^m} (-1)^{<\mathbf{w},\mathbf{a}>} \quad (3.28)$$

$$\sum_{\mathbf{a}\in GF(2)^m} (-1)^{<\mathbf{w},\mathbf{a}>} = \sum_{\mathbf{a}\in GF(2)^m} (-1)^{<\mathbf{w},\mathbf{a}>}(-1)^{<\mathbf{b},\mathbf{a}>}\Big|_{\mathbf{b}=0} = \delta(\mathbf{w}\oplus\mathbf{b}) = \begin{cases} 1 & \text{if } \mathbf{b}=\mathbf{w} \\ 0 & \text{else} \end{cases} \quad (3.29)$$

Since $\mathbf{b}=(0\ \ldots\ 0)$, and $\mathbf{w}\neq(0\ \ldots\ 0)$, $wt(\boldsymbol{\lambda})=2^{m-1}$ for any nonzero covering sequence calculated from (3.15).

Corollary 3.5: Hamming weights of the covering sequences calculated from Proposition 3.2 of Carlet and Tarannikov, where k is the dimension of the largest subspace $E^{\perp}=\{\mathbf{x}\in GF(2)^m \mid \mathbf{v}^T\mathbf{x}=0, \forall\mathbf{v}\in E\}$ constructed by Walsh transform nulls of an m-variable Boolean function f, are all 2^{m-k}.

Proof: In Proposition 3.2, the indicator of every coset of E is given to be a covering sequence $\boldsymbol{\lambda}$ of function f. Then, $wt(\boldsymbol{\lambda})=2^{\dim(E)}=2^{m-k}$.

Corollary 3.6: Any pair of covering sequences $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}'$ calculated from Walsh transform null frequencies through (3.15) are linearly independent, i.e.,

$$k\boldsymbol{\lambda}+j\boldsymbol{\lambda}'\neq(0\ \ldots\ 0) \text{ for any integers } k, j\neq 0 \text{ and } \boldsymbol{\lambda}\neq\boldsymbol{\lambda}' \quad (3.30)$$

Proof: Let $\mathbf{w}$ and $\mathbf{w}'$ be two Walsh transform nulls; $W_f(\mathbf{w})=0$, $W_f(\mathbf{w}')=0$ and $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}'$ be the corresponding covering sequences, so

$\boldsymbol{\lambda}=(\lambda_{\mathbf{a}}=<\mathbf{w},\mathbf{a}>)_{\mathbf{a}\in GF(2)^m}$ and $\boldsymbol{\lambda}'=(\lambda'_{\mathbf{a}}=<\mathbf{w}',\mathbf{a}>)_{\mathbf{a}\in GF(2)^m}$. We will use proof by contradiction. Now assume that $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}'$ are linearly dependent.

Then $$k\boldsymbol{\lambda}+j\boldsymbol{\lambda}'=(0\ \ldots\ 0) \quad \text{for } k, j\neq 0 \quad (3.31)$$

Now using (3.31),

$$\begin{aligned} &k(\lambda_0,\ w_1,\ w_2,\ (w_1 \oplus w_2),\ \ldots\ ,\ (w_1 \oplus \ldots \oplus w_m)) \oplus \\ &j(\lambda_0',\ w_1',\ w_2',\ (w_1' \oplus w_2'),\ \ldots\ ,\ (w_1' \oplus \ldots \oplus w_m')) = (0\ 0\ \ldots \cdots 0) \end{aligned} \tag{3.32}$$

(3.31) holds if and only if $k\,w_i \oplus jw_i' = 0\ \forall i \in \{0,1,\ldots m\}$. Notice that $\boldsymbol{\lambda} \neq \boldsymbol{\lambda}'$ implies that $\mathbf{w} \neq \mathbf{w}'$**;** hence there exists at least one w_i such that $w_i \neq w_i'$. Without lost of generality assume $w_i = 0$ and $w_i' = 1$. $k\boldsymbol{\lambda} + j\boldsymbol{\lambda}' = (0\ \ldots\ 0)$ implies that k=0. Therefore j=0, which contradicts the assumption of (3.31). Hence, $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}'$ are linearly independent.

Theorem 3.5: The covering sequences calculated from Walsh transform null frequencies through equation (3.15) can not be indicators (see (3.3) for the definition) of any subspace.

Proof: The elements of a covering sequence that satisfies $\boldsymbol{\lambda} = (\lambda_{\mathbf{a}} =< \mathbf{w}, \mathbf{a} >)_{\mathbf{a} \in GF(2)^m}$ are related to each other by (3.20). We will use the proof by contradiction. Assume λ is an indicator of a subspace E. Then λ satisfies

$$\lambda_a = \begin{cases} 1 & \text{if} \quad \mathbf{a} \in E \\ 0 & \text{if} \quad \mathbf{a} \notin E \end{cases} \tag{3.33}$$

Let $\mathbf{a}, \mathbf{b} \in E$, as λ is the indicator of E, $\lambda_{\mathbf{a}} = 1, \lambda_{\mathbf{b}} = 1$. Since E is a subspace, it is closed, so $(\mathbf{a} \oplus \mathbf{b}) \in E$; therefore, $\lambda_{\mathbf{a} \oplus \mathbf{b}} = 1$. However, λ obtained by (3.15) should also satisfy Corollary 3.1, which implies $\lambda_{\mathbf{a} \oplus \mathbf{b}} = 1 \oplus 1 = 0$. This is a contradiction. Hence $\boldsymbol{\lambda} = (\lambda_{\mathbf{a}} =< \mathbf{w}, \mathbf{a} >)$ can not be the indicator of any subspace.

In the rest of this paper, k will refer to the dimension of the largest subspace

$E^{\perp} = \{\mathbf{x} \in GF(2)^m \mid \mathbf{v}^T \mathbf{x} = 0, \forall \mathbf{v} \in E\}$ constructed by Walsh transform nulls of an m-variable Boolean function f. Then, the dimension of the subspace E is $(m - k)$.

Corollary 3.7: The set of covering sequences found from Proposition 3.2 and Theorem 3.3 are distinct.

Proof: This follows from Corollaries 3.4 and 3.5 and Theorem 3.5.

Corollary 3.8: The set of covering sequences found from Proposition 3.2 and Theorem 3.4 are distinct.

Proof: This follows from the definition of indicator (3.3) and the fact that any

(-1,1) valued sequence can not be the indicator of a subspace.

Corollary 3.9: The number of covering sequences that can be calculated from Proposition 3.2 is 2^k, which is equal to the number of elements of the largest subspace constructed by Walsh transform nulls.

Proof: The number of cosets that can be constructed from E is

$2^n / 2^{m-k} = 2^k$. Since every coset indicator is a covering sequence, their total number is $2^k = |E^{\perp}|$, which is the number of elements in $E^{\perp}$.

Remark 3.1: Our relations (3.14), (3.15) and Theorem 3.2 have very different meanings. Theorem 3.2 implies that a covering sequence gives some of the Walsh transform nulls (those which have weights less than or equal to the correlation immunity order), but calculation of covering sequence from these nulls is not given and it is impossible to find covering sequences without the knowledge of all nulls. However (3.14) says that every Walsh transform null implies a covering sequence $\lambda_{\mathbf{a}} = (-1)^{<\mathbf{w},\mathbf{a}>}$ and some of the covering sequences with the property

$\lambda_{\mathbf{a}} = (-1)^{<\mathbf{w},\mathbf{a}>}$ indicates a Walsh transform null **w**. Hence every Walsh transform null frequency can be calculated from covering sequences and some of the covering sequences can be calculated from Walsh transform null frequencies.

3.4. Covering Sequences of Affine Equivalent Boolean Functions

In this section, relations between the covering sequences of affine equivalent Boolean functions are studied. Affine equivalence is defined using the definition in [60]. If there exists a nonsingular binary mxm matrix **A** and $mx1$ vectors **b**, $\mathbf{c} \in GF(2)^m$ and $d \in GF(2)$ such that

$$f(\mathbf{x}) = g(\mathbf{Ax} \oplus \mathbf{b}) \oplus < \mathbf{c}, \mathbf{x} > \oplus d \tag{3.34}$$

then f and g are said to be affine equivalent. Walsh and autocorrelation spectra of affine equivalent Boolean functions are studied in [23, 24]. The following proposition is given in [60] on the Walsh spectra relation of affine equivalent Boolean functions.

Proposition 3.3 [60]: Let $f(\mathbf{x}), g(\mathbf{x})$ be two functions satisfying (3.51). Then for any $\mathbf{w} \in GF(2)^m$, [91],

$$W_g(\mathbf{w}) = (-1)^{d+\mathbf{b}<\mathbf{A}^{-1},(\mathbf{c}+\mathbf{w})>} W_f(<(\mathbf{c}+\mathbf{w}), \mathbf{A}^{-1}>) \tag{3.35}$$

Proposition 3.4 [91]**:** The Walsh spectrum of $f(\mathbf{x})$ at i is equal to the Walsh spectrum of $g(\mathbf{x})$ at **j**, where $\mathbf{j} = \mathbf{c} + \mathbf{iA}^{\mathbf{T}}$. Therefore the distribution of absolute value of Walsh spectra of $f(\mathbf{x})$ is same to that of $g(\mathbf{x})$.

Therefore, the number of Walsh transform null frequencies are same for affine equivalent Boolean functions. This means same number of covering sequences can be found from Walsh nulls. However affine equivalent Boolean functions can have different number of covering sequences. This is because they can have covering sequences other than found from (3.14) and (3.15). Since they have different Walsh nulls the corresponding covering sequnces are different. Here we study the covering sequences of affine equivalent functions in detail. Three important questions are:

Question 1: If f does not have any covering sequence, does g have any covering sequence?

Question 2: Let $\boldsymbol{\lambda}$ be one of the covering sequences of f with level ρ. What is the corresponding covering sequence and its level for g?

Question 3: Are all covering sequences of f and g related?

Let us now investigate these questions in three steps.

Answer 1: Assume f does not have any covering sequence. Thus,

$\sum_{\mathbf{a}\in GF(2)^m} \lambda_{\mathbf{a}} \mathbf{D_a f}$ is not a constant vector. Then,

$$
\begin{aligned}
\sum \lambda'_a \mathbf{D_a g} &= \sum_{\mathbf{a}} \lambda'_{\mathbf{a}} (g(\mathbf{x}) \oplus g(\mathbf{x}\oplus\mathbf{a})) \\
&= \sum \lambda'_{\mathbf{a}} (f(\mathbf{Ax}\oplus\mathbf{b}) \oplus <\mathbf{c},\mathbf{x}> \oplus d \\
&\quad + f(\mathbf{A}(\mathbf{x}\oplus\mathbf{a})\oplus\mathbf{b}) \oplus <\mathbf{c},(\mathbf{x}\oplus\mathbf{a})> \oplus d) \\
&= \sum \lambda'_{\mathbf{a}} (f(\mathbf{Ax}\oplus\mathbf{b}) \oplus f(\mathbf{Ax}\oplus\mathbf{Aa}\oplus\mathbf{b})) \oplus <\mathbf{c},\mathbf{a}> \\
&= \sum_{\mathbf{a}} \lambda'_{\mathbf{a}} (f(\mathbf{y}) \oplus f(\mathbf{y}\oplus\mathbf{Aa})) \oplus <\mathbf{c},\mathbf{a}>) \\
&= \sum_{\mathbf{a}} \lambda'_{\mathbf{a}} D_{Aa} f \oplus \lambda'_{\mathbf{a}} <\mathbf{c},\mathbf{a}>
\end{aligned}
\qquad (3.36)
$$

Since $<\mathbf{c},\mathbf{a}>$ is constant for given $\mathbf{a}$ and $\mathbf{c}$, $\sum \lambda_{\mathbf{a}} \mathbf{D_a g}$ can not be a constant vector. Hence, if f does not have any covering sequence then its affine equivalent function g does not have any covering sequence either.

<u>Answer 2:</u> If f and g are affine equivalent then

$$f(\mathbf{x}) = g(\mathbf{Ax} \oplus \mathbf{b}) \oplus <\mathbf{c}, \mathbf{x}> \oplus d = g'(\mathbf{x}) \oplus <\mathbf{c}, \mathbf{x}> \oplus d \,. \tag{3.37}$$

From the fact that if $\mathbf{B} = \mathbf{Ax} \oplus \mathbf{b}$ with nxn matrix $\mathbf{A}$ and $nx1$ vector $\mathbf{b}$ we have from [8],

$$\mathbf{D_a}(f \circ \mathbf{B}) = \mathbf{D_{Aa}} f \circ \mathbf{B}\,, \tag{3.38}$$

one has for $g \circ \mathbf{B} = g(\mathbf{A}x \oplus \mathbf{b}) = g'(\mathbf{x})$

$$\mathbf{D_a} g' = \mathbf{D_{Aa}} g \circ \mathbf{B}\,. \tag{3.39}$$

Then,

$$\begin{aligned} \mathbf{D_a} f &= \mathbf{D_a} g' \oplus <\mathbf{c}, \mathbf{x}> \oplus d \oplus <\mathbf{c}, (\mathbf{x} \oplus \mathbf{a})> \oplus d \\ &= \mathbf{D_{Aa}} g \circ \mathbf{B} \oplus <\mathbf{c}, \mathbf{a}> \end{aligned} \,. \tag{3.40}$$

Covering sequence relation for f is

$$\sum_{\mathbf{a} \in GF(2)^m} \lambda_{\mathbf{a}} \mathbf{D_a f} = (\rho_f \,.....\, \rho_f) = \boldsymbol{\rho}_{\mathbf{f}}. \tag{3.41}$$

Covering sequence relation for g is

$$\sum_{\mathbf{a} \in GF(2)^m} \lambda'_{\mathbf{a}} \mathbf{D_a g} = (\rho_g, \rho_g, ...\rho_g) \quad = \boldsymbol{\rho}_{\mathbf{g}}. \tag{3.42}$$

(3.41) can also be written as:

$$\sum_{\mathbf{a} \in GF(2)^m} \lambda_{\mathbf{a}} (\mathbf{D_{Aa} g} \circ \mathbf{B} + <\mathbf{c}, \mathbf{a}>) = \boldsymbol{\rho}_{\mathbf{f}}\,. \tag{3.43}$$

Then

$$
\begin{array}{l}
\lambda_{00...01}(\mathbf{D}_{\mathbf{A}(00...01)}\mathbf{g}(00...01))\circ\mathbf{B}+<\mathbf{c},00...01>+\lambda_{00...010}(\mathbf{D}_{\mathbf{A}(00...010)}\mathbf{g}(00...01))\circ\mathbf{B}+.....\\
\lambda_{00...01}(\mathbf{D}_{\mathbf{A}(00...01)}\mathbf{g}(2))\circ\mathbf{B}+<\mathbf{c},00...01>+\lambda_{00...010}(\mathbf{D}_{\mathbf{A}(00...010)}\mathbf{g}(00...010))\circ\mathbf{B}+...\\
.\\
.\\
.\\
\lambda_{00...01}(\mathbf{D}_{\mathbf{A}(00...01)}\mathbf{g}(11...11))\circ\mathbf{B}+<\mathbf{c},00...01>+\lambda_{00...010}(\mathbf{D}_{\mathbf{A}(00...010)}\mathbf{g}(11...11))\circ\mathbf{B}+...
\end{array}
=\boldsymbol{\rho}_{\mathbf{f}}
\tag{3.44}
$$

Here $\mathbf{D_i f}(j)$ is j^{th}-indexed position of the vector $\mathbf{D_a f}$ for $\mathbf{a}$=$\mathbf{i}$. (3.42) is equal to

$$
\begin{array}{l}
\left[\lambda'_{00...01}\mathbf{D}_{00...01}\mathbf{f}(00...01)\right]+\left[\lambda'_{00..10}\mathbf{D}_{00..10}\mathbf{f}(0..01)\right]+....\\
\left[\lambda'_{00...01}\mathbf{D}_{00...01}\mathbf{f}(00...10)\right]+.....................................\\
\vdots\\
\left[\lambda'_{00...01}\mathbf{D}_{00...01}\mathbf{f}(1...11)\right]+....................................
\end{array}
=\boldsymbol{\rho}_{\mathbf{g}}
\tag{3.45}
$$

$$
\begin{array}{l}
\left[\lambda_{00...01}\mathbf{D}_{\mathbf{A}(00...01)}\mathbf{g}(\mathbf{A}(00....01)+\mathbf{b})\right]+\lambda_{00...01}<00...01,\mathbf{c}>+\left[\lambda_{00...10}\mathbf{D}_{\mathbf{A}(00...10)}\mathbf{g}(\mathbf{A}(00...01)+\mathbf{b})\right]\\
+\left[\lambda_{00...01}\mathbf{D}_{\mathbf{A}(00...10)}\mathbf{g}(\mathbf{A}(00...10)+\mathbf{b})\right]+\lambda_{00...01}<00...01,\mathbf{c}>+...........\\
.\\
.\\
.\\
\left[\lambda_{00...01}\mathbf{D}_{\mathbf{A}(11...11)}\mathbf{g}(\mathbf{A}(11...11)+\mathbf{b})\right]+\lambda_{00...01}<00...01,\mathbf{c}>+.........
\end{array}
=\boldsymbol{\rho}_{\mathbf{f}}
\tag{3.46}
$$

Define,

$$
\beta=(\oplus_{\mathbf{a}}\lambda_{\mathbf{a}}<\mathbf{a},\mathbf{c}>)
\tag{3.47}
$$

Then (3.46) can be written as

$$
\begin{array}{l}
\lambda_{00...01}(\mathbf{D}_{\mathbf{A}(00...01)}\mathbf{g}(\mathbf{A}(00...01))+\mathbf{b})+\lambda_{00..10}D_{\mathbf{A}(00..10)}\mathbf{g}(\mathbf{A}(00..10)+\mathbf{b})+\\
\cdots+\beta=\lambda'_{00...01}\mathbf{D}_{00...01}\mathbf{f}(00...01)+\lambda'_{00..10}\mathbf{D}_{00..10}\mathbf{f}(00..10)+\cdots
\end{array}
\tag{3.48}
$$

There are 2^{m-1} such equations. If one can find all $(\lambda'_{\mathbf{a}})_{\mathbf{a}\in GF(2)^m}$ from

$(\lambda_{\mathbf{a}})_{\mathbf{a}\in GF(2)^m}$ then

$$\rho_f = \rho_g \tag{3.49}$$

Hence, at least one of the covering sequences of two affine equivalent functions are related by (3.48) and (3.49).

Answer 3: In Answer-2 it is seen that every covering sequence of affine equivalent Boolean functions are related. However the relation we have found does not show that there is a bijective mapping between covering sequences of f and g. Therefore the numbers of covering sequences of affine equivalent functions do not have to be equal. This is also conformed by Proposition 3.2 of Carlet and Tarannikov, because the largest subspaces of Walsh transform nulls of affine equivalent functions do not have the same size in general.

3.5. Conclusions

In this chapter, we show that some covering sequences of a Boolean function can be obtained using the Walsh transform nulls. We prove that each null frequency of the Walsh transform defines one covering sequence; and if the Boolean function is balanced, each null is associated with two covering sequences. We present a lower bound for the number of covering sequences and confirm that the set of covering sequences that we find from Walsh transform nulls are distinct from those given by Carlet and Mesnager. Relations from a covering sequence to a Walsh transform null frequency are given as (3.14) and (3.15). We have shown that

i- for any m-variable Boolean function f , each nonzero Walsh transform null frequency $\mathbf{w} \in GF(2)^m$ uniquely defines a covering sequence $\boldsymbol{\lambda} \in \{1, -1\}$ with elements $\lambda_{\mathbf{a}} = (-1)^{<\mathbf{w},\mathbf{a}>}$ and for each covering sequence $\boldsymbol{\lambda}$ which can be represented as $\lambda_{\mathbf{a}} = (-1)^{<\mathbf{w},\mathbf{a}>}$, there exists a nonzero Walsh transform null **w**.

ii- for an *m*-variable balanced Boolean function f, each nonzero Walsh transform null frequency $\mathbf{w} \in GF(2)^m$ defines a covering sequence $\boldsymbol{\lambda} \in GF(2)^{2^m}$ with elements $\lambda_{\mathbf{a}} =< \mathbf{w}, \mathbf{a} >$ and for each covering sequence $\boldsymbol{\lambda}$ which can be represented as $\lambda_{\mathbf{a}} =< \mathbf{w}, \mathbf{a} >$, there exists a nonzero Walsh transform null **w**, and

Hence one can obtain some of the (in fact as much as the number of Walsh transform nulls) covering sequences from Walsh transform null frequencies. It is proven that all the covering sequences calculated from Walsh transform null frequencies through equation (3.15) are linearly independent and none of them can be an indicator of a subspace. From this point, we come to the conclusion that, the set of covering sequences which can be calculated from Proposition 3.2 of Carlet and Mesnager and Theorem 3.3 [39] are distinct, i.e., our theorems 3.3 and 3.4 give a covering sequence for each Walsh transform null frequency and if these nulls form a subspace called $E^{\perp}$, Carlet- Mesnager Proposition 3.2 gives a covering sequence for each coset of *E*.

On ther other hand, we have obtained a relation between covering sequences of affine equivalent functions and have proven that if one of the affine functions does not have any covering sequence then its affine equivalent function does not have any either. Also it is shown that number of covering sequences of affine equivalent Boolean functions does not have to be equal.

CHAPTER 4

k-DOT PRODUCT AND *k*-AFFINE FUNCTIONS OVER Z_8

Relations between error correcting codes and Boolean functions are studied extensively in the litterature [12-16, 26-31, 45, 46, 55, 59, 64, 67, 71-76, 86, 93, 96, 98-108, 110-112]. Since 1990's, coding theory researchers intensively study nonlinear codes [13, 76] that can be transformed into linear codes [26, 67, 74, 103] in other metric spaces via appropriate mappings. Some of the best-known examples of nonlinear binary error-correcting codes that are better than any linear code are the Nordstrom-Robinson [55, 59, 86, 98], Kerdock and Preparata codes [29, 46, 81, 86]. Calderbank et'al [29] showed that, when properly defined, Kerdock and Preparata codes are linear over the ring Z_4; and as Z_4-codes, they are the duals of each other. All these codes are in fact just extended cyclic codes [46, 81]. Tokareva [104-108] used Krotov matrices [72, 73] to generate Z_4-linear codes [12, 14, 15, 45, 59, 71, 93, 99, 112] and from these codes she introduced *k*-affine binary functions which are affine in an alternative sense. From *k*-affine functions, she then defined *k*-bent functions and a special form of dot-product the *k*-dot product.

In this chapter, we examine Tokareva's studies on Z_4-linear codes. We understand and give the origins of *k*-affine functions and *k*-dot product definitions of Tokareva in Section 4.2. Then in Proposition 4.2, we show that the Krotov matrices $\mathbf{A}^{k,(m-2k)}$, which are used to construct Z_4-linear Hadamard like codes, in fact have as columns as the lexicographically ordered codewords of the Z_4-linear (2^m, m) code C. We observe that, from a Z_4-linear (2^m, m) code C of type $4^k 2^{m-2k}$, which consists of k many Z_4 elements and (m-2k) many Z_2 elements, Tokareva defines a Z_4-linear, (2^{2^m}, m+1) code A_m^k. Then as the binary image of this code, she obtains the code A_m^k. Each codeword of A_m^k defines the truth table of a k-affine function, which then leads to the definition of k-dot products. We give Proposition 4.5 in order to describe the rules that quadratic parts of k-affine functions must obey. In Section 4.3, we give examples of the Z_4-linear codes of types $4^0 2^2$, $4^1 2^0$, $4^1 2^1$ and $4^2 2^0$ as to clarify the subject. Finally Section 4.4 contains our contributions on the extension of these definitions to a larger ring, Z_8. We drive a new class of functions, which we call t,k-affine, using linear codes over the ring Z_8. We then give propositions 4.7 to 4.11. Proposition 4.7 gives the properties of the $\mathbf{C}_\mathbf{m}^\mathbf{t,k}$ matrix. Proposition 4.8 shows that for t=0, k-affine and t,k-affine functions are exactly the same which then imply Proposition 4.9 with the proposal that k-dot product and t,k-dot product values are equivalent for t=0. Proposition 4.10 gives the properties, wheras Proposition 4.11 gives the explicit formula of the t,k-dot product. The new class of functions contain all affine functions, some quadratic functions and some cubic functions. Examples of these functions are given at the end of this chapter starting from Z_8-linear codes.

4.1 Z_4-Linear Codes and Krotov Matrices

We will start this section by giving the definition of Z_4-Linear Codes. Then using this definition we give Proposition 4.2 to give the relation of Z_4-Linear Codes and Krotov matrices.

4.1.1 Z_4-Linear Codes

By a quaternary linear code C of length m, a linear block code over Z_4, i.e., an additive subgroup of Z_4^m is meant. A binary code is Z_4-linear if its coordinates can be permuted so that it is the image of a linear code over Z_4. The folowing proposition gives the generator matrices of quaternary linear codes.

Proposition 4.1: [104] Any Z_4-linear code C containing some nonzero codewords is permutation equivalent to a Z_4-linear code with the generator matrix of the form

$$\begin{pmatrix} \mathbf{I_{k_1}} & \mathbf{A} & \mathbf{B} \\ \mathbf{0} & 2\mathbf{I_{k_2}} & \mathbf{D} \end{pmatrix} \tag{4.1}$$

where $\mathbf{I_{k_1}}$ and $\mathbf{I_{k_2}}$ denote the $k_1 x k_1$ and $k_2 x k_2$ identity matrices, respectively, and **A** and **D** are Z_2 matrices and **B** is a Z_4 matrix. Then C is an abelian group of type $4^{k_1} 2^{k_2}$. C contains $2^{2k_1+k_2}$ codewords. C is a free Z_4 module if and only if $k_2 = 0$.

4.1.2. Krotov Matrices

In [73] D.S. Krotov introduced matrices of size $(r_1 + r_2)\, x\, 2^{(2r_1 + r_2)}$ and named them as A^{r_1, r_2}. These matrices consists of lexicographically ordered columns $\mathbf{z}^T$, where **z** runs through $Z_4^{r_1} x Z_2^{r_2}$. For example,

$$\mathbf{A}^{0.0} = [1], \mathbf{A}^{0,1} = \begin{bmatrix} 1\,1 \\ 0\,2 \end{bmatrix}, \mathbf{A}^{1,1} = \begin{bmatrix} 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1 \\ 0\ 0\ 1\ 1\ 2\ 2\ 3\ 3 \\ 0\ 2\ 0\ 2\ 0\ 2\ 0\ 2 \end{bmatrix}. \tag{4.2}$$

Later than Tokareva [104] named these matrices as $\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}$ with $k = r_1$ and $m = 2r_1 + r_2$ with $0 \le k \le (m/2)$. Tokareva used these matrices to define *k*-affine functions and *k*-bentness crireria. We will give the origins of the $\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}$ matrix in order to understand the origins of the *k*-dot product and *k*-affine functions. We will now give Proposition 4.2 for the $\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}$ matrix.

Proposition 4.2: Columns of the $(m-k)x2^m$ Krotov matrix $\mathbf{A}^{k,(m-2k)} = \begin{bmatrix} 1 \cdots 1 \\ \mathbf{G}_{\mathbf{m}}^{\mathbf{k}} \end{bmatrix}$ are the lexicographically ordered codewords generated by

$$\begin{pmatrix} \mathbf{I_k} & \mathbf{0} \\ \mathbf{0} & 2\mathbf{I_{m-2k}} \end{pmatrix}. \tag{4.3}$$

and an extra symbol ‘1’ in the first position.

Proof: We know that by definition $\mathbf{A}^{k,(m-2k)}$ contains lexicographically ordered columns $\mathbf{z}^T$, where **z** runs through $Z_4^k x Z_2^{m-2k}$. Each column of $\mathbf{A}^{k,(m-2k)}$ consists of *k* many Z_4 symbols and (*m*—2*k*) many Z_2 symbols.

Notice that (4.3) is equivalent to (4.1) with the matrices **A=0 B=0, D=0.** Then (4.3) produces codewords containing $k_1=k$ many Z_4 symbols and $k_2=(m\text{-}2k)$ many Z_2 symbols.

Hence Columns of $\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}$ are the lexicographically ordered codewords generated by (4.3).

4.2 Generating a Z_4-linear, $(2^{2^m}, m+1)$ code A_m^k from a Z_4-linear $(2^m, m)$ code C

It is observed that from a Z_4-linear $(2^m, m)$ code C of type $4^k 2^{m-2k}$, which consists of k many Z_4 elements and $(m\text{-}2k)$ many Z_2 elements, Tokareva defines a Z_4-linear, $(2^{2^m}, m+1)$ code A_m^k using the $2^m x 1$ vectors $\mathbf{h^u}$. A code of type $4^k 2^{m-2k}$ contains $(m-k)$ symbols, k of which are from (0, 1, 2, 3) and $(m-2k)$ of which are from (0, 2).

$$\mathbf{h^u} = \boldsymbol{\varphi}_{\mathbf{k}}^{-1}(\mathbf{u})\,\mathbf{G}_{\mathbf{m}}^{\mathbf{k}} \tag{4.4}$$

where

$$\boldsymbol{\varphi}_{\mathbf{k}}(\mathbf{u}',\mathbf{u}'') = (\phi(\mathbf{u}'),\mathbf{u}'') = \mathbf{u} \tag{4.5}$$

with $\mathbf{u}' \in Z_4^k$ and $\mathbf{u}'' \in Z_2{}^{m-2k}$ and ϕ is the Gray map which is defined by,

$$\begin{aligned} \phi : Z_4 &\to Z_2^2 \\ 0 &\to 00 \\ 1 &\to 01\ . \\ 2 &\to 11 \\ 3 &\to 10 \end{aligned} \tag{4.6}$$

So, $\varphi_{\mathbf{k}}^{-1}(\mathbf{u}) = [\mathbf{u}'\ \mathbf{u}'']$ and $\mathbf{h}^{\mathbf{u}} = [\mathbf{u}'\ \mathbf{u}'']\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}$. Each $\mathbf{h}^{\mathbf{u}}$ can be seen as a linear combination of the rows of $\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}$. Then (4.4) can be written as,

$$\mathbf{h}^{\mathbf{u}} = \mathbf{u}' \begin{bmatrix} \mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(0,0)\cdots\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(0,2^m-1) \\ \mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(1,0)\cdots\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(1,2^m-1) \\ \vdots \\ \mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(k,0)\cdots\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(k,2^m-1) \end{bmatrix} + \mathbf{u}'' \begin{bmatrix} \mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(k+1,0)\cdots\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(k+1,2^m-1) \\ \mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(k+2,0)\cdots\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(k+2,2^m-1) \\ \vdots \\ \mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(m-k,0)\cdots\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(m-k,2^m-1) \end{bmatrix},$$

which is also equal to

$$\begin{aligned} \mathbf{h}^{\mathbf{u}} &= \mathbf{u}'(0)\left[\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(0,0)\cdots\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(0,2^m-1)\right] + \cdots \\ &+ \mathbf{u}'(k)\left[\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(k,0)\cdots\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(k,2^m-1)\right] + \\ &+ \mathbf{u}''(0)\left[\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(k+1,0)\cdots\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(k+1,2^m-1)\right] + \cdots \\ &+ \mathbf{u}''(m-2k)\left[\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(m-k,0)\cdots\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(m-k,2^m-1)\right] \end{aligned} \tag{4.7}$$

+ represents addition on Z_4. In (4.7), $\left[\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(i,0)\cdots\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(i,2^m-1)\right]$ represents i^{th} row of $\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}$ and $\left[\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(i,0)\cdots\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}(i,2^m-1)\right] = \left[c_0(i)\, c_1(i)\cdots c_{2^m-1}(i)\right]$ where $c_j(i)$ is the i^{th} symbol of the j^{th} codeword. Thus i^{th} row of $\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}$ is a vector of size $2^m x1$ which contains i^{th} symbols of all the lexicographically ordered codewords of the Z_4-linear code *C*. For example for a Z_4-linear code C of type $4^1 2^{1_1}$, we have

$$\begin{aligned} &c_0 = [0\ 0], c_1 = [0\ 2], c_2 = [1\ 0], c_3 = [1\ 2], \\ &c_4 = [2\ 0], c_5 = [2\ 2], c_6 = [3\ 0], c_7 = [3\ 2], \end{aligned}$$. So,

$$\mathbf{h}^{\mathbf{21}} = [2\ 1] \begin{bmatrix} 0\ 0\ 1\ 1\ 2\ 2\ 3\ 3 \\ 0\ 2\ 0\ 2\ 0\ 2\ 0\ 2 \end{bmatrix} = (02200220).$$

Notice that, since columns of $\mathbf{G_m^k}$ are ordered lexicographically, elements of the vector $\mathbf{h^u}$ are in some kind of order.

Tokareva defined a $2^m \times 2^m$ matrix $\mathbf{C_m^k} = (c_{u,v}{}^k)$ over Z_4 with the rows $\mathbf{h^u}$. Its rows are in lexicographical order of vectors $\boldsymbol{\varphi}_\mathbf{k}^{-1}\mathbf{(u)}$. Thus $\mathbf{C_m^k}$ has all linear combinations of the symbols of all the codewords of C (Z_4-linear code of type $4^k 2^{m-2k}$). This new code of size 2^{m+1} is also a Z_4-linear code. Its of type 4^{2^m}. The linearity of the new code comes from the fact that the new code contains $\mathbf{h^u}$ for $\forall \mathbf{u} \in Z_2^m$ i.e., all linear combinations of the codeword symbols are in the new code. Thus Z_4-linear code of type $4^k 2^{m-2k}$ is extended to the Z_4-linear code of type 4^{2^m} by Tokareva. A_m^k which contains all $\mathbf{h^u}$ and $\mathbf{h^u}+\mathbf{2}$ is an affine code (+2 complements the corresponding binary vector after mapping by β). Mapping this code to Z_2 by β $\mathrm{A_m^k}$ binary code is obtained.

$$\begin{aligned} \beta : Z_4 &\rightarrow Z_2 \\ 0,1 &\rightarrow 0 \\ 2,3 &\rightarrow 1 \end{aligned} \tag{4.8}$$

As an ilustration for more understanding; we consider the code C as an $(m-k)\times(m-k)$, S-box. Then each codeword of C will be an S-box output. This S-box consists of $(m-k)$ component functions (symbols of the codwords). Each row of $\mathbf{G_m^k}$ then corresponds to the truth table of one component function. Hence each $\mathbf{h^u}$ is a the truth table of a linear combination of the component functions of the S-box which is the so called extended output function of the S-box, which is defined in (2.7). Then $\mathbf{C_m^k}$ contains the truth tables of all of the extended outputs of the S-box as its rows.

Hence the codewords of Z_4-linear code A_m^k are the truth tables of the extended output function of the S-Box (or the code *C*). Table 4.1 summarizes our illustration.

Table 4.1: Summary of our illustration between Tokareva's notations and S-boxes

C	$(m-k)\times(m-k)$ S-box
Symbols of *C*	$(m-k)$ component functions of the S-box
Rows of $\mathbf{G_m^k}$	Truth table of one component function
$\mathbf{h^u}$	Truth table of a extended output function of the S-box
Codewords of Z_4-linear code A_m^k	Truth tables of the extended output functions of the S-box
Codewords of binary code $\mathrm{A_m^k}$	Binary image of the truth tables of the extended output functions of the S-box

4.3 *k*-dot Product and *k*-Affine Functions

To every codeword of the binary code $\mathrm{A_m^k}$ a truth table of a Boolean function can be matched. Codewords of the binary code $\mathrm{A_m^k}$ are illustrated as binary images of the truth tables of the extended output functions of the S-box (Z_4-linear code *C*) in Table 4.1. The corresponding Boolean functions are said to be *k*-affine by Tokareva [104]. Thus the extended output functions of the Z_4-linear code *C* are said to be k-affine. The set of all k-affine functions is denoted by ψ_m^k. For k=0,1 k-affine functions corresponds to affine functions. However for $k \geq 2$ some of the k-affine functions are affine and rest are quadratic.

Proposition 4.3: [108] For any integer m, $0 \leq k \leq m/2$, the class ψ_m^k consists of $2^{m-k+1}(k+1)$ many affine functions and $2^{m-k+1}(2^k - k - 1)$ many quadratic functions.

Corollary 4.1: [108] The part of affine functions in the class $\psi_m^{m/2}$ tends to zero as m grows up.

If g be the Boolean function corresponding to $\boldsymbol{\beta}(\mathbf{h}^{\mathbf{u}})$ which is the vector $\mathbf{h}^{\mathbf{u}}$ whose elements are $\boldsymbol{\beta}$ mapped to Z_2, then Theorem 4.1 gives g.

Theorem 4.1: [104] For integer m, k such that $0 \leq k \leq (m/2)$, a k-affine function with variable $\mathbf{v}$ and constant parameter $\mathbf{u}$ can be written as,

$$\begin{aligned} g(\mathbf{v}) = &\left(\oplus_{i=1}^{k} \oplus_{j=i}^{k} (u_{2i-1} \oplus u_{2i})(u_{2j-1} \oplus u_{2j})(v_{2i-1} \oplus v_{2i})(v_{2j-1} \oplus v_{2j})\right) \oplus \\ &\left(\oplus_{s=1}^{m} u_s v_s\right) \oplus a \end{aligned} \tag{4.9}$$

where $\mathbf{u} \in Z_2^m$ and $a \in Z_2$. For instance, any 2-affine 4-variable function g is uniquely determined by a binary vector $\mathbf{u} = (u_4\ u_3\ u_2\ u_1)$ and an element $a \in Z_2$ as,

$$\begin{aligned} g(v_4\ v_3\ v_2\ v_1) = &(u_1 \oplus u_2)(u_3 \oplus u_4)(v_1 v_3 \oplus v_1 v_4 \oplus v_2 v_3 \oplus v_2 v_4) \oplus \\ &u_2 v_1 \oplus u_1 v_2 \oplus u_4 v_3 \oplus u_3 v_4 \oplus a \end{aligned}.$$

The class ψ_4^2 consists of 24 affine and 8 quadratic functions. Quadratic functions can be given by the vectors $\mathbf{u} \in \{(0101), (0110), (1001), (1010)\}$ and $a \in \{0,1\}$.

Definition 4.1: [104] *k*-dot product of the two *m*-bit binary vectors **u** and **v** is defined to be,

$$\begin{aligned}\langle \mathbf{u}, \mathbf{v} \rangle_k &= \beta(c_{\mathbf{u},\mathbf{v}}^k) \\ &= \left(\oplus_{i=1}^k \oplus_{j=i}^k (u_{2i-1} \oplus u_{2i})(u_{2j-1} \oplus u_{2j})(v_{2i-1} \oplus v_{2i})(v_{2j-1} \oplus v_{2j}) \right) \\ &\quad \oplus \left(\oplus_{s=1}^m u_s v_s \right)\end{aligned} \tag{4.10}$$

Hence *k*-dot product definition comes from the *k*-affine function.

Proposition 4.4: [104] For any integer *n*, *m*, *k* such that $n = 2^m$, $0 \le k \le m/2$, it holds;

(i) $\mathbf{C_{m+1}^k} = \left(\mathbf{C_m^k} \otimes \mathbf{J_2}\right) \oplus (\mathbf{J_n} \otimes \mathbf{C_1^0})$

(ii) $\mathbf{C_{m+2}^{k+1}} = \left(\mathbf{J_4} \otimes \mathbf{C_m^k}\right) \oplus (\mathbf{C_2^1} \otimes \mathbf{J_n})$

(iii) $\left(\mathbf{C_m^k}\right)^T = \mathbf{C_m^k}$

A_m^k is a code, which contains the truth tables of *k*-affine functions, i.e.,

$\mathsf{A}_m^k = \{codewords \,|\, codewords = <\mathbf{u}, \mathbf{v}>_k \oplus a\}$. The algebrac normal forms (ANF) of *k*-affine functions contain a linear part and/or a quadratic part. However it is observed that only a certain type of quadratic terms are included. We define these quadratic terms in Proposition 4.5.

Proposition 4.5: Only the quadratic tems obeying (i), (ii), and (iii) can be included in the algebraic normal forms of the *k*-affine functions. Let us pair the *m*-bit binary vector **v** as $G(v) = \{(v_1, v_2), (v_3, v_4), ...(v_{m-1}, v_m)\}$.

(i) Quadratic part can not contain any product of bits from the same pair, i.e., no product term like $v_1 v_2$ can be included.

(ii) Quadratic part can only contain products of bits from the first *k*-pairs i.e., $v_1 v_3, v_1 v_4 \cdots v_{2k-2} v_{2k}$ can be included.

(iii) Quadratic part is nonzero if and only if only one of the bits in the same pair of the coefficient vector **u** is nonzero,i.e., $u_1 = 1 \Rightarrow u_2 = 0$.

Proof: From the definition of *k*-dot product, it is seen that quadratic part can only result from $Y_i Y_j$ terms with $i \neq j$. All (i), (ii) and (iii) comes from the definition of Y_i.

(iii) part of Proposition 4.5 explains the reason that the class ψ_4^2 consists of 24 affine and 8 quadratic functions. Quadratic functions can be given by the vectors $\mathbf{u} \in \{(0101), (0110), (1001), (1010)\}$ which obey (iii) and $a \in \{0,1\}$. We will now make some examples in order to understand *k*-affine functions.

Example 4.1: Let us begin with the Z_4-linear code of type $4^0 2^2$. This code contains 2 binary symbols and no Z_4 symbol. Now if we write all possible 2-bit binary vectors, we get (00),(01),(10),(11). Columns of G_2^0 consists of 2 times these 2-bit binary vectors, $G_2^0 = \begin{pmatrix} 0\,0\,2\,2 \\ 0\,2\,0\,2 \end{pmatrix}$. For $\mathbf{u} \in \{(00), (01), (10), (11)\}$ we have

$$\mathbf{C_2^0} = \begin{pmatrix} \mathbf{h^{00}} \\ \mathbf{h^{01}} \\ \mathbf{h^{10}} \\ \mathbf{h^{11}} \end{pmatrix} = \begin{pmatrix} 0\,0\,0\,0 \\ 0\,2\,0\,2 \\ 0\,0\,2\,2 \\ 0\,2\,2\,0 \end{pmatrix} \text{ and } \beta(\mathbf{C_2^0}) = \begin{pmatrix} \beta(\mathbf{h^{00}}) \\ \beta(\mathbf{h^{01}}) \\ \beta(\mathbf{h^{10}}) \\ \beta(\mathbf{h^{11}}) \end{pmatrix} = \begin{pmatrix} 0\,0\,0\,0 \\ 0\,1\,0\,1 \\ 0\,0\,1\,1 \\ 0\,1\,1\,0 \end{pmatrix}.$$

The binary code $\mathbf{A}_2^0$ consists of codewords which are the rows of $\beta(\mathbf{C_2^0})$ and their complements. Binary Boolean functions corresponding to these codeword vectors are,

$g(v_2, v_1) = 0 = \langle (00), (v_2, v_1) \rangle = l_{00}$ for the first row of $\beta(\mathbf{C_2^0})$

$g(v_2, v_1) = v_1 = \langle (01), (v_2, v_1) \rangle = l_{01}$ for the second row of $\beta(\mathbf{C_2^0})$

$g(v_2, v_1) = v_2 = \langle (10), (v_2, v_1) \rangle = l_{10}$ for the third row of $\beta(\mathbf{C_2^0})$

$g(v_2, v_1) = v_1 \oplus v_2 = \langle (11), (v_2, v_1) \rangle = l_{11}$ for the fourth row of $\beta(\mathbf{C_2^0})$

From the above equations one gets, $\beta(\mathbf{h^u}) = \langle \mathbf{u}, \mathbf{v} \rangle = l_{\mathbf{u}}$ where $< \mathbf{u}, \mathbf{v} >$ represents the dot product of the vectors **u** and **v** and $l_{\mathbf{u}}(\mathbf{v}) =< \mathbf{u}, \mathbf{v} >$ is the linear function of **v**. Notice that every row of $\beta(\mathbf{C_2^0})$ is the truth table of a linear function of **v**. The complement functions corresponding to $(\mathbf{h^u} + 2)$ are then affine functions of **v**. Thus the binary code $\mathbf{A}_2^0$ contains affine functions. Table 4.2 shows the illustration for this example.

Table 4.2: Illustration for Example 4.1.

C	2×2 S-box with 2-bit binary outputs multiplied by 2, each is a linear mapping
Symbols of C	Component functions of the S-box, a binary linear function
Rows of $\mathbf{G}_{\mathbf{m}}^{\mathbf{k}}$	Truth table of one component function, truth table of a linear function
$\mathbf{h}^{\mathbf{u}}$	Truth table of one extended output function of the S-box, linear combination of linear functions.
Codewords of Z_4-linear code A_m^k	Truth tables of one extended output function of the S-box, linear combination of linear functions.
Codewords of binary code $\mathrm{A}_{\mathrm{m}}^{\mathrm{k}}$	Truth tables of linear combination of linear functions divided by 2. This gives linear function truth tables.

Thus beginning from a binary linear code of size 2^m, 2^m affine functions are obtained. Since no Z_4 term was included in the forming code C, Tokareva called the resultant functions 0-affine.

Example 4.2: Let us now begin with the Z_4-linear code of type $4^1 2^0$. This code contains one Z_4 symbol and no binary symbols. Now if we write all possible Z_4 symbols, we get (0),(1),(2),(3). Columns of $\mathbf{G}_{\mathbf{2}}^{\mathbf{1}}$ consists of these symbols, $\mathbf{G}_{\mathbf{2}}^{\mathbf{1}} = (0\,1\,2\,3)$. For $\phi_k^{-1}(\mathbf{u}) \in \{0, 1, 2, 3\}$, $\mathbf{u} \in \{(00), (01), (11), (10)\}$,. Then

$$\mathbf{C}_{\mathbf{2}}^{\mathbf{1}} = \begin{pmatrix} \mathbf{h}^{\mathbf{00}} \\ \mathbf{h}^{\mathbf{01}} \\ \mathbf{h}^{\mathbf{11}} \\ \mathbf{h}^{\mathbf{10}} \end{pmatrix} = \begin{pmatrix} 0\,0\,0\,0 \\ 0\,1\,2\,3 \\ 0\,2\,0\,2 \\ 0\,3\,2\,1 \end{pmatrix} \text{ and } \beta(\mathbf{C}_{\mathbf{2}}^{\mathbf{1}}) = \begin{pmatrix} 0\,0\,0\,0 \\ 0\,0\,1\,1 \\ 0\,1\,0\,1 \\ 0\,1\,1\,0 \end{pmatrix}.$$

The binary code $\mathbf{A}_2^1$ consists of codewords which are the rows of $\beta(\mathbf{C}_{\mathbf{2}}^{\mathbf{1}})$ and their complements. Binary Boolean functions corresponding to these codeword vectors are,

$g(v_2, v_1) = 0 = \langle (00), (v_2, v_1) \rangle = l_{00}$ for the first row of $\beta(\mathbf{C}_{\mathbf{2}}^{\mathbf{1}})$

$g(v_2, v_1) = v_2 = \langle (10), (v_2, v_1) \rangle = l_{10}$ for the second row of $\beta(\mathbf{C}_{\mathbf{2}}^{\mathbf{1}})$

$g(v_2, v_1) = v_1 = \langle (01), (v_2, v_1) \rangle = l_{01}$ for the third row of $\beta(\mathbf{C}_{\mathbf{2}}^{\mathbf{1}})$

$g(v_2, v_1) = v_1 \oplus v_2 = \langle (11), (v_2, v_1) \rangle = l_{11}$ for the fourth row of $\beta(\mathbf{C}_{\mathbf{2}}^{\mathbf{1}})$

From the above equations one gets, $\beta(\mathbf{h}^{\mathbf{u}}) = \langle \mathbf{u}, \mathbf{v} \rangle_1 = l_{\hat{\mathbf{u}}}$ where $\langle \mathbf{u}, \mathbf{v} \rangle_1$ represents 1-dot product of the vectors **u** and **v** which was defined by Tokeareva [104]. Notice that every row is the truth table of a linear function of **v**. The complement functions corresponding to ($\mathbf{h}^{\mathbf{u}} + 2$) are then affine functions of **v**. Thus the binary code $\mathbf{A}_2^1$ contains affine functions. Since one Z_4 term was included in the forming code *C*. Tokareva called the obtained functions 1-affine which are also affine. Thus only one Z_4 term in the codewords of the forming code *C*, does not destroy the affine property of resultant functions.

Example 4.3: Let us now begin with the Z_4-linear code of type $4^1 2^1$. This code contains one Z_4 symbol and one binary symbol. Now if we write all possible Z_4 symbols, we get (0),(1),(2),(3), and all possible 1-bit binary vectors, we get (0),(1). Columns of $\mathbf{G}_3^1$ consists of one Z_4 symbol and twice the Z_2 symbol, $\mathbf{G}_3^1 = \begin{pmatrix} 0\,0\,1\,1\,2\,2\,3\,3 \\ 0\,2\,0\,2\,0\,2\,0\,2 \end{pmatrix}$. For $\phi_k^{-1}(\mathbf{u}) \in \{00, 01, 10, 11, 20, 21, 30, 31\}$,

$\mathbf{u} \in \{(000), (001), (010), (011), (110), (111), (100), (101)\}$,. Then

$$\mathbf{C}_\mathbf{3}^\mathbf{1} = \begin{pmatrix} \mathbf{h}^{\mathbf{000}} \\ \mathbf{h}^{\mathbf{001}} \\ \mathbf{h}^{\mathbf{010}} \\ \mathbf{h}^{\mathbf{011}} \\ \mathbf{h}^{\mathbf{110}} \\ \mathbf{h}^{\mathbf{111}} \\ \mathbf{h}^{\mathbf{100}} \\ \mathbf{h}^{\mathbf{101}} \end{pmatrix} = \begin{pmatrix} 0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,2\,0\,2\,0\,2\,0\,2 \\ 0\,0\,1\,1\,2\,2\,3\,3 \\ 0\,2\,1\,3\,2\,0\,3\,1 \\ 0\,0\,2\,2\,0\,0\,2\,2 \\ 0\,2\,2\,0\,0\,2\,2\,0 \\ 0\,0\,3\,3\,2\,2\,1\,1 \\ 0\,2\,3\,1\,2\,0\,1\,3 \end{pmatrix} \text{ and } \beta(\mathbf{C}_\mathbf{3}^\mathbf{1}) = \begin{pmatrix} \beta(\mathbf{h}^{\mathbf{000}}) \\ \beta(\mathbf{h}^{\mathbf{001}}) \\ \beta(\mathbf{h}^{\mathbf{010}}) \\ \beta(\mathbf{h}^{\mathbf{011}}) \\ \beta(\mathbf{h}^{\mathbf{110}}) \\ \beta(\mathbf{h}^{\mathbf{111}}) \\ \beta(\mathbf{h}^{\mathbf{100}}) \\ \beta(\mathbf{h}^{\mathbf{101}}) \end{pmatrix} = \begin{pmatrix} 0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,1\,0\,1\,0\,1\,0\,1 \\ 0\,0\,0\,0\,1\,1\,1\,1 \\ 0\,1\,0\,1\,1\,0\,1\,0 \\ 0\,0\,1\,1\,0\,0\,1\,1 \\ 0\,1\,1\,0\,0\,1\,1\,0 \\ 0\,0\,1\,1\,1\,1\,0\,0 \\ 0\,1\,1\,0\,1\,0\,0\,1 \end{pmatrix}.$$

The binary code $\mathbf{A}_3^1$ consists of codewords which are the rows of $\beta(\mathbf{C}_3^\mathbf{1})$ and their complements. Binary Boolean functions corresponding to these codeword vectors are,

$g(v_{3,}\ v_2, v_1) = 0 = \langle (000), (v_{3,}\ v_2, v_1) \rangle_1 = l_{000}$ for the first row of $\beta(\mathbf{C}_3^\mathbf{1})$

$g(v_{3,}\ v_2, v_1) = v_1 = \langle (010), (v_{3,}\ v_2, v_1) \rangle_1 = l_{001}$ for the second row of $\beta(\mathbf{C}_3^\mathbf{1})$

$g(v_{3,}\ v_2, v_1) = v_3 = \langle (100), (v_{3,}\ v_2, v_1) \rangle_1 = l_{100}$ for the third row of $\beta(\mathbf{C}_3^{\mathbf{1}})$

$g(v_{3,}\ v_2, v_1) = v_3 \oplus v_1 = \langle (110), (v_{3,}\ v_2, v_1) \rangle_1 = l_{101}$ for the fourth row,

$g(v_{3,}\ v_2, v_1) = v_2 = \langle (001), (v_{3,}\ v_2, v_1) \rangle_1 = l_{010}$ for the fifth row,

$g(v_{3,}\ v_2, v_1) = v_2 \oplus v_1 = \langle (011), (v_{3,}\ v_2, v_1) \rangle_1 = l_{011}$ for the sixth row,

$g(v_{3,}\ v_2, v_1) = v_2 \oplus v_3 = \langle (101), (v_{3,}\ v_2, v_1) \rangle_1 = l_{110}$ for the seventh row, and

$g(v_{3,}\ v_2, v_1) = v_3 \oplus v_2 \oplus v_1 = \langle (111), (v_{3,}\ v_2, v_1) \rangle_1 = l_{111}$ for the last row.

From the above equations one gets, $\beta(\mathbf{h}^{\mathbf{u}}) = \langle \mathbf{u}''', \mathbf{v} \rangle_1 = l_{\hat{\mathbf{u}}}$ where $\langle \mathbf{u}, \mathbf{v} \rangle_1$ represents 1-dot product of the vectors **u** and **v** which was defined by Tokareva [104]. Notice that every row is the truth table of a linear function of **v**. The complement functions corresponding to $(\mathbf{h}^{\mathbf{u}} + 2)$ are then affine functions of **v**. Thus the binary code $\mathbf{A}_3^1$ contains affine functions. Tokareva called the obtained functions 1-affine which are also affine.

Example 4.4: Let us now begin with the Z_4-linear code of type $4^2 2^0$. This code contains two Z_4 symbols and no binary symbols. Now if we write all possible 2-symbol Z_4 vectors, we get (00), (01), (02), (03), (10), (11), (12), (13), (20),(21),(22),(23), (30),(31),(32),(33). Columns of $\mathbf{G}_4^2$ consists of two Z_4 symbols, $\mathbf{G}_4^2 = \begin{pmatrix} 0\,0\,0\,0\,1\,1\,1\,1\,2\,2\,2\,2\,3\,3\,3\,3 \\ 0\,1\,2\,3\,0\,1\,2\,3\,0\,1\,2\,3\,0\,1\,2\,3 \end{pmatrix}$. Then,

$$
\mathbf{C}_{\mathbf{4}}^{\mathbf{2}} = \begin{pmatrix} \mathbf{h}^{0000} \\ \mathbf{h}^{0001} \\ \mathbf{h}^{0011} \\ \mathbf{h}^{0010} \\ \mathbf{h}^{0100} \\ \mathbf{h}^{0101} \\ \mathbf{h}^{0111} \\ \mathbf{h}^{0110} \\ \mathbf{h}^{1100} \\ \mathbf{h}^{1101} \\ \mathbf{h}^{1111} \\ \mathbf{h}^{1110} \\ \mathbf{h}^{1000} \\ \mathbf{h}^{1001} \\ \mathbf{h}^{1011} \\ \mathbf{h}^{1010} \end{pmatrix} = \begin{pmatrix}
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\
0&1&2&3&0&1&2&3&0&1&2&3&0&1&2&3 \\
0&2&0&2&0&2&0&2&0&2&0&2&0&2&0&2 \\
0&3&2&1&0&3&2&1&0&3&2&1&0&3&2&1 \\
0&0&0&0&1&1&1&1&2&2&2&2&3&3&3&3 \\
0&1&2&3&1&2&3&0&2&3&0&1&3&0&1&2 \\
0&2&0&2&1&3&1&3&2&0&2&0&3&1&3&1 \\
0&3&2&1&1&0&3&2&2&1&0&3&3&2&1&0 \\
0&0&0&0&2&2&2&2&0&0&0&0&2&2&2&2 \\
0&1&2&3&2&3&0&1&0&1&2&3&2&3&0&1 \\
0&2&0&2&2&0&2&0&0&2&0&2&2&0&2&0 \\
0&3&2&1&2&1&0&3&0&3&2&1&2&1&0&3 \\
0&0&0&0&3&3&3&3&2&2&2&2&1&1&1&1 \\
0&1&2&3&3&0&2&1&2&3&0&1&1&2&3&0 \\
0&2&0&2&3&1&3&1&2&0&2&0&1&3&1&3 \\
0&3&2&1&3&2&1&0&2&1&0&3&1&0&3&2
\end{pmatrix} \text{ and}
$$

$$
\beta(\mathbf{C}_4^2) = \begin{pmatrix}
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \\
0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1 \\
0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1 \\
0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0 \\
0\,0\,0\,0\,0\,0\,0\,0\,1\,1\,1\,1\,1\,1\,1\,1 \\
0\,0\,1\,1\,0\,1\,1\,0\,1\,1\,0\,0\,1\,0\,0\,1 \\
0\,1\,0\,1\,0\,1\,0\,1\,1\,0\,1\,0\,1\,0\,1\,0 \\
0\,1\,1\,0\,0\,0\,1\,1\,1\,0\,0\,1\,1\,1\,0\,0 \\
0\,0\,0\,0\,1\,1\,1\,1\,0\,0\,0\,0\,1\,1\,1\,1 \\
0\,0\,1\,1\,1\,1\,0\,0\,0\,0\,1\,1\,1\,1\,0\,0 \\
0\,1\,0\,1\,1\,0\,1\,0\,0\,1\,0\,1\,1\,0\,1\,0 \\
0\,1\,1\,0\,1\,0\,0\,1\,0\,1\,1\,0\,1\,0\,0\,1 \\
0\,0\,0\,0\,1\,1\,1\,1\,1\,1\,1\,1\,0\,0\,0\,0 \\
0\,0\,1\,1\,1\,0\,1\,0\,1\,1\,0\,0\,0\,1\,1\,0 \\
0\,1\,0\,1\,1\,0\,1\,0\,1\,0\,1\,0\,0\,1\,0\,1 \\
0\,1\,1\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0\,1\,1
\end{pmatrix}.
$$

The binary code $\mathbf{A}_4^2$ consists of codewords which are the rows of $\beta(\mathbf{C}_4^2)$ and their complements. The binary Boolean function corresponding to these codeword vectors are,

$g(v_4, v_3, v_2, v_1) = 0 = \langle (0000), (v_4, v_3, v_2, v_1) \rangle_2 = l_{0000}$ for the first row,

$g(v_4, v_3, v_2, v_1) = v_2 = \langle (0001), (v_4, v_3, v_2, v_1) \rangle_2 = l_{0010}$ for the second row,

$g(v_4, v_3, v_2, v_1) = v_1 = \langle (0010), (v_4, v_3, v_2, v_1) \rangle_2 = l_{0001}$ for the third row,

$g(v_4, v_3, v_2, v_1) = v_1 \oplus v_2 = \langle (0011), (v_4, v_3, v_2, v_1) \rangle_2 = l_{0011}$ for the fourth,

$g(v_4, v_3, v_2, v_1) = v_1 v_4 \oplus v_2 \oplus v_3 = \langle (1001), (v_4, v_3, v_2, v_1) \rangle_2$ for the tenth, and

$g(v_4, v_3, v_2, v_1) = v_2 v_4 \oplus v_1 \oplus v_3 = \langle (1010), (v_4, v_3, v_2, v_1) \rangle_2$ for the 11th row.

Other rows can be similary shown to satisfy (4.10). From Example 4.4 it is seen that Boolean functions corresponding to the codewords of the binary code $\mathbf{A}_4^2$ both contain a linear part and a quadratic part. From Proposition 4.5, each function $g(\mathbf{v})$ contains quadratic parts which are the products of first 2 pairs of input vector $\mathbf{v}$. Tokareva called these functions 2-affine since two Z_4 symbols were included in the codewords of the forming code C, and the functions can contain $2k$ many quadratic terms.

4.4 New *t,k*-dot Product and *t,k*-affine Functions Beginning from Z_8-Linear Codes

In previous sections we examined the *k*-dot product and *k*-affine functions, which were defined beginning from Z_4-linear codes. As a summary, we observed that from a $(m, m-k)$ Z_4-linear code C of type $4^k 2^{m-2k}$, which consists of k many Z_4 elements and $(m-2k)$ many Z_2 elements, Tokareva defined a Z_4-linear code A_m^k. Then from this code she obtained a binary $(2^m, m+1)$ code A_m^k. Each codeword of A_m^k then defined a truth table of a *k*-affine function which leaded to *k*-dot products.

Now we will define *t,k*-dot product and *t,k*-affine functions beginning from Z_8-linear codes in a similar way Tokareva defined *k*-dot product and *k*-affine functions from Z_4-linear codes. Our road map is:

I. First of all we will start with a (m, m─k─$2t$) Z_8-linear code C of type $8^t 4^k 2^{m-3t-2k}$, which consists of t many Z_8 elements, k many Z_4 elements and (m─$3t$─$2k$) many Z_2 elements.

II. By writing all codewords lexicographically as columns, we will obtain the matrix $\mathbf{G}_{\mathbf{m}}^{t,k}$.

III. Then we will obtain a (2^m, m+1) Z_8-linear code $A_m^{t,k}$ using $\mathbf{G}_{\mathbf{m}}^{t,k}$ as the generator matrix.

IV. Later then from the code obtained in (III) we will produce a binary (2^m, m+1) code A_m^k using the map π which will be defined in (4.14).

V. Each codeword of A_m^k then defines a truth table of a t,k-affine function as the Definition 4.8, which leads to t,k-dot products whose explicit formula is given in Proposition 4.11.

Before using the above road map we will first give some definitions for Z_{2^s}-linear codes given by Carlet [45].

Definition 4.4: [45] Let k be any positive integer, **u** any element of Z_{2^s} and $\sum_{i=1}^{s} 2^{i-1} u_i$ its binary expansion ($u_i = 0, 1$).The image of **u** by the generalized Gray map is the following Boolean function on

$$GF(2)^{i-1}, G(u) : (y_1 \cdots y_{s-1}) \rightarrow u_s + \sum_{i=1}^{s-1} u_i y_i \ .$$

The generalized Gray map is a mapping from Z_{2^s} onto the Reed–Muller code of order 1, R(1; k−1). When k = 2, R (1; 1) being equal to the set of all the Boolean functions on $GF(2)$, we obtain the usual Gray map, which is a mapping from Z_4 to $GF(2)^2$. For instance, when k = 3, the images of the elements of Z_8 are the following words of length 4: G(0) =(0; 0; 0; 0); G(1) = (0; 1; 0; 1); G(2) = (0; 0; 1; 1); G(3) =(0; 1; 1; 0);G(4) = (1; 1; 1; 1); G(5) = (1; 0; 1; 0); G(6) =(1; 1; 0; 0); G(7) = (1; 0; 0; 1).

Definition 4.5: [45] A binary code is called Z_{2^s} -linear if its coordinates can be arranged so that it is the image of a linear Z_{2^s} -ary code by the generalized Gray map.

Now we will define k-dot product and k-bentness criteria beginning from Z_8-linear codes. First of all we will give the mapping table between Z_8 and Z_4 and Z_2 as Table 4.3.

Table 4.3: Generalized Gray mapping between Z_8 and Z_4 and Z_2 symbols

Z_8	Generalized Gray map	Z_4	Gray map	Z_2
0	0000	0	00	0
1	0101	0	01	0
2	0011	1	00	0
3	0110	1	01	0
4	1111	2	11	1
5	1010	2	10	1
6	1100	3	11	1
7	1001	3	10	1

The main quality of the Gray map is that, it is distance preserving. However there does not exist a distance preserving mapping from Z_8 [45], to Z_2^3. The Gray map preserves distances, i.e.,

$$d_L(\mathbf{x},\mathbf{y}) = d(\phi(\mathbf{x}), \phi(\mathbf{y})) \tag{4.11}$$

for all $\mathbf{x},\mathbf{y} \in Z_4^n$. Here $d_L(\mathbf{x},\mathbf{y})$ is the Lee distance of two Z_4 vectors which is defined as [27],

$$d_L(\mathbf{x},\mathbf{y}) = w_L(\mathbf{x}-\mathbf{y}) \tag{4.12}$$

where $w_L(\mathbf{x})$ is the Lee weight of the Z_4 vector $\mathbf{x} = (x_1,\cdots,x_m)$.

$$w_L(\mathbf{x}) = \sum_{i=1}^{m} w_L(x_i) \quad (4.13)$$

with $w_L(0) = 0,\ w_L(1) = 1,\ w_L(2) = 2,\ w_L(3) = 1$.

Carlet uses the generalized Gray map as it is a distance preserving map. However representation of Z_8 ring elements by 4 bit is redundant. We will use an alternative map which uses 3-bit representation but not distance invariant. Table 4.4 gives our map, θ which is given by,

$$\begin{aligned} &\theta : Z_8 \to Z_2^3 \\ &\quad 0 \to 000, 1 \to 010,\ 2 \to 001,\ \ 3 \to 011. \\ &\quad 4 \to 111, 5 \to 101\ 6 \to 110,\ \ 7 \to 100 \end{aligned} \quad (4.14)$$

We construct Table 4.4 from the knowledge that if **M** is a Z_2 matrix then 4**M** is a proper Z_8 matrix. Then binary symbols are multiplied by 4, i.e., $0 \to 0, 1 \to 4$. Similarly if **N** is a Z_4 matrix then 2**N** is a proper Z_8 matrix. Then Z_4 symbols are multipled by 2, i.e., $0 \to 0, 1 \to 2, 2 \to 4, 3 \to 6$.

The map $\beta : Z_4 \to Z_2$ was given in (5.8) and is a part of Gray map. We define the map $\pi : Z_8 \to Z_2$ according to Table 4.4 as,

$$\begin{aligned} \pi &: 0, 1, 2, 3 \to 0 \\ &\ \ 4, 5, 6, 7 \to 1 \end{aligned} . \quad (4.15)$$

Table 4.4: Our mapping between Z_8 and Z_4 and Z_2 symbols

Z_8	Our map, θ	Z_4	Gray map	Z_2
0	000	0	00	0
1	010	0	00	0
2	001	1	01	0
3	011	1	01	0
4	111	2	11	1
5	101	2	11	1
6	110	3	10	1
7	100	3	10	1

Now we will return to our road map.

Road map I: The first step is to start with a (m, $m-k-2t$) Z_8-linear code C of type $8^t 4^k 2^{m-3t-2k}$, which consists of t many Z_8 elements, k many Z_4 elements and ($m-3t-2k$) many Z_2 elements. We define the generator matrix for Z_8-linear codes in Definition 4.6.

Definition 4.6: The generator matrices for Z_8-linear codes of type $8^t 4^k 2^{m-3t-2k}$ are equivalent to

$$\begin{pmatrix} \mathbf{I}_{\mathbf{k_1}} & \mathbf{A} & \mathbf{B} \\ 0 & 2\mathbf{I}_{\mathbf{k_2}} & \mathbf{F} \\ \mathbf{0} & 0 & 4\mathbf{I}_{\mathbf{k_3}} \end{pmatrix} \tag{4.16}$$

where $\mathbf{I_{k_1}}$ and $\mathbf{I_{k_2}}$ and $\mathbf{I_{k_3}}$ denote the $k_1 \times k_1$, $k_2 \times k_2$ and $k_3 \times k_3$ identity matrices, respectively, and **A** and **F** are Z_4 matrices and **B** is a Z_8 matrix. Then C contains $2^{3k_1+2k_2+k_3}$ codewords. C is a free Z_8 module if and only if $k_2 = 0$ and $k_3 = 0$.

Road map II: We use (4.16) with $k_1 = t, k_2 = k, k_3 = m - 3t - 2k$, and obtained

$$\begin{pmatrix} \mathbf{I_t} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & 2\mathbf{I_k} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 4\mathbf{I_{m-3t-2k}} \end{pmatrix}. \tag{4.17}$$

Then by writing all codewords lexicographically of this code as columns, we will obtain the matrix $\mathbf{G}_\mathbf{m}^{t,k}$, for $0 \le t \le m/3$ and $0 \le k \le (m-3t)/2$. Notice that $\mathbf{G_m^{t,k}}$ is an extension of the matrix $\mathbf{G_m^k}$ defined by Tokareva. Let us give some examples;

$$\mathbf{G}_2^{0,0} = \begin{pmatrix} 0\,0\,4\,4 \\ 0\,4\,0\,4 \end{pmatrix},\ \mathbf{G}_2^{0,1} = (0\,2\,4\,6),$$

$$\mathbf{G}_3^{0,1} = \begin{pmatrix} 0\,0\,2\,2\,4\,4\,6\,6 \\ 0\,4\,0\,4\,0\,4\,0\,4 \end{pmatrix}, \qquad \mathbf{G}_3^{1,0} = (0\,1\,2\,3\,4\,5\,6\,7),$$

$$\mathbf{G}_5^{1,1} = \begin{pmatrix} 0\,0\,0\,0 & 1\,1\,1\,1 & 2\,2\,2\,2 & 3\,3\,3\,3 & 4\,4\,4\,4 & 5\,5\,5\,5 & 6\,6\,6\,6 & 7\,7\,7\,7 \\ 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 \end{pmatrix}.$$

Road map III: We will obtain a $(2^m, m+1)$ Z_8-linear code $A_m^{t,k}$ using $\mathbf{G_m^{t,k}}$ as the generator matrix. $A_m^{t,k}$ contains as codewords as the vectors $\mathbf{h}_8^\mathbf{u}$ and $\mathbf{h}_8^\mathbf{u} + \mathbf{4}$. We define the $2^m \times 1$ vector $\mathbf{h}_8^\mathbf{u}$ as,

$$\mathbf{h}_8^{\mathbf{u}} = \varphi_{\mathbf{t,k}}^{-1}(\mathbf{u})\, \mathbf{G}_{\mathbf{m}}^{\mathbf{t,k}} \tag{4.18}$$

where

$$\varphi_{\mathbf{t,k}}(\mathbf{u}', \mathbf{u}'', \mathbf{u}''') = (\theta(\mathbf{u}'), \phi(\mathbf{u}''), \mathbf{u}''') = \mathbf{u} \tag{4.19}$$

with $\mathbf{u}' \in Z_8^t$, $\mathbf{u}'' \in Z_4{}^k$ and $\mathbf{u}''' \in Z_2^{m-3t-2k}$. ϕ is the Gray map and we give θ in (4.14). We now define the matrix $\mathbf{C}_{\mathbf{m}}^{\mathbf{t,k}}$, which has rows $\mathbf{h}_8^{\mathbf{u}}$ as an extension to the matrix $\mathbf{C}_{\mathbf{m}}^{\mathbf{k}}$ defined by Tokareva. Then $\mathsf{A}_m^{t,k}$ have codewords as the rows of $\mathbf{C}_{\mathbf{m}}^{\mathbf{t,k}}$ and $\mathbf{C}_{\mathbf{m}}^{\mathbf{t,k}} + 4\mathbf{J}_{\mathbf{2^m}}$. We give the following examples for $\mathbf{C}_{\mathbf{m}}^{\mathbf{t,k}}$ matrices.

$$\mathbf{C}_{\mathbf{2}}^{\mathbf{0,0}} = \begin{pmatrix} 0\,0\,0\,0 \\ 0\,4\,0\,4 \\ 0\,0\,4\,4 \\ 0\,4\,4\,0 \end{pmatrix},\ \mathbf{C}_{\mathbf{2}}^{\mathbf{0,1}} = \begin{pmatrix} 0\,0\,0\,0 \\ 0\,2\,4\,6 \\ 0\,4\,0\,4 \\ 0\,6\,4\,2 \end{pmatrix},$$

$$
\mathbf{C}_{\mathbf{3}}^{\mathbf{0,1}} = \begin{pmatrix} 0&0&0&0&0&0&0&0 \\ 0&4&0&4&0&4&0&4 \\ 0&0&2&2&4&4&6&6 \\ 0&4&2&6&4&0&6&2 \\ 0&0&4&4&0&0&4&4 \\ 0&4&4&0&0&4&4&0 \\ 0&0&6&6&4&4&2&2 \\ 0&4&6&2&4&0&2&6 \end{pmatrix}, \qquad
\mathbf{C}_{3}^{\mathbf{1,0}} = \begin{pmatrix} 0&0&0&0&0&0&0&0 \\ 0&1&2&3&4&5&6&7 \\ 0&2&4&6&0&2&4&6 \\ 0&3&6&1&4&7&2&5 \\ 0&4&0&4&0&4&0&4 \\ 0&5&2&7&4&1&6&3 \\ 0&6&4&2&0&6&4&2 \\ 0&7&6&5&4&3&2&1 \end{pmatrix},
$$

$$
\mathbf{C}_{5}^{\mathbf{1,1}} = \begin{pmatrix} 0\,0\,0\,0 \quad 0\,0\,0\,0 \quad 0\,0\,0\,0 \quad 0\,0\,0\,0 \quad 0\,0\,0\,0 \quad 0\,0\,0\,0 \quad 0\,0\,0\,0 \quad 0\,0\,0\,0 \\ 0\,2\,4\,6 \quad 0\,2\,4\,6 \quad 0\,2\,4\,6 \quad 0\,2\,4\,6 \quad 0\,2\,4\,6 \quad 0\,2\,4\,6 \quad 0\,2\,4\,6 \quad 0\,2\,4\,6 \\ 0\,0\,0\,0 \quad 1\,1\,1\,1 \quad 2\,2\,2\,2 \quad 3\,3\,3\,3 \quad 4\,4\,4\,4 \quad 5\,5\,5\,5 \quad 6\,6\,6\,6 \quad 7\,7\,7\,7 \\ 0\,2\,4\,6 \quad 1\,3\,5\,7 \quad 2\,4\,6\,0 \quad 3\,5\,7\,1 \quad 4\,6\,0\,2 \quad 5\,7\,1\,3 \quad 6\,0\,2\,4 \quad 7\,1\,3\,5 \\ \vdots \end{pmatrix}.
$$

Proposition 4.7: For any integers n, m, t, k such that $n = 2^m$, $0 \le t \le m/3$ and $0 \le k \le (m-3t)/2$, it holds;

(i) $\mathbf{C}_{\mathbf{m+1}}^{\mathbf{t,k}} = \left(\mathbf{C}_{\mathbf{m}}^{\mathbf{t,k}} \otimes \mathbf{J}_{\mathbf{2}}\right) \oplus (\mathbf{J}_{\mathbf{n}} \otimes \mathbf{C}_{\mathbf{1}}^{\mathbf{0,0}})$ (4.20)

(ii) $\mathbf{C}_{\mathbf{m+2}}^{\mathbf{t,k+1}} = \left(\mathbf{J}_{\mathbf{4}} \otimes \mathbf{C}_{\mathbf{m}}^{\mathbf{t,k}}\right) \oplus (\mathbf{C}_{\mathbf{2}}^{\mathbf{0,1}} \otimes \mathbf{J}_{\mathbf{n}})$ (4.21)

(iii) $\left(\mathbf{C}_{\mathbf{m}}^{0,\mathbf{k}}\right)^T = \mathbf{C}_{\mathbf{m}}^{0,\mathbf{k}}$ (4.22)

(iv) $\mathbf{C}_{\mathbf{m+3}}^{\mathbf{t+1,k}} = \left(\mathbf{J}_{\mathbf{8}} \otimes \mathbf{C}_{\mathbf{m}}^{\mathbf{t,k}}\right) \oplus (\mathbf{C}_{3}^{\mathbf{1,0}} \otimes \mathbf{J}_{\mathbf{n}})$ (4.23)

(v) $\left(\mathbf{C}_{\mathbf{m}}^{\mathbf{t,k}}\right)^T = \mathbf{C}_{\mathbf{m}}^{\mathbf{t,k}}$ (4.24)

Proof: (i) Consider $\mathbf{G}_{\mathbf{m}}^{\mathbf{t,k}} = \left(z_1^T, z_2^T, \cdots z_{2^m}^T\right)$, then

$\mathbf{G}_{\mathbf{m}+1}^{\mathbf{t,k}} = \begin{pmatrix} z_1^T, z_1^T, z_2^T, z_2^T, \cdots, z_{2^m}^T, z_{2^m}^T \\ 0 \quad 4 \quad 0 \quad 4 \qquad 0 \quad 4 \end{pmatrix}$ and using the definition of $\boldsymbol{\varphi}_{\mathbf{t,k}}^{-\mathbf{1}}(\mathbf{u})$ and $\mathbf{h}_{\mathbf{8}}^{\mathbf{u}} = [h_1 \cdots h_n]$, we have

$\mathbf{h}_{\mathbf{8}}^{(\mathbf{u},a)} = \left(\boldsymbol{\varphi}_{\mathbf{t,k}}^{-\mathbf{1}}(\mathbf{u}), a\right)\mathbf{G}_{\mathbf{m+1}}^{\mathbf{t,k}} = (h_1, h_1 + 4a, \cdots h_n, h_n + 4a)$ for $a \in GF(2)$.

Thus, in order to obtain the matrix $\mathbf{C}_{\mathbf{m}+\mathbf{1}}^{\mathbf{t,k}}$ we should replace any element $c_{\mathbf{u},\mathbf{v}}^{t,k}$ of $\mathbf{C}_{\mathbf{m}}^{\mathbf{t,k}}$ by the matrix $\begin{pmatrix} c_{\mathbf{u},\mathbf{v}}^{t,k} & c_{\mathbf{u},\mathbf{v}}^{t,k} \\ c_{\mathbf{u},\mathbf{v}}^{t,k} & c_{\mathbf{u},\mathbf{v}}^{t,k} + 4 \end{pmatrix}$. Hence (i) is true.

(ii) and (iii) can be similarly proven as in the proof of Proposition 1 given by Tokareva in [104].

(iv) Consider $\mathbf{G}_{\mathbf{m}}^{\mathbf{t,k}} = \left(z_1^T, z_2^T, \cdots z_{2^m}^T \right)$, then

$\mathbf{G}_{\mathbf{m}+1}^{\mathbf{t}+1,\mathbf{k}} = \begin{pmatrix} 0\cdots0\,1\cdots1\,2\cdots2\,\cdots7\cdots7 \\ \mathbf{G}_{\mathbf{m}}^{\mathbf{t,k}}\ \mathbf{G}_{\mathbf{m}}^{\mathbf{t,k}}\ \mathbf{G}_{\mathbf{m}}^{\mathbf{t,k}} \cdots \mathbf{G}_{\mathbf{m}}^{\mathbf{t,k}} \end{pmatrix}$ and using the definition of $\boldsymbol{\varphi}_{\mathbf{t,k}}^{-\mathbf{1}}(\mathbf{u})$ we have

$\mathbf{h}_{\mathbf{8}}^{(a,b,c,\mathbf{u})} = (\mathbf{h}_{\mathbf{8}}^{\mathbf{u}}, \mathbf{h}_{\mathbf{8}}^{\mathbf{u}} + \delta 1, \cdots, h_8^{2^m} + \delta 7)$ for $\delta = \theta^{-1}(a,b,c)$. (iv) is then true.

(v) comes from (iv) and (i). Proposition 4.7 will be used to derive the explicit expression of the *t,k*-dot product in Propositon 4.11.

Road map IV: The binary image of the code $A_m^{t,k}$ is denoted by $\mathsf{A}_m^{t,k}$. We use the map π, which was defined in (4.14), for this purpose. Then $\mathsf{A}_m^{t,k}$ is a (2^m, m+1) code, which has as codewords as the rows $\pi(\mathbf{C}_{\mathbf{m}}^{\mathbf{t,k}})$ and $\pi(\mathbf{C}_{\mathbf{m}}^{\mathbf{t,k}} + 4\mathbf{J}_{\mathbf{2^m}})$. For instance,

$$\pi(\mathbf{C}_{\mathbf{2}}^{\mathbf{0,0}}) = \begin{pmatrix} 0\,0\,0\,0 \\ 0\,1\,0\,1 \\ 0\,0\,1\,1 \\ 0\,1\,1\,0 \end{pmatrix},\ \pi(\mathbf{C}_{\mathbf{2}}^{\mathbf{0,1}}) = \begin{pmatrix} 0\,0\,0\,0 \\ 0\,0\,1\,1 \\ 0\,1\,0\,1 \\ 0\,1\,1\,0 \end{pmatrix},$$

$$\pi(\mathbf{C}_{\mathbf{3}}^{\mathbf{0,1}}) = \begin{pmatrix} 0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,1\,0\,1\,0\,1\,0\,1 \\ 0\,0\,0\,0\,1\,1\,1\,1 \\ 0\,1\,0\,1\,1\,0\,1\,0 \\ 0\,0\,1\,1\,0\,0\,1\,1 \\ 0\,1\,1\,0\,0\,1\,1\,0 \\ 0\,0\,1\,1\,1\,1\,0\,0 \\ 0\,1\,1\,0\,1\,0\,0\,1 \end{pmatrix},\ \pi(\mathbf{C}_{3}^{\mathbf{1,0}}) = \begin{pmatrix} 0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,1\,1\,1\,1 \\ 0\,0\,1\,1\,0\,0\,1\,1 \\ 0\,0\,1\,0\,1\,1\,0\,1 \\ 0\,1\,0\,1\,0\,1\,0\,1 \\ 0\,1\,0\,1\,1\,0\,1\,0 \\ 0\,1\,1\,0\,0\,1\,1\,0 \\ 0\,1\,1\,1\,1\,0\,0\,0 \end{pmatrix}.$$

Road map V: Each codeword of $\mathsf{A}_m^{t,k}$ corresponds to the truth table of a Boolean function. We call these functions *t*,*k*-affine. We mean the forming Z_8-linear code contains *t* many Z_8 symbols and *k* many Z_4 symbols. Each *t*,*k*-affine Boolean function is in the form of *t*,*k*-dot product as will be given in our Definition 4.7. The set of *t*,*k*-affine functions is denoted by $\psi_m^{t,k}$.

Definition 4.7: (*t*,*k*-dot product):

$$<\mathbf{u},\mathbf{v}>_{t,k} = \pi(c_{\mathbf{u},\mathbf{v}}^{t,k}) \tag{4.25}$$

Proposition 4.8:

$$\beta(\mathbf{C}_m^k) = \pi(\mathbf{C}_m^{\mathbf{0},k}) \tag{4.26}$$

Proof: For t=0, the generator matrix of the Z_8-linear codes given by (4.17) is equivalent to generator matrix of the Z_4-linear codes given by (4.3). Then the matrices $\mathbf{G}_{\mathbf{m}}^{t,k}$ and $\mathbf{G}_{\mathbf{m}}^k$ are equal except that binary symbols are multiplied by 4 in $\mathbf{G}_{\mathbf{m}}^{t,k}$ and by 2 in $\mathbf{G}_{\mathbf{m}}^k$. So β mapping (dividing by 2) of the matrix $\mathbf{C}_{\mathbf{m}}^{\mathbf{k}}$ and So π mapping (dividing by 4) of the matrix $\mathbf{C}_{\mathbf{m}}^{\mathbf{t,k}}$ will be equal. Then

$$\beta(c_{\mathbf{u},\mathbf{v}}^k) = \pi(c_{\mathbf{u},\mathbf{v}}^{0,k}) \tag{4.27}$$

which leads to (4.26).

Proposition 4.9: *t*,*k*- dot product is equal to *k*-dot product for *t*=0, i.e.,

$$<\mathbf{u},\mathbf{v}>_{0,k} = <\mathbf{u},\mathbf{v}>_k \tag{4.28}$$

Proof: Recall equations (4.10), (4.25) and (4.27) as,

$\langle \mathbf{u},\mathbf{v} \rangle_k = \beta(c_{\mathbf{u},\mathbf{v}}^k)$,

$<\mathbf{u},\mathbf{v}>_{t,k} = \pi(c_{\mathbf{u},\mathbf{v}}^{t,k})$ and

$\beta(c_{\mathbf{u},\mathbf{v}}^k) = \pi(c_{\mathbf{u},\mathbf{v}}^{0,k})$

Then (4.28) is true.

Proposition 4.10: The following are true for *t*,*k*-dot product

(i) $<\mathbf{u},\mathbf{v}>_{t,k} = <\mathbf{v},\mathbf{u}>_{t,k}$ (4.29)

(ii) $< a\,\mathbf{u}, \mathbf{v} >_{t,k} = a < \mathbf{u}, \mathbf{v} >_{t,k}$ for any $a \in Z_2$ (4.30)

(iii) $< [\mathbf{u}\ \ a], [\mathbf{v}\ \ b] >_{t,k} = < \mathbf{u}, \mathbf{v} >_{t,k} \oplus ab$ for any $a, b \in Z_2$ (4.31)

(iv) $< [a\ \ a'], [b\ \ b'] >_{0,1} = < [a'\ \ a], [b'\ \ b] >_{0,0}$ (4.32)

(v) $< [a\ \ a'\ \ a''], [b\ \ b'\ \ b''] >_{1,0} = < [a''\ \ a'\ \ a], [b''\ \ b'\ \ b] >_{0,0} \oplus a\,a'\,b\,b'$ (4.33)

for any $a, a', b, b' \in Z_2$.

(vi) $< [a\ \ a'\,\mathbf{u}], [b\ \ b'\,\mathbf{v}] >_{t,k+1} = < [a\ \ a'], [b\ \ b'] >_{t,\varepsilon} \oplus < \mathbf{u}, \mathbf{v} >_{t,k}$ (4.34)

for any $a, a', b, b', \varepsilon \in Z_2$ and $\varepsilon = < \mathbf{u}, \mathbf{v} >_{t,k} \oplus < \gamma_k(\mathbf{u}), \mathbf{v} >_{t,k} \oplus 1$ where γ_k is a permutation on (1,2)(3,4)(5,6)…(2k-1, 2k) on m elements.

(vii) $< [a\ \ a'\,a''\,\mathbf{u}], [b\ \ b'\,b''\,\mathbf{v}] >_{t+1,k} = < [a\ \ a'\,a''], [b\ \ b'\,b''] >_{\varepsilon,0} \oplus < \mathbf{u}, \mathbf{v} >_{t,k}$ (4.35)

for any $a, a', b, b', \varepsilon \in Z_2$ and $\varepsilon = < \mathbf{u}, \mathbf{v} >_{t,0} \oplus < \alpha_k(\mathbf{u}), \mathbf{v} >_{t,0} \oplus a\,a'\,b\,b' \oplus 1$ where α_k is a permutation on (1,3)(4,6)(7,9)…(3t-2, 3t) on m elements.

Proof: (i) comes directly from (4.22).

(ii) comes from the definition of $\mathbf{C_m^{t,k}}$

(iii) is given for t=0 in Proposition 6 of (147). For $t > 0$, according to Proposition 4.7,

$$c^{t,k}_{[\mathbf{u}\,a],[\mathbf{v}\,b]} = c^{t,k}_{\mathbf{u},\mathbf{v}} \oplus 4ab. \tag{4.36}$$

Then $\pi(c^{t,k}_{[\mathbf{u}\,a],[\mathbf{v}\,b]}) = < [\mathbf{u}\ \ a], [\mathbf{v}\ \ b] >_{t,k} = \pi(c^{t,k}_{\mathbf{u},\mathbf{v}} \oplus 4ab) = < \mathbf{u}, \mathbf{v} >_{t,k} \oplus ab$.

(iv) can be observed comparing the matrices $\beta(\mathbf{C_2^0})$ and $\beta(\mathbf{C}_\mathbf{2}^1)$.

(v) can be observed comparing the matrices $\pi(\mathbf{C_3^{0,0}})$ and $\pi(\mathbf{C_3^{1,0}})$.

(vi) is proven in Proposition 6 of (147).

(vii) comes from (4.33) and (4.34).

Now in the following proposition we give the explicit formula of the *t*,*k*-dot product.

Proposition 4.11:

$$< \mathbf{u}, \mathbf{v} >_{t,k} = \left(\oplus_{i=1}^{t} \mathrm{L_i}\right) \oplus \left(\oplus_{s=1}^{k} \oplus_{i=1}^{t} \oplus_{j=i}^{t} K_s T_i T_j\right) \oplus$$
$$\oplus_{i=1}^{t} \oplus_{j=1, j\neq i}^{t} K_i v_{3i-1}(u_{2k+3i-1} \oplus u_{2k+3j-2} \; v_{2k+3j-2}) \quad (4.37)$$
$$\oplus < u, v >_k$$

$$L_i = (u_{2k+3i-2} \oplus u_{2k+3i})(v_{2k+3i-2} \oplus v_{2k+3i}) \quad (4.38)$$

$$K_i = u_{3i-2} v_{3i-2} \quad (4.39)$$

$$T_i = \left(u_{2k+3i-2} v_{2k+3i-1} + u_{2k+3i-1} v_{2k+3i-2}\right) \quad (4.40)$$

Proof: For *t*=0 it can be observed that (4.37) is equal to *k*-dot product. This is in accordance with (4.28). Induction on *t* with a fixed *k* (for simplicity fix it to 0) will be sufficient to prove Propositon 4.10. Let’s start with *t*=1,

$$\pi(\mathbf{C}_3^{\mathbf{1,0}}) = \begin{pmatrix} 0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,1\,1\,1\,1 \\ 0\,0\,1\,1\,0\,0\,1\,1 \\ 0\,0\,1\,0\,1\,1\,0\,1 \\ 0\,1\,0\,1\,0\,1\,0\,1 \\ 0\,1\,0\,1\,1\,0\,1\,0 \\ 0\,1\,1\,0\,0\,1\,1\,0 \\ 0\,1\,1\,1\,1\,0\,0\,0 \end{pmatrix} = \begin{pmatrix} <[000],\mathbf{v}>_{1,0} \\ <[001],\mathbf{v}>_{1,0} \\ <[010],\mathbf{v}>_{1,0} \\ <[011],\mathbf{v}>_{1,0} \\ <[100],\mathbf{v}>_{1,0} \\ <[101],\mathbf{v}>_{1,0} \\ <[110],\mathbf{v}>_{1,0} \\ <[111],\mathbf{v}>_{1,0} \end{pmatrix}..$$

Simplification shows that

$$<\mathbf{u},\mathbf{v}>_{1,0}= u_1v_3 \oplus u_2v_2 \oplus u_3v_1 \oplus u_1u_2v_1v_2$$

On the other hand, $L_1 = (u_1 \oplus u_3)(v_1 \oplus v_3)$, $K_1 = u_1v_1$ and $T_1 = (u_1v_2 + u_2v_1)$

(4.37) also gives $<\mathbf{u},\mathbf{v}>_{1,0}= u_1v_3 \oplus u_2v_2 \oplus u_3v_1 \oplus u_1u_2v_1v_2$.

$$\begin{aligned} <\mathbf{u},\mathbf{v}>_{t,k} &= \left(\oplus_{i=1}^{t} \mathrm{L_i}\right) \oplus \left(\oplus_{s=1}^{k} \oplus_{\iota=1}^{t} \oplus_{j=i}^{t} K_s T_i T_j\right) \oplus \\ &\oplus_{i=1}^{t} \oplus_{j=1,j\neq i}^{t} K_i v_{3i-1}(u_{2k+3i-1} \oplus u_{2k+3j-2}\ v_{2k+3j-2}) \\ &\oplus <u,v>_k \end{aligned}$$

Let the proposition be right for some t, then show that it is true for $t+1$.

$$\begin{aligned} <\mathbf{u},\mathbf{v}>_{t+1,k} &=<\mathbf{u},\mathbf{v}>_{t,k} \oplus L_{t+1} \oplus \left(\oplus_{i=1}^{t+1} T_i T_{t+1}\right) \\ &\oplus \left(\oplus_{i=1,i\neq j}^{t+1} K_i v_{3i-1}(u_{3i-1} \oplus u_{3t+1}\ v_{3t+1})\right) \end{aligned}$$

From (4.35) it is true that

$<[a\ \ a'\ a''\ \mathbf{u}],[b\ \ b'\ b''\ \mathbf{v}]>_{t+1,k} =<[a\ \ a'\ a''],[b\ \ b'\ b'']>_{\varepsilon,0} \oplus <\mathbf{u},\mathbf{v}>_{t,k}$ for any $a, a', b, b', \varepsilon \in Z_2$ and $\varepsilon =<\mathbf{u},\mathbf{v}>_{t,0} \oplus <\alpha_k(\mathbf{u}),\mathbf{v}>_{t,0} \oplus a\, a'\, b\, b' \oplus 1$ where α_k is a permutation on (1,3)(4,6)(7,9)…(3t-2, 3t) on m elements.

Then, $L_{t+1}=(a\oplus a'')(b\oplus b'')$ and $T_{t+1}=a\,b'\oplus a'b$,

$\varepsilon=<\mathbf{u},\mathbf{v}>_{t,k}\oplus<\alpha_k(\mathbf{u}),\mathbf{v}>_{t,k}\oplus a\,a'\,b\,b'\oplus 1$ where α_k is a permutation on (1,3)(4,6)(7,9)…(3t-2, 3t) on m elements.

(i) first case, $<\mathbf{u},\mathbf{v}>_{t,k}=<\alpha_k(\mathbf{u}),\mathbf{v}>_{t,k}$ for symmetric $\mathbf{u}$ vectors such that $\mathbf{u}=\alpha_k(\mathbf{u})$ for which $\varepsilon=a\,a'\,b\,b'\oplus 1$, if $a\,a'\,b\,b'=0$ then $\varepsilon=1$ and

$$\begin{aligned}<[a\ \ a'\ a''],[b\ \ b'\ b'']>_{1,0}&=ab''\oplus a''b\oplus a'b'\oplus aa'bb'\\&=ab''\oplus a''b\oplus a'b'\end{aligned}$$

(ii) second case, $<\mathbf{u},\mathbf{v}>_{t,k}=<\alpha_k(\mathbf{u}),\mathbf{v}>_{t,k}$ for symmetric $\mathbf{u}$ vectors such that $\mathbf{u}=\alpha_k(\mathbf{u})$ for which $\varepsilon=a\,a'\,b\,b'\oplus 1$, if $a\,a'\,b\,b'=1$ then $\varepsilon=0$ and $<[a\ \ a'\ a''],[b\ \ b'\ b'']>_{0,0}=ab''\oplus a''b\oplus a'b'\oplus aa'bb'$

(iii) third case, $<\mathbf{u},\mathbf{v}>_{t,0}=<\alpha_k(\mathbf{u}),\mathbf{v}>_{t,0}\oplus aa'bb'$ for asymmetric $\mathbf{u}$ vectors such that $\mathbf{u}\neq\alpha_k(\mathbf{u})$ for which $\varepsilon=0$ and $<[a\ \ a'\ a''],[b\ \ b'\ b'']>_{0,0}=ab''\oplus a''b\oplus a'b'\oplus aa'bb'$

Then for all cases (i), (ii), and (iii) numerical calculations show that

$$L_{t+1}\oplus\left(K_k\oplus_{i=1}^{t+1}T_{t+1}\right)\oplus\left(K_{i1}v_{3i-1}(u_{2k+3i-1}\oplus u_{3t+1}\,v_{3t+1})\right)=<[a\ \ a'\ a''],[b\ \ b'\ b'']>_{\varepsilon,k}$$

which finishes the proof.

All numerical examples given below from Example 4.5 to 4.9 satisfy (4.37).

Example 4.5: Let us begin with the Z_8-linear code of type $8^0 4^1 2^1$. This code contains one binary symbol and one Z_4 symbol. Now if we write all possible Z_4

symbols, we get (0),(1),(2),(3), and all possible 1-bit binary vectors, we get (0),(1). Columns of $\mathbf{G}_3^{0,1}$ consists of twice the Z_4 symbol and four times the Z_2 symbol, $\mathbf{G}_3^{0,1} = \begin{pmatrix} 0\,0\,2\,2\,4\,4\,6\,6 \\ 0\,4\,0\,4\,0\,4\,0\,4 \end{pmatrix}$. For $\phi_k^{-1}(\mathbf{u}) \in \{00, 01, 10, 11, 20, 21, 30, 31\}$, $\mathbf{u} \in \{(000), (001), (010), (011), (110), (111), (100), (101)\}$,. Then

$$\mathbf{C}_{\mathbf{3}}^{0,\mathbf{1}} = \begin{pmatrix} \mathbf{h}^{\mathbf{000}} \\ \mathbf{h}^{\mathbf{001}} \\ \mathbf{h}^{\mathbf{010}} \\ \mathbf{h}^{\mathbf{011}} \\ \mathbf{h}^{\mathbf{110}} \\ \mathbf{h}^{\mathbf{111}} \\ \mathbf{h}^{\mathbf{100}} \\ \mathbf{h}^{\mathbf{101}} \end{pmatrix} = \begin{pmatrix} 0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,4\,0\,4\,0\,4\,0\,4 \\ 0\,0\,2\,2\,4\,4\,6\,6 \\ 0\,4\,2\,6\,4\,0\,6\,2 \\ 0\,0\,4\,4\,0\,0\,4\,4 \\ 0\,4\,4\,0\,0\,4\,4\,0 \\ 0\,0\,6\,6\,4\,4\,2\,2 \\ 0\,4\,6\,2\,4\,0\,2\,6 \end{pmatrix} \text{ and}$$

$$\pi(\mathbf{C}_{\mathbf{3}}^{0,\mathbf{1}}) = \begin{pmatrix} \beta(\mathbf{h}^{\mathbf{000}}) \\ \beta(\mathbf{h}^{\mathbf{001}}) \\ \beta(\mathbf{h}^{\mathbf{010}}) \\ \beta(\mathbf{h}^{\mathbf{011}}) \\ \beta(\mathbf{h}^{\mathbf{110}}) \\ \beta(\mathbf{h}^{\mathbf{111}}) \\ \beta(\mathbf{h}^{\mathbf{100}}) \\ \beta(\mathbf{h}^{\mathbf{101}}) \end{pmatrix} = \begin{pmatrix} 0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,1\,0\,1\,0\,1\,0\,1 \\ 0\,0\,0\,0\,1\,1\,1\,1 \\ 0\,1\,0\,1\,1\,0\,1\,0 \\ 0\,0\,1\,1\,0\,0\,1\,1 \\ 0\,1\,1\,0\,0\,1\,1\,0 \\ 0\,0\,1\,1\,1\,1\,0\,0 \\ 0\,1\,1\,0\,1\,0\,0\,1 \end{pmatrix}.$$

Notice that for t=0, $\beta(\mathbf{C}_{\mathbf{3}}^{\mathbf{1}}) = \pi(\mathbf{C}_{\mathbf{3}}^{\mathbf{0,1}})$ which is expected.

The binary code $\mathbf{A}_3^{0,1}$ consists of codewords which are the rows of $\pi(\mathbf{C}_\mathbf{3}^{\mathbf{0,1}})$ and their complements. Binary Boolean functions corresponding to these codeword vectors are,

$g(v_{3,}\ v_2, v_1) = 0 = \left\langle (000), (v_{3,}\ v_2, v_1) \right\rangle_1 = l_{000}$ for the first row of $\pi(\mathbf{C}_\mathbf{3}^{\mathbf{0,1}})$

$g(v_{3,}\ v_2, v_1) = v_1 = \left\langle (010), (v_{3,}\ v_2, v_1) \right\rangle_1 = l_{001}$ for the second row,

$g(v_{3,}\ v_2, v_1) = v_3 = \left\langle (100), (v_{3,}\ v_2, v_1) \right\rangle_1 = l_{100}$ for the third row,

$g(v_{3,}\ v_2, v_1) = v_3 \oplus v_1 = \left\langle (110), (v_{3,}\ v_2, v_1) \right\rangle_1 = l_{101}$ for the fourth row,

$g(v_{3,}\ v_2, v_1) = v_2 = \left\langle (001), (v_{3,}\ v_2, v_1) \right\rangle_1 = l_{010}$ for the fifth row,

$g(v_{3,}\ v_2, v_1) = v_2 \oplus v_1 = \left\langle (011), (v_{3,}\ v_2, v_1) \right\rangle_1 = l_{011}$ for the sixth row,

$g(v_{3,}\ v_2, v_1) = v_2 \oplus v_3 = \left\langle (101), (v_{3,}\ v_2, v_1) \right\rangle_1 = l_{110}$ for the seventh row, and $g(v_{3,}\ v_2, v_1) = v_3 \oplus v_2 \oplus v_1 = \left\langle (111), (v_{3,}\ v_2, v_1) \right\rangle_1 = l_{111}$ for the last row. From the above equations one gets, $\pi(\mathbf{h}^\mathbf{u}) = \left\langle \mathbf{u}''', \mathbf{v} \right\rangle_{0,1} = l_{\hat{\mathbf{u}}}$ where $\left\langle \mathbf{u}, \mathbf{v} \right\rangle_{0,1}$ represents 0,1-dot product of the vectors **u** and **v**. Notice that every row is the truth table of a linear function of **v**. The complement functions corresponding to $(\mathbf{h}^\mathbf{u} + \mathbf{4})$ are then affine functions of **v**.

Example 4.6: Let us now begin with the Z_8-linear code of type $8^0 4^2 2^0$. This code contains two Z_4 symbols and no binary symbols. Now if we write all possible 2-symbol Z_4 vectors, we get (00), (01), (02), (03), (10), (11), (12), (13), (20),(21),(22),(23), (30),(31),(32),(33. Columns of $\mathbf{G}_4^{0,2}$ consists of two Z_4 symbols multiplied by 2,

$$\mathbf{G}_4^{0,2} = \begin{pmatrix} 0&0&0&0&2&2&2&2&4&4&4&4&6&6&6&6 \\ 0&2&4&6&0&2&4&6&0&2&4&6&0&2&4&6 \end{pmatrix}.$$

$$\mathbf{C}_{\mathbf{4}}^{0,\mathbf{2}} = \begin{pmatrix} 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&2&4&6&0&2&4&6&0&2&4&6&0&2&4&6 \\ 0&4&0&4&0&4&0&4&0&4&0&4&0&4&0&4 \\ 0&6&4&2&0&6&4&2&0&6&4&2&0&6&4&2 \\ 0&0&0&0&1&1&1&1&2&2&2&2&3&3&3&3 \\ 0&1&2&3&1&2&3&0&2&3&0&1&3&0&1&2 \\ 0&2&0&2&1&3&1&3&2&0&2&0&3&1&3&1 \\ 0&3&2&1&1&0&3&2&2&1&0&3&3&2&1&0 \\ 0&0&0&0&2&2&2&2&0&0&0&0&2&2&2&2 \\ 0&1&2&3&2&3&0&1&0&1&2&3&2&3&0&1 \\ 0&2&0&2&2&0&2&0&0&2&0&2&2&0&2&0 \\ 0&3&2&1&2&1&0&3&0&3&2&1&2&1&0&3 \\ 0&0&0&0&3&3&3&3&2&2&2&2&1&1&1&1 \\ 0&1&2&3&3&0&2&1&2&3&0&1&1&2&3&0 \\ 0&2&0&2&3&1&3&1&2&0&2&0&1&3&1&3 \\ 0&3&2&1&3&2&1&0&2&1&0&3&1&0&3&2 \end{pmatrix} \text{ and}$$

$$\pi(\mathbf{C}_4^{\mathbf{0},2}) = \begin{pmatrix} 0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1 \\ 0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1 \\ 0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0\,1\,1\,1\,1\,1\,1\,1\,1 \\ 0\,0\,1\,1\,0\,1\,1\,0\,1\,1\,0\,0\,1\,0\,0\,1 \\ 0\,1\,0\,1\,0\,1\,0\,1\,1\,0\,1\,0\,1\,0\,1\,0 \\ 0\,1\,1\,0\,0\,0\,1\,1\,1\,0\,0\,1\,1\,1\,0\,0 \\ 0\,0\,0\,0\,1\,1\,1\,1\,0\,0\,0\,0\,1\,1\,1\,1 \\ 0\,0\,1\,1\,1\,1\,0\,0\,0\,0\,1\,1\,1\,1\,0\,0 \\ 0\,1\,0\,1\,1\,0\,1\,0\,0\,1\,0\,1\,1\,0\,1\,0 \\ 0\,1\,1\,0\,1\,0\,0\,1\,0\,1\,1\,0\,1\,0\,0\,1 \\ 0\,0\,0\,0\,1\,1\,1\,1\,1\,1\,1\,1\,0\,0\,0\,0 \\ 0\,0\,1\,1\,1\,0\,1\,0\,1\,1\,0\,0\,0\,1\,1\,0 \\ 0\,1\,0\,1\,1\,0\,1\,0\,1\,0\,1\,0\,0\,1\,0\,1 \\ 0\,1\,1\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0\,1\,1 \end{pmatrix}.$$

The binary code $\mathbf{A}_4^{0,2}$ consists of codewords which are the rows of $\pi(\mathbf{C}_\mathbf{4}^{\mathbf{0,2}})$ and their complements. The binary Boolean function corresponding to these codeword vectors are,

$g(v_4, v_3, v_2, v_1) = 0 = \langle (0000), (v_4, v_3, v_2, v_1) \rangle_2 = l_{0000}$ for the first row of $\pi(\mathbf{C}_\mathbf{4}^{\mathbf{0,2}})$,

$g(v_4, v_3, v_2, v_1) = v_2 = \langle (0001), (v_4, v_3, v_2, v_1) \rangle_2 = l_{0010}$ for the second row,

$g(v_4, v_3, v_2, v_1) = v_1 = \langle (0010), (v_4, v_3, v_2, v_1) \rangle_2 = l_{0001}$ for the third row, $g(v_4, v_3, v_2, v_1) = v_1 \oplus v_2 = \langle (0011), (v_4, v_3, v_2, v_1) \rangle_2 = l_{0011}$ for the fourth row and other rows also satisfy (4.37). Thus the code $\mathbf{A}_4^{0,2}$ is equal to the code $\mathbf{A}_4^2$.

Example 4.7: Let us now begin with the Z_8-linear code of type $8^1 4^0 2^0$. This code contains one Z_8 symbol.

$\mathbf{G}_3^{1,0} = (0\,1\,2\,3\;\;4\,5\,6\,7\,)$,

$$\mathbf{C}_3^{1,0} = \begin{pmatrix} 0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,1\,2\,3\,4\,5\,6\,7 \\ 0\,2\,4\,6\,0\,2\,4\,6 \\ 0\,3\,4\,2\,4\,7\,2\,5 \\ 0\,4\,0\,4\,0\,4\,0\,4 \\ 0\,5\,2\,7\,4\,1\,6\,3 \\ 0\,6\,4\,2\,0\,6\,4\,2 \\ 0\,7\,6\,5\,4\,3\,2\,1 \end{pmatrix}, \ \pi(\mathbf{C}_3^{1,0}) = \begin{pmatrix} 0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,1\,1\,1\,1 \\ 0\,0\,1\,1\,0\,0\,1\,1 \\ 0\,0\,1\,0\,1\,1\,0\,1 \\ 0\,1\,0\,1\,0\,1\,0\,1 \\ 0\,1\,0\,1\,1\,0\,1\,0 \\ 0\,1\,1\,0\,0\,1\,1\,0 \\ 0\,1\,1\,1\,1\,0\,0\,0 \end{pmatrix}.$$

Table 4.5: Binary Boolean function corresponding to the codeword vectors $\pi(\mathbf{h_8}^{\mathbf{u}})$

u	$\langle u,v\rangle_{1,0}$
000	0
001	v_3
010	v_2
011	$v_3 \oplus v_2 \oplus v_1 v_2$
100	v_1
101	$v_1 \oplus v_3$
110	$v_1 \oplus v_2$
111	$v_1 \oplus v_2 \oplus v_3 \oplus v_1 v_2$

All rows of the matrix $\pi(\mathbf{C}_3^{\mathbf{1,0}})$ satisfy (4.37). Six of the 1,0-affine functions are affine and two of them are quadratic.

Example 4.8: Let us now begin with the Z_8-linear code of type $8^1 4^1 2^0$.. This code contains one Z_4 symbols and one Z_8 symbol.

$$\mathbf{G}_5^{1,1} = \begin{pmatrix} 0\,0\,0\,0 & 1\,1\,1\,1 & 2\,2\,2\,2 & 3\,3\,3\,3 & 4\,4\,4\,4 & 5\,5\,5\,5 & 6\,6\,6\,6 & 7\,7\,7\,7 \\ 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 \end{pmatrix}.$$

$$
\mathbf{C}_5^{1,1} = \begin{pmatrix} 0\,0\,0\,0 & 0\,0\,0\,0 & 0\,0\,0\,0 & 0\,0\,0\,0 & 0\,0\,0\,0 & 0\,0\,0\,0 & 0\,0\,0\,0 & 0\,0\,0\,0 \\ 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 & 0\,2\,4\,6 \\ 0\,0\,0\,0 & 1\,1\,1\,1 & 2\,2\,2\,2 & 3\,3\,3\,3 & 4\,4\,4\,4 & 5\,5\,5\,5 & 6\,6\,6\,6 & 7\,7\,7\,7 \\ 0\,2\,4\,6 & 1\,3\,5\,7 & 2\,4\,6\,0 & 3\,5\,7\,1 & 4\,6\,0\,2 & 5\,7\,1\,3 & 6\,0\,2\,4 & 7\,1\,3\,5 \\ \vdots & & & & & & & \end{pmatrix} \quad \text{and}
$$

$$
\pi(\mathbf{C}_5^{1,1}) = \begin{pmatrix} 0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1 \\ 0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1 \\ 0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0\,0\,1\,1\,0 \\ 0\,0\,0\,0\,0\,0\,0\,0\,1\,1\,1\,1\,1\,1\,1\,1\,0\,0\,0\,0\,0\,0\,0\,0\,1\,1\,1\,1\,1\,1\,1\,1 \\ 0\,0\,1\,1\,0\,1\,1\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0\,0\,1\,1\,1\,1\,1\,1\,1\,1\,0\,0\,0\,0\,0 \\ \vdots \\ 0\,1\,1\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0\,1\,1\,0\,1\,1\,0\,1\,1\,0\,0\,1\,0\,0\,1\,0\,0\,1\,1 \end{pmatrix}.
$$

The binary code $\mathbf{A}_5^{1,1}$ consists of codewords which are the rows of $\pi(\mathbf{C}_5^{1,1})$ and their complements. The binary Boolean function corresponding to these codeword vectors are given in Table 4.6.

Table 4.6: Binary Boolean function corresponding to these codeword vectors $\pi(\mathbf{h}_8^{\mathbf{u}})$

u	$<u,v>_{1,1}$
00000	0
00001	v_2
00010	v_1
00011	$v_1 \oplus v_2$
00100	v_5
00101	$v_2 \oplus v_2 \oplus v_1 v_4$
00110	$v_1 \oplus v_5$
00111	$v_1 \oplus v_2 \oplus v_5 \oplus v_1 v_4$
01000	v_4
01001	$v_2 \oplus v_4 \oplus v_1 v_3$
01010	$v_1 \oplus v_4$
01011	$v_1 \oplus v_2 \oplus v_4 \oplus v_1 v_3$
01100	$v_5 \oplus v_4 \oplus v_4 v_5$
01101	$v_2 \oplus v_5 \oplus v_4 \oplus v_1 v_4 \oplus v_3 v_4$
01110	$v_1 \oplus v_5 \oplus v_4 \oplus v_3 v_4$
01111	$v_1 \oplus v_2 \oplus v_5 \oplus v_4 \oplus v_1 v_3 \oplus v_1 v_4 \oplus v_3 v_4$

Table 4.6 (continued)

10000	v_3
10001	$v_2 \oplus v_3$
10010	$v_1 \oplus v_3$
10011	$v_1 \oplus v_2 \oplus v_3$
10100	$v_5 \oplus v_3$
10101	$v_2 \oplus v_3 \oplus v_5 \oplus v_1 v_4$
10110	$v_5 \oplus v_3 \oplus v_1$
10111	$v_5 \oplus v_3 \oplus v_2 \oplus v_1 \oplus v_1 v_4$
11000	$v_4 \oplus v_3$
11001	$v_4 \oplus v_3 \oplus v_2 \oplus v_1 v_3$
11010	$v_4 \oplus v_3 \oplus v_1$
11011	$v_4 \oplus v_3 \oplus v_2 \oplus v_1 \oplus v_1 v_3$
11100	$v_5 \oplus v_4 \oplus v_3 \oplus v_4 v_3$
11101	$v_5 \oplus v_4 \oplus v_3 \oplus v_2 \oplus v_1 v_3 \oplus v_1 v_4 \oplus v_3 v_4$
11110	$v_5 \oplus v_4 \oplus v_3 \oplus v_1 \oplus v_1 v_3 \oplus v_3 v_4$
11111	$v_5 \oplus v_4 \oplus v_3 \oplus v_2 \oplus v_1 \oplus v_1 v_3 \oplus v_1 v_4$

All rows of the matrix $\pi(\mathbf{C}_5^{1,1})$ satisfy (5.38). It is observed that five variables are partitioned into one 2-bit for *k*=1 part and 3-bit for *t*=1,i.e., $(v_1 \quad v_2)(v_3 \quad v_4 \quad v_5)$ and on first part a *k*-dot product is performed and on the second part a *tk*-dot product is performed (with *k*=0).

Example 4.9: Let us now begin with the Z_8-linear code of type $8^2 4^0 2^0$ .. This code contains two Z_8 symbols.

$$\mathbf{G}_\mathbf{6}^{\mathbf{2,0}} = \begin{pmatrix} 0000000011111111\cdots 77777777 \\ 0123456701234567\cdots 01234567 \end{pmatrix},$$

$$\mathbf{C}_\mathbf{6}^{\mathbf{2,0}} = \begin{pmatrix} 0000000000000000\cdots 00000000 \\ 0123456701234567\cdots 01234567 \\ 0246024602460246\cdots 02460246 \\ 03424725\cdots \\ 04040404\cdots \\ 05274163\cdots \\ \vdots \end{pmatrix} \text{ and}$$

$$\pi(\mathbf{G}_\mathbf{6}^{\mathbf{2,0}}) = \begin{pmatrix} 0000000000000000\cdots 00000000 \\ 0000111100001111\cdots 00001111 \\ 0011001100110011\cdots 00110011 \\ 01101101\cdots \\ 01010101\cdots \\ 01011010\cdots \\ \vdots \end{pmatrix}$$

Table 4.7 shows 6-bit Boolean functions corresponding to each of $\pi(\mathbf{G}_\mathbf{6}^{\mathbf{2,0}})$.

Table 4.7: Binary Boolean function corresponding to these codeword vectors $\pi(\mathbf{h}_8^{\mathbf{u}})$

u	<u,v>$_{2,0}$
000000	0
000001	v_3
000010	v_2
000011	$v_2 \oplus v_3 \oplus v_1v_2$
000100	v_1
000101	$v_1 \oplus v_3$
000110	$v_1 \oplus v_2$
000111	$v_1 \oplus v_2 \oplus v_3 \oplus v_1v_2$
001000	v_6
001001	$v_3 \oplus v_6 \oplus v_1v_2v_4 \oplus v_2v_5 \oplus v_1v_4v_5$
001010	$v_2 \oplus v_6 \oplus v_1v_5$
001011	$v_1 \oplus v_3 \oplus v_6 \oplus v_1v_2 \oplus v_2v_5 \oplus v_1v_5 \oplus v_1v_2v_4 \oplus v_1v_4v_5$
001100	$v_1 \oplus v_6$
001101	$v_1 \oplus v_3 \oplus v_6 \oplus v_2v_5 \oplus v_3v_4 \oplus v_1v_2v_4 \oplus v_1v_4v_5$
001110	$v_1 \oplus v_2 \oplus v_6 \oplus v_1v_5$

Table 4.7 (continued)

001111	$v_1 \oplus v_2 \oplus v_3 \oplus v_6 \oplus v_1v_2 \oplus v_1v_4 \oplus v_1v_5 \oplus v_1v_2v_4 \oplus v_1v_4v_5$
010000	v_5
010001	$v_5 \oplus v_3 \oplus v_2v_4$
010010	$v_5 \oplus v_2 \oplus v_1v_4$
010011	$v_5 \oplus v_3 \oplus v_2 \oplus v_2v_4 \oplus v_1v_2 \oplus v_1v_4$
010100	$v_5 \oplus v_1$
010101	$v_1 \oplus v_3 \oplus v_5 \oplus v_2v_4$
010110	$v_1 \oplus v_2 \oplus v_5 \oplus v_1v_4$
010111	$v_1 \oplus v_2 \oplus v_3 \oplus v_5 \oplus v_1v_2 \oplus v_2v_4 \oplus v_1v_4$
011000	$v_6 \oplus v_5 \oplus v_5v_4$
011001	$v_3 \oplus v_5 \oplus v_6 \oplus v_1v_4 \oplus v_2v_4 \oplus v_1v_2v_4 \oplus v_2v_5 \oplus v_4v_5 \oplus v_1v_4v_5$
011010	$v_2 \oplus v_5 \oplus v_6 \oplus v_1v_5 \oplus v_2v_4 \oplus v_1v_4$
011011	$v_2 \oplus v_3 \oplus v_5 \oplus v_6 \oplus v_1v_2 \oplus v_1v_4 \oplus v_2v_4 \oplus v_1v_2v_4 \oplus v_2v_5 \oplus v_1v_5 \oplus$

Table 4.7 (continued)

011100	$v_1 \oplus v_5 \oplus v_6 \oplus v_5 v_4$
011101	$v_1 \oplus v_3 \oplus v_5 \oplus v_6 \oplus v_1 v_4 \oplus v_2 v_4 \oplus v_1 v_2 v_4 \oplus v_2 v_5 \oplus v_4 v_5$
011110	$v_1 \oplus v_2 \oplus v_5 \oplus v_6 \oplus v_1 v_4 \oplus v_1 v_5 \oplus v_4 v_5$
011111	$v_1 \oplus v_2 \oplus v_3 \oplus v_5 \oplus v_6 \oplus v_1 v_2 \oplus v_1 v_4 \oplus v_2 v_4 \oplus v_1 v_2 v_4$ $\oplus v_2 v_5 \oplus v_1 v_5 \oplus v_4 v_5 \oplus v_1 v_4 v_5$
100000	v_4
100001	$v_3 \oplus v_4$
100010	$v_2 \oplus v_4$
100100	$v_1 \oplus v_4$
101000	$v_6 \oplus v_4$
110000	$v_5 \oplus v_4$

CHAPTER 5

k-BENT AND *t,k*-BENT FUNCTIONS

Bent functions, which are at maximum distance to affine functions, form a well-known topic in cryptology. They are first studied by Dillon [49] and Rothaus [94] in seventies. Rothaus used the word “bent” for the first time in the litterature in 1970. MacWilliams and Sloane [76] observed that bent functions are stronly linked with first order Reed Muller codes. And in 2008, Tokareva defined [104] *k*-bent functions starting from Z_4-linear codes.

In this chapter, we study bent functions, from the conventional Rothaus and Dillan as well as Maiorana McFarland bent functions to the Tokareva’s *k*-bent functions. We defined *t,k*-Walsh transform and *t,k*-nonlinearity to propose the *t,k*-bent functions. We give Propositons 5.3 to show that the *t,k*-Walsh transform of a Boolean function satisfies the Parseval’s equation. We then relate the *t,k*-nonlinearity to *t,k*-Walsh transform in Propositons 5.4. Next, we suggest a new class of bent functions, the *t,k*-bent functions, which are extensions of *k*-bent functions, depending on the *t,k*-dot product definition given in Chapter 4. We give Proposition 5.5 to show that the set of $(t+1),k$-bent functions and $t,(k+1)$-bent functions are subsets of the set of *t,k*-bent functions. In sections 5.3 and 5.4, we show that the newly defined classes of bent, namely Tokareva's *k*-bent and our *t,k*-bent functions are affine equivalent to the well-known Maiorana McFarland class of bent functions. As a cryptology application, in section 5.5, we propose the method of cubic cryptanalysis for block ciphers. It is a generalization of the well-known method of linear cryptanalysis given in 1993 by M. Matsui [79]. In our method we approximate Boolean functions by *t,k*- affine functions. The newly introduced *t,k*-bent functions are claimed to be strong against cubic cryptanalysis, since they are at maximum distance to *t,k*- affine functions, which contain affine, quadratic and cubic functions.

5.1 Conventional Bent Function Definitions and Properties

For the rest of the chapter, let $f : GF(2)^m \to GF(2)$ be an *m*-bit binary function. In this section, we will give conventional definitions of bent functions including Rothaus and Maiorana McFarland class bent functions.

Definition 5.1: A function *f* is called bent if all of the components of the Walsh spectrum of *f* have the same magnitude, up to the absolute value.

Definition 5.2: A function *f* is called bent if it is at maximum possible distance to all affine functions. This implies that bent functions have maximum possible nonlinearity.

From Definition 5.1 and Parseval's equation it is observed for bent *f*

$$\left|W_f(\mathbf{w})\right| = 2^{m/2} \text{ for } \mathbf{w} \in GF(2)^m. \tag{5.1}$$

(5.1) requires *m* to be even. Since bent functions are defined only for even values of *m*, from now on unless otherwise stated explicitly we assume that *m* is even and *m* > 2.

Theorem 5.1.: [49] If *f* is a bent function, with $m = 2k$; then the degree of *f* is at most *k*, except for the case *k* = 1.

Proof of this theorem is given in [49]. This theorem gives us an obvious upper bound for the number of bent functions which is

$$\text{max number of } f = 2^{\binom{n}{0} + \binom{n}{1} + \dots \binom{n}{n/2}} \tag{5.2}$$

Theorem 5.2.: [49] A bent function is invariant

(i) Under a linear or an affine transformation in coordinates, that is *f* is bent if and only if the function $h = f \circ \theta$ is bent where $\theta(\mathbf{x}) = x\mathbf{A} \oplus \mathbf{b}$, **A** is a nonsingular matrix of order *m* and **b** is any vector in $GF(2)^m$.

(ii) By adding an affine function, that is *f* is bent if and only if $f \oplus \phi$ is bent for any affine function ϕ.

5.1.1 Rothaus' Bent Function Classes

In 1975, Rothaus [49] presented the first two classes of bent functions. He made an exhaustive search on all polynomials in $GF(2)^6$. He found two general classes of bent functions.

Theorem 5.3: (Rothaus Class I) [49] Let $m = 2k$ and $\mathbf{x}, \mathbf{y} \in GF(2)^k$ and *f* be a *k*-variable function. Then the *m* variable function

$$Q(\mathbf{x}, \mathbf{y}) = x_1 y_1 \oplus x_2 y_2 \oplus \dots \oplus x_k y_k \oplus f \tag{5.3}$$

is bent.

Theorem 5.4: (Rothaus Class II) [49] Let $A(\mathbf{x}), B(\mathbf{x}), C(\mathbf{x})$ be $2k$-variable bent functions such that $A(\mathbf{x}) \oplus B(\mathbf{x}) \oplus C(\mathbf{x})$ be also bent. Let $y, z \in GF(2)$. Then the function

$$Q(\mathbf{x}, y, z) = A(\mathbf{x})B(\mathbf{x}) \oplus A(\mathbf{x})C(\mathbf{x}) \oplus B(\mathbf{x})C(\mathbf{x}) \oplus (A(\mathbf{x}) \oplus B(\mathbf{x}))y \oplus (A(\mathbf{x}) \oplus C(\mathbf{x}))z \oplus yz \tag{5.4}$$

is a bent function on $GF(2)^{2k+2}$.

5.1.2 Maiorana McFarland's Class

Maiorana McFarland's class of bent functions is a generalization of Rothaus' Class I.

Theorem 5.5: (Maiorana McFarland Class) [80] Let k be an arbitrary positive integer and $m = 2k$. Then the m-variable function f given by,

$$f(\mathbf{x}, \mathbf{y}) = <\mathbf{y}, \pi(\mathbf{x})> \oplus g(\mathbf{x}) \tag{5.5}$$

where $\mathbf{x}, \mathbf{y} \in GF(2)^k$ and π is an arbitrary permutation of $GF(2)^k$ and g is an arbitrary k-variable function, is bent.

5.1.3 Tokareva's *k*-bent Functions

Tokareva defined k-bent functions [104] from the definition of k-affine functions, which were defined in Section 4.3 of this thesis.

Definition 5.3: [104] The k-Walsh transform of a Boolean function $f \in GF(2)^m$ is the integer valued function

$$W_f^{(k)}(\mathbf{w}) = \sum_{\mathbf{x} \in GF(2)^m} (-1)^{<\mathbf{x}, \mathbf{w}>_k} (-1)^{f(\mathbf{x})} \tag{5.6}$$

where $0 \le k \le m/2$.

Definition 5.4: [104] By k-nonlinearity $N_f^{(k)}$ of a function f the distance between f and the class ψ_m^k is meant.

Proposition 5.1: [104] It is true that

$$N_f^{(k)} = 2^{m-1} - \frac{1}{2}\max_{\mathbf{w}}\left|W_f^{(k)}(\mathbf{w})\right| \tag{5.7}$$

Definition 5.5: [105] For any integers m, k such that $0 \le k \le m/2$ we call a function f, k-bent if and only if all $W_f^{(k)}(\mathbf{w}) = \pm 2^{m/2}$.

Proposition 5.2: [106] For k-bent functions B_m^k we have

$$B_m^0 = B_m^1 \supset B_m^2 \cdots \supset B_m^{m/2} \tag{5.8}$$

Proof is given in [106].

For m=4 all 1-bent and 2-bent functions are examined numerically and Table 5.1 is constructed.

Table 5.1: Properties of 1 and 2-bent 4-variable functions

k	# of k-bent functions	R_f	Deg_f	N_f^0	N_f^1	N_f^2
1	896	16,0...,0	2	6	6	4,6
2	384	16,0...,0	2	6	6	6

There are 896 1-bent 4-variable functions. 384 of them are 2-bent and 512 are not 2-bent (only 1-bent). Maximum possible nonlinearity value is 6 for m=4. 2-bent functions have

$$N_f^0 = N_f^1 = N_f^2 = 6\,. \tag{5.9}$$

But only 384 of 1-bent functions satisfy (5.9). These functions are shown to be exactly equivalent to the 2-bent functions. 512 of 1-bent functions have

$$N_f^0 = N_f^1 = 6,\; N_f^2 = 4 \tag{5.10}$$

All 1-bent and 2-bent functions have autocorrelation spectrum (16 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0) which is the property of bent functions. This is expected.

Note that all 1-bent and 2-bent functions have degree equal to 2. They are quadratic. Since bent functions must be distinct from affine functions and Theorem 5.1 says that $\deg(f) \le k = 2$ for m=4. This is what we expect.

Example 5.1: Numerical analysis gives all 1 and 2-bent, 4-variable functions. Some examples for the truth tables of these functions are listed below.

$f_1 = [0000001101010110]$, $f_2 = [0000011000111010]$, $f_3 = [0011000010101001]$ and $f_4 = [0110000011001010]$ are truth tables of 1-bent functions.

$f_5 = [0000001101011001]$ and $f_6 = [1000001011011000]$ are truth tables of 2-bent functions.

Example 5.2: Numerical analysis give some of the 1, 2 and 3-bent 6-variable functions. Some examples for the truth tables of these functions are listed below.

$f_1 = [0110000011001010011000101100001001100000100010100110000011001010]$
is 1-bent.

$f_2 = [0000010001001010011000101000001001100000100010100110000011001110]$
is 2-bent.

$f_3 = [0000001110001010011000101100001000000001100010101110000011001010]$
is 3-bent.

5.2 *t,k*-bent Functions

We will now define *t*,*k*-bent functions from the definition of *t*,*k*-affine functions which were defined in Section 4.4 of this thesis.

Definition 5.6: The *t*,*k*-Walsh transform of a Boolean function $f(\mathbf{x}) \in GF(2)$ with $\mathbf{x} \in GF(2)^m$ is the integer valued function

$$W_f^{(t,k)}(\mathbf{w}) = \sum_{\mathbf{x} \in GF(2)^m} (-1)^{<\mathbf{x},\mathbf{w}>_{t,k}} (-1)^{f(\mathbf{x})} \tag{5.11}$$

where, $0 \le t \le m/3$ and $0 \le k \le (m-3t)/2$. Here $<\mathbf{x},\mathbf{w}>_{t,k}$ is the *t*,*k*-dot product defined in section 4.4 of this thesis.

Proposition 5.3: The *t*,*k*-Walsh transform of a Boolean function satisfies the Parseval's equation,

$$\sum_{\mathbf{w} \in GF(2)^m} (W_f^{t,k}(\mathbf{w}))^2 = 2^{2m}. \tag{5.12}$$

Proof: Note that for *t*=*k*=0 (5.11) gives the Walsh transform. For *t*=0 (5.11) is equal to the *k*-Walsh transform which obeys the Parseval's rule [104],

$$\sum_{\mathbf{w}\in GF(2)^m} (W_f^k(\mathbf{w}))^2 = 2^{2m}. \tag{5.13}$$

If $t \neq 0$ then the matrix $\pi(\mathbf{C_m^{t,k}})$ after replacing any element c by $(-1)^c$ becomes a Hadamard matrix.

$$\begin{aligned}
\sum_{\mathbf{w}\in GF(2)^m} (W_f^{t,k}(\mathbf{w}))^2 &= \sum_{\mathbf{w}\in GF(2)^m} \left(\sum_{\mathbf{x}\in GF(2)^m} (-1)^{<\mathbf{x},\mathbf{w}>_{t,k}} (-1)^{f(\mathbf{x})} \right)^2 \\
&= \sum_{\mathbf{w}} \sum_{\mathbf{x},v} (-1)^{<\mathbf{x},\mathbf{w}>_{t,k} \oplus f(\mathbf{x})} (-1)^{<v,\mathbf{w}>_{t,k} \oplus f(v)} \\
&= \sum_{\mathbf{x},v} (-1)^{f(\mathbf{x}) \oplus f(v)} \sum_{\mathbf{w}} (-1)^{<\mathbf{x},\mathbf{w}>_{t,k} \oplus <v,\mathbf{w}>_{t,k}}
\end{aligned} \tag{5.14}$$

and since,

$$\sum_{\mathbf{w}} (-1)^{<\mathbf{x},\mathbf{w}>_{t,k} \oplus <v,\mathbf{w}>_{t,k}} = \begin{cases} 2^m & if\ \mathbf{x} = v \\ 0 & else \end{cases} \tag{5.15}$$

Then,

$$\sum_{\mathbf{w}\in GF(2)^m} (W_f^{t,k}(\mathbf{w}))^2 = \sum_{\mathbf{x},v} 2^m = 2^{2m}.$$

Definition 5.7: The *t*,*k*-nonlinearity $N_f^{(t,k)}$ of a function f, is the distance between *f* and the class $\psi_m^{t,k}$, which contains all *t*,*k*-affine functions.

Proposition 5.4: It is true that

$$N_f^{(t,k)} = 2^{m-1} - \frac{1}{2} \max_{\mathbf{w}} \left| W_f^{(t,k)}(\mathbf{w}) \right| \tag{5.16}$$

Proof: Let a binary vector $g_{\mathbf{u}} = \pi(\mathbf{h}_{\mathbf{8}}^{\mathbf{u}})$, then we have $g_{\mathbf{u}}(\mathbf{v}) = <\mathbf{u},\mathbf{v}>_{t,k}$.

$$N_f^{(t,k)} = \min_{g_{\mathbf{u}} \in \psi_m^{t,k}} (dist(f,g)) = \min(d(f,g_{\mathbf{u}}), d(f,g_{\mathbf{u}} \oplus 1)) \qquad (5.17)$$

From the definition of $W_f^{t,k}(\mathbf{w})$ and from (4.30),

$$d(f,g_{\mathbf{u}}) = 2^{m-1} - \frac{1}{2} W_f^{t,k}(\mathbf{w}) \qquad (5.18)$$

Using (5.17) and (5.18) we get (5.16).

Definition 5.8: For any integers *m*, t, *k* such that $0 \le t \le m/3$ and $0 \le k \le (m-3t)/2$ we call a function f *t*,*k*-bent if and only if all

$$W_f^{(t,k)}(\mathbf{w}) = \pm 2^{m/2}. \qquad (5.19)$$

Note that the *t*,*k*-bent functions are at maximum distance to *t*,*k*-affine functions.

Denote by $B_m^{t,k}$ the class of all *t*,*k*-bent functions in *m* variables. Then we give Proposition 5.5 to show that the set of (*t*+1),*k*-bent functions and *t*,(*k*+1)-bent functions are subsets of the set of *t*,*k*-bent functions.

Proposition 5.5: For *t*,*k*-bent functions $B_m^{t,k}$ we have

$$\text{(i) } B_m^{t,0} = B_m^{t,1} \supset B_m^{t,2} \cdots \supset B_m^{t,(m/2)} \qquad (5.20)$$

$$\text{(ii) } B_m^{0,k} = B_m^{1,k} \supset B_m^{2,k} \cdots \supset B_m^{(m/3),k} \qquad (5.21)$$

Proof:

(i) (5.20) comes from (5.8).

(ii) The *m*-variable function

$$f(a_1, a_1', \cdots a_{t-1}, a_{t-1}', \mathbf{u}', \mathbf{u}'') = \left(\oplus_{i=1}^{t} s_i(a_i, a_i')\right) \oplus \varphi(\mathbf{u}') \oplus q(\mathbf{u}'') \qquad (5.22)$$

is t,k-bent but it is not $(t+1),k$ bent. Here s_i are $1,k$-bent 2-variable functions, $q(\mathbf{u}'')$ is a $(m\text{-}3t\text{-}2)$ variable $1,k$-bent function and $\varphi(\mathbf{u}')$ is a t,k-bent t-variable function.

For m=6 all 1,0-bent and 2,0-bent functions are numerically examined and Table 5.2 is constructed.

Table 5.2: Properties of 1,0 and 2,0-bent 6-variable functions, k=0

t	R_f	N_f^0	N_f^1	N_f^2
1	64,0...,0	28	28	24,28
2	64,0...,0	28	28	28

All 1,0-bent and 2,0-bent functions have autocorrelation spectrum (64 0 0 0 0 0 0 0 0 0 0 0 0 0 0) which is the property of bent functions. This is expected.

Example 5.3: Numerical analysis give some of the 1,0 and 2,0-bent 6-variable functions. Some examples for these functions are listed below.

$$f_1(x_1, x_2, x_3, x_4, x_5, x_6) = x_1x_2 \oplus x_3x_4 \oplus x_5x_6,$$

$$f_2(x_1, x_2, x_3, x_4, x_5, x_6) = x_1x_4 \oplus x_2x_5 \oplus x_3x_6$$

are 2,0-bent functions which are also Maiorana McFarland type bent functions.

$$f_3(x_1, x_2, x_3, x_4, x_5, x_6) = x_1x_2x_3 \oplus x_3x_4 \oplus x_5x_6,$$

$$f_4(x_1, x_2, x_3, x_4, x_5, x_6) = x_1x_4x_5 \oplus x_2x_5 \oplus x_3x_6$$

are 1,0-bent functions.

Example 5.4: Numerical analysis give some of the 1, 2 and 3-bent 10-variable functions.

$$f_1(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}) = x_1 x_2 \oplus x_3 x_4 \oplus x_5 x_6 \oplus x_7 x_8 \oplus x_9 x_{10}$$

is a 3,0-bent function which is also Maiorana McFarland type bent functions. $N_{f_1}^{3,0} = 496$.

$$f_2(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}) = x_1 x_2 x_3 \oplus x_3 x_4 \oplus x_5 x_6 \oplus x_7 x_8 \oplus x_9 x_{10}$$

is 2,0-bent function. $N_{f_2}^{2,0} = 496$ and $N_{f_2}^{3,0} = 492$.

5.3 Affine Equivalence Analysis of Tokareva's *k*-bent Functions and Maiorana McFarland Class Bent Functions

In this section, we will show that Tokareva's *k*-bent functions are affine equivalent to the well-known Maiorana McFarland class of bent functions in Proposition 5.6.

Proposition 5.6: Tokareva's *k*-bent functions are affine equivalent to the Maiorana McFarland class of bent functions. Maiorana McFarland class bent functions $f(\mathbf{x}, \mathbf{y}) = < \mathbf{y}, \pi(\mathbf{x}) > \oplus g(\mathbf{x})$ with the permutation $\pi_4(\mathbf{x})$ and $g(\mathbf{x}) = 0$ and the notation that (x_{2i-1}, x_{2i}) is the i^{th} pair, with $1 \le i \le m/2$, such that,

(i) Permutations of different pairs, or

(ii) Permutations in a pair

result in Tokareva's (*m*/2)-bent functions.

Proof: We will prove by induction, take m=4 and k=2, $\mathbf{x} = [x_1 \ x_2]$ and $\mathbf{y} = [x_3 \ x_4]$, then $f_1(x_1,\cdots x_4) = [x_3 \ \ x_4].[x_2 \ \ x_1] = x_2x_3 \oplus x_1x_4$ with $\pi_4(\mathbf{x}) = (x_1 \ x_2)$ and $g(\mathbf{x}) = 0$. Then $W_f^1 = W_f^2 = 6$ implies that f_1 is 2-bent.

Assume for m=2k, that $f_2(\mathbf{x},\mathbf{y}) =< \mathbf{y}, \pi_4(\mathbf{x}) > \oplus\, g(\mathbf{x})$ is k-bent. Then show that for m=2k+2, that $f_3(\mathbf{x},\mathbf{y}) =< \mathbf{y}, \pi_4(\mathbf{x}) > \oplus\, g(\mathbf{x})$ is (k+1)-bent.

For m=2k, take $\mathbf{x} = [x_1 \ x_3 \cdots x_{2k-1}]$ and $\mathbf{y} = [x_2 \ x_4 \cdots x_{2k}]$, then assume $f_2(x_1,\cdots x_m) = [x_2 \ x_4 \cdots x_{2k}][x_3 \ x_1 \cdots x_{2k-1}] = x_3x_2 \oplus x_4x_1 \oplus \cdots \oplus x_{2k}x_{k-1}$ is k-bent with $W_{f_2}^k = 2^{2k-1} - 2^{k-1}$.

Then for m=2k+2, take $\mathbf{x} = [x_1 \ x_3 \cdots x_{2k-1}x_{2k+1}]$ and $\mathbf{y} = [x_2 \ x_4 \cdots x_{2k} \ \ x_{2k+2}]$, then for $f_3(x_1,\cdots x_m) = [x_2 \ x_4 \cdots x_{2k} \ \ x_{2k+2}][\, x_3x_1 \cdots x_{2k-1}x_{2k+1}] = x_3x_2 \oplus x_4x_1 \oplus \cdots \oplus x_{2k+2}x_{2k+1}$ show that $W_{f_3}^k = 2^{2k+1} - 2^k$. It is easy to observe that,

$$f_3(x_1,\cdots x_{2k+2}) = f_2(x_1,\cdots x_{2k}) \oplus x_{2k+2}x_{2k+1}. \tag{5.23}$$

Then the (k+1)-Walsh transform of f_3 is,

$$W_{f_3}^{k+1}(\mathbf{w}) = \sum_{\mathbf{x}\in GF(2)^{2k+2}} (-1)^{<\mathbf{x},\mathbf{w}>_{t,k+1}} (-1)^{f_3(\mathbf{x})}$$

$$= \sum_{\mathbf{x}\in GF(2)^{2k+2}} (-1)^{<\mathbf{x},\mathbf{w}>_{t,k+1}} (-1)^{f_2(\mathbf{x})\oplus x_{2k+2}x_{2k+1}}$$, which is then equal to

$$W_{f_3}^{k+1}(\mathbf{w}) = \sum_{\mathbf{x}\in GF(2)^{2k+2}, x_{2k+2}=0, x_{2k+1}=0} (-1)^{<\mathbf{x},\mathbf{w}>_{t,k+1}} (-1)^{f_2(\mathbf{x})}$$
$$+ \sum_{\mathbf{x}\in GF(2)^{2k+2}, x_{2k+2}\neq 0, x_{2k+1}\neq 0} (-1)^{<\mathbf{x},\mathbf{w}>_{t,k+1}} (-1)^{f_2(\mathbf{x})\oplus x_{2k+2}x_{2k+1}}.$$

The first term on the right hand side of the above equation is k-Walsh transform of f_2.

$$W_{f_3}^{k+1}(\mathbf{w}) = W_{f_2}^{k}(\mathbf{w}) + \sum_{\mathbf{x}\in GF(2)^{2k+2}, x_{2k+2}x_{2k+1}=0} (-1)^{<\mathbf{x},\mathbf{w}>_{t,k+1}} (-1)^{f_2(\mathbf{x})}$$
$$+ \sum_{\mathbf{x}\in GF(2)^{2k+2}, x_{2k+2}x_{2k+1}\neq 0} (-1)^{<\mathbf{x},\mathbf{w}>_{t,k+1}} (-1)^{f_2(\mathbf{x})\oplus 1}.$$

This is then equal to

$$W_{f_3}^{k+1}(\mathbf{w}) = W_{f_2}^{k}(\mathbf{w}) + \sum_{\mathbf{x}\in GF(2)^{2k+2}, x_{2k+2}x_{2k+1}=0} (-1)^{<\mathbf{x},\mathbf{w}>_{t,k+1}} (-1)^{f_2(\mathbf{x})}$$
$$- \sum_{\mathbf{x}\in GF(2)^{2k+2}, x_{2k+2}x_{2k+1}\neq 0} (-1)^{<\mathbf{x},\mathbf{w}>_{t,k+1}} (-1)^{f_2(\mathbf{x})}.$$

Since f_2 is $2k$-variable k-bent function, $W_{f_3}^{k+1}(\mathbf{w}) = 2W_{f_2}^{k}(\mathbf{w}) + 2^{2k}$, which then gives $W_{f_3}^{k} = 2^{2k+1} - 2^{k}$ completing the proof.

Example 5.5: For m=4,

$f_1(x_1, x_2, x_3, x_4) = x_1x_3 \oplus x_2x_4$ is a Maiorana McFarland class bent function and also Tokareva's 2-bent function.

For m=6,

$f_1(x_1, x_2, x_3, x_4, x_5, x_6) = x_1x_2 \oplus x_3x_4 \oplus x_5x_6$, and

$f_1(x_1, x_2, x_3, x_4, x_5, x_6) = x_1x_2 \oplus x_3x_4 \oplus x_5x_6$, are Maiorana McFarland class bent functions and also Tokareva's 3-bent functions.

For m=8,

$f_1(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8) = x_1x_2 \oplus x_3x_4 \oplus x_5x_6 \oplus x_7x_8$, and

$f_1(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8) = x_4x_5 \oplus x_3x_6 \oplus x_1x_7 \oplus x_2x_8$, and

$f_1(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8) = x_2x_5 \oplus x_1x_6 \oplus x_4x_7 \oplus x_3x_8$ are Maiorana McFarland class bent functions and also Tokareva's 4-bent functions.

For m=10,

$f_1(x_1, x_2, \cdots, x_9, x_{10}) = x_1x_2 \oplus x_3x_4 \oplus x_5x_6 \oplus x_7x_8 \oplus x_9x_{10}$, and

$f_1(x_1, x_2, \cdots, x_8, x_9, x_{10}) = x_2x_6 \oplus x_1x_7 \oplus x_3x_8 \oplus x_4x_9 \oplus x_5x_{10}$, and

$f_1(x_1, x_2, \cdots, x_9, x_{10}) = x_2x_6 \oplus x_1x_7 \oplus x_4x_8 \oplus x_3x_9 \oplus x_5x_{10}$ are Maiorana McFarland class bent functions and also Tokareva's 5-bent functions.

5.4 Affine Equivalence Analysis of our *t,k*-bent Functions and Maiorana McFarland Class Bent Functions

Next, we will show that our *t,k*-bent functions are affine equivalent to the well-known Maiorana McFarland class of bent functions in Proposition 5.7.

Proposition 5.7: *t,k*-bent functions are affine equivalent to the Maiorana McFarland class of bent functions . Maiorana McFarland class bent functions

$f(\mathbf{x},\mathbf{y}) = <\mathbf{y}, \pi(\mathbf{x})> \oplus g(\mathbf{x})$ with the permutation $\pi_8(\mathbf{x})$ and $g(\mathbf{x}) = 0$ and the notation that $(x_{3i-2}, x_{3i-1}, x_{3i})$ is the i^{th} pair, with $1 \le i \le m/3$, such that,

(i) Permutations of different pairs, or

(ii) the permutation (x_{3i-2}, x_{3i-1}) on the i^{th} pair

Result in our (*m*/3),*k*-bent functions with $k = m \mod 3$.

Proof: For *t*=0 the proof follows from Proposition 5.6. We will prove by induction on *t*. Assume (5.25) is true for *t*, then show that it is true for *t*+1.

Assume for *m*=3*t*, that $f_4(\mathbf{x},\mathbf{y}) = <\mathbf{y}, \pi_8(\mathbf{x})>$ is *t*,*k*-bent. Then show that for *m*=3*t*+3, that $f_5(\mathbf{x},\mathbf{y}) = <\mathbf{y}, \pi_8(\mathbf{x})>$ is (*t*+1),*k*-bent.

For *m*=3*t*, and *k=0*, take $\mathbf{x} = [x_1\ x_3 \cdots x_{2k-1}]$ and $\mathbf{y} = [x_2\ x_4 \cdots x_{2k}]$, then assume $f_4(x_1, \cdots x_m) = [x_2\ x_4 \cdots x_{3t}][x_3\ x_1 \cdots x_{3t-1}] = x_3x_2 \oplus x_4x_1 \oplus \cdots \oplus x_{3t}x_{3t-1}$ is *t*,*k*-bent with $W_{f_4}^{t,k} = 2^{3t-1} - 2^{(3t-2)/2}$.

Then for *m*=3*k*+6, take $\mathbf{x} = [x_1\ x_3 \cdots x_{3t+1}\ x_{3t+3} x_{3t+5}]$ and $\mathbf{y} = [x_2\ x_4 \cdots x_{3t+2}\ x_{3t+4}\ x_{3t+6}]$, then for $\pi(\mathbf{x}) = (1, 2)$, which is one permutation which obeys Proposition 5.7,

$$\begin{aligned} f_5(x_1, \cdots x_m) &= [x_2\ x_4 \cdots x_{3t+2}\ x_{3t+4}\ x_{3t+6}]\ [x_3\ x_1 \cdots x_{3t+1}\ x_{3t+3} x_{3t+5}] \\ &= x_3x_2 \oplus x_4x_1 \oplus \cdots \oplus x_{3t+1}x_{3t+2} \oplus x_{3t+3}x_{3t+4} \oplus x_{3t+5}x_{3t+6} \end{aligned}$$

show that $W_{f_5}^{k} = 2^{3t+5} - 2^{(3t+4)/2}$. It is easy to observe that,

$$\begin{aligned} f_5(x_1, \cdots x_{3t+3}) = {} & f_4(x_1, \cdots x_{3t+2}) \oplus x_{3t+1}x_{3t+2} \oplus x_{3t+3}x_{3t+4} \\ & \oplus x_{3t+5}x_{3t+6} \end{aligned} . \quad (5.24)$$

Similar steps as in the proof of Proposition 5.6 gives $W_{f_5}^{k} = 2^{3t+5} - 2^{(3t+4)/2}$. This proves Proposition 5.7 only for one permutation, $\pi(\mathbf{x}) = (1, 2)$. Similar steps for all possible permutations given by Proposition 5.7, need to be proven. It seems they require similar steps as the above proof.

Example 5.6: For *m*=6,

$f_1(x_1, x_2, x_3, x_4, x_5, x_6) = x_1x_4 \oplus x_2x_5 \oplus x_3x_6$, and

$f_1(x_1, x_2, x_3, x_4, x_5, x_6) = x_2x_4 \oplus x_1x_5 \oplus x_3x_6$, are Maiorana McFarland class bent functions and also our 2,0-bent function.

For *m*=8,

$f_1(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8) = x_1x_5 \oplus x_2x_6 \oplus x_3x_7 \oplus x_4x_8$, and

$f_1(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8) = x_2x_5 \oplus x_1x_6 \oplus x_3x_7 \oplus x_4x_8$ are Maiorana McFarland class bent functions and also and also our 2,0-bent function,

$f_1(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8) = x_1x_6 \oplus x_1x_7 \oplus x_3x_8 \oplus x_4x_9 \oplus x_5x_{10}$, and

$f_1(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8) = x_2x_6 \oplus x_1x_7 \oplus x_3x_8 \oplus x_5x_9 \oplus x_4x_{10}$ are Maiorana McFarland class bent functions and also Tokareva's 3,0-bent functions.

5.5 Cubic Cryptanalysis

A cryptographic system consists of three basic components, namely the plaintext,which is the input to the system, the ciphertext, which is the output and the key. Cryptanalysis try to break the cryptosystem by finding the relation between these three components. Linear cryptanalysis tries to aproximate this relation by linear equations. It was proposed by M. Matsui [79] in 1993.Similarly the quadratic cryptanalysis tries to aproximate the relation between plantext, ciphertext and the key by quadratic equations, whose degree is at most 2. It was proposed by Tokareva [108] in 2008. She used *k*-affine and *k*-bent function definitions for extending linear cryptanalysis to quadratic cryptanalysis. She applied her method to S-boxes of well-known ciphers, such as GOST, DES and s^3DES and showed that quadratic equations have higher probability than linear equations have to define these cryptosystems.

As a cryptology application of our *t,k*-bent and *t,k*-affine functions, we introduce the method of cubic cryptanalysis for block ciphers. We call this new method as cubic cryptanalysis according to the main idea of it: to use (linear, quadratic and cubic) Boolean functions from $\psi_m^{t,k}$ for aproximations. In our method we approximate Boolean functions by *t,k*- affine functions. The newly introduced *t,k*-bent functions are claimed to be strong against cubic cryptanalysis, since they are as far as possible to *t,k*-affine functions, which are composed of affine, quadratic and cubic functions.

We introduce a generalization of the Matsui's algorithm for the one key bit determination. Our algorithm is based on the equality,

$$< a, \alpha(P) >_{t_1,k_1} \oplus < b, \gamma(C) >_{t_2,k_2} = < c, \sigma(K) >_{t_3,k_3} \tag{5.27}$$

where P is the plaintext (cryptosystem input), C is the ciphertext (cryptosystem output) and K is the key. Integers satisfy $0 \le t_1, t_2 \le m/3$, $0 \le t_3 \le m_{key}/3$, $0 \le k_1 \le (m-3t_1)/2$, $0 \le k_2 \le (m-3t_2)/2$ and $0 \le k_3 \le (m_{key}-3t_3)/3$. Here

m is the even length of plaintext and ciphertext, m_{key} is even length of the key. $F : Z_2^m x Z_2^{m_{key}} \to Z_2^m$ is a one-to-one transform if we fix the second argument.

$$C = F(P, K) \tag{5.28}$$

$F_i : Z_2^m x Z_2^{m'_{key}} \to Z_2^m$ is a transform for the i^{th} round of ciphering, it is one-to-one if we fix the second argument. Here m'_{key} is the subkey for the i^{th} round.

Assume that (5.23) holds with probability $p = 1/2 + \varepsilon$ where $0 \le |\varepsilon| \le 1/2$. ε is called the bias of (5.23). Notice that if the parameters $t_i = 0, k_i = 1$ then the dependence of the corresponding block *P*, *C*, or *K* is linear. And if the parameters $t_i = 0, k_i = 2$ or $t_i = 1$ then the dependence of the corresponding block *P*, *C*, or *K* is quadratic. For all other cases the dependence is cubic.

Let us fix a key *K*. Consider the set of known pairs of plaintext and ciphertext.

$$\{P_s, C_s | s = 1 \cdots N\} \tag{5.29}$$

The algorithm (as in the linear case) is based on the principle of maximum likelihood. Steps of the algorithm are given below,

(i) Define $N_0 = \left| \{ t_s | < a, \alpha(P_s) >_{t_1,k_1} \oplus < b, \gamma(C_s) >_{t_2,k_2} = 0 \} \right|$.

(ii) Guess $< c, \sigma(K) >_{t_3,k_3} = \begin{cases} 0 & if \quad (N_0 - N/2) * \varepsilon > 0 \\ 1 & else \end{cases}$.

(iii) Try to find *K* using the correlation obtained.

Further analysis of cubic cryptanalysis is left for future study. Cubic cryptanalysis must be studided on S-boxes of well-known cryptosystems. Linear, quadratic and cubic cryptanalysis of these cryptosystems must be compared in the future research.

An m-bit input/m-bit output cryptosystem can be considered as an mxm S-box. Our claim is that, for a fixed key, we should use ($m/3$),0-bent functions as the m-variable component functions of F in order to have the guaranteed high resistance to the cubic cryptanalysis. We left the studies of the properties of strong Boolean functions against cubic cryptanalysis and affine equivalence analysis of these functions to the newly introduced t,k-bent functions for future research.

CHAPTER 6

CONCLUSION

In this dissertation, we have concentrated on basic Boolean function properties such as affine equivalence classes, covering sequences and bentness. We have also studied the Z_4 and Z_8-linear codes and using these codes, we have introduced a new class of bent Boolean functions, which we show to be affine equivalent to the well-known Maiorana McFarland class of bent functions. As a cryptological application, we have defined the method of cubic cryptanalysis for block ciphers and introduced *t,k*-bent functions, which we consider to be strong against cubic cryptanalysis.

6.1 Results

Firstly, in Chapter 3, we show that some covering sequences of a Boolean function can be obtained using the Walsh transform nulls. We prove that each null frequency of the Walsh transform defines one covering sequence; and if the Boolean function is balanced, each null is associated with two covering sequences. We present a lower bound for the number of covering sequences and confirm that the set of covering sequences that we find from Walsh transform nulls are distinct from those given by Carlet and Mesnager [39]. Relations between a Walsh transform null frequency and the associated covering sequence are as given in (3.14) and (3.15). We have shown that:

i) For an arbitrary *m*-variable Boolean function f, each nonzero Walsh transform null frequency $\mathbf{w} \in GF(2)^m$ defines a covering sequence $\boldsymbol{\lambda} \in \{1,-1\}$ with elements $\lambda_{\mathbf{a}} = (-1)^{<\mathbf{w},\mathbf{a}>}$, and for each covering sequence $\boldsymbol{\lambda}$ which can be represented as $\lambda_{\mathbf{a}} = (-1)^{<\mathbf{w},\mathbf{a}>}$, there exists a nonzero Walsh transform null **w**.

ii) For a balanced *m*-variable Boolean function f, each nonzero Walsh transform null frequency $\mathbf{w} \in GF(2)^m$ defines a covering sequence $\boldsymbol{\lambda} \in GF(2)^{2^m}$ with elements $\lambda_{\mathbf{a}} =< \mathbf{w}, \mathbf{a} >$, and for each covering sequence $\boldsymbol{\lambda}$ which can be represented as $\lambda_{\mathbf{a}} =< \mathbf{w}, \mathbf{a} >$, there exists a nonzero Walsh transform null **w**,

Hence one can obtain some of the covering sequences, at least as much as the number of Walsh transform nulls, using the Walsh transform null frequencies. It is proven that all the covering sequences calculated from Walsh transform null frequencies through equation (3.15) are linearly independent and none of them can be an indicator of a subspace. Starting from this point, we come to the conclusion that, the set of covering sequences that can be calculated from Proposition 3.2 of Carlet and Mesnager [39] and our Theorem 3.3 are distinct. We have also obtained a relation between covering sequences of affine equivalent Boolean functions and proven that if a function *f* does not have any covering sequence, any other function affinely equivalent to *f* does not have a covering sequence either. Moreover, we also show that numbers of covering sequences of affine equivalent Boolean functions do not have to be equal.

Secondly, in Chapter 4, we examine Tokareva's studies [104-108] on Z_4-linear codes. We discuss and give the origins of *k*-affine functions and *k*-dot product definitions of Tokareva in Section 4.2. In Proposition 4.2, we show that the Krotov matrices $\mathbf{A}^{k,(m-2k)}$, which are used to construct Z_4-linear Hadamard like codes, have the lexicographically ordered codewords of the Z_4-linear (2^m, m) code C, as columns.

We define the quadratic terms in the algebraic normal forms of *k*-affine functions in Proposition 4.5. Then Section 4.4 contains our contributions on the extension of Tokareva's definitions to a larger ring, Z_8. For this objective, we derive a new class of functions, which we call *t*,*k*-affine, using linear codes over the ring Z_8. We then state propositions 4.7 to 4.11, where Proposition 4.7 gives the properties of the $\mathbf{C}_\mathbf{m}^{\mathbf{t,k}}$ matrix, Proposition 4.8 shows that for *t*=0, *k*-affine and *t*,*k*-affine functions are exactly the same, which then implies Proposition 4.9 saying that *k*-dot product and *t*,*k*-dot product values are equivalent for *t*=0. Proposition 4.10 gives the properties and Proposition 4.11 gives the explicit formula of the *t*,*k*-dot product. The set of *t*,*k*-affine functions contain affine functions, and some of the quadratic and cubic functions. Examples of these functions are given at the end of Chapter 4 starting from Z_8-linear codes.

Finally in Chapter 5, we study bent functions, which are at maximum distance to affine functions (Rothaus and Dillon), particularly Maiorana McFarland bent construction. We review Tokareva's *k*-bent functions [104-108] and extend her work by defining the *t*,*k*-Walsh transform and *t*,*k*-nonlinearity. We give Propositon 5.3 to show that the *t*,*k*-Walsh transform of a Boolean function satisfies the Parseval's equation; and then relate the *t*,*k*-nonlinearity to *t*,*k*-Walsh transform in Propositon 5.4. Next, we suggest the new class of bent functions, namely the *t*,*k*-bent functions, which depend upon the *t*,*k*-dot product definition given in Chapter 4. We state Proposition 5.5 to show that the set of (*t*+1),*k*-bent functions and

t,(k+1)-bent functions are subsets of the set of t,k-bent functions. In sections 5.3 and 5.4, we show that these new classes, namely Tokareva's k-bent and our t,k-bent functions, are affine equivalent to the well-known Maiorana McFarland class of bent functions. As a cryptological application, we define the method of cubic cryptanalysis for block ciphers in section 5.5, following Matsui's work on linear cryptanalysis. We conjecture that for a fixed key, one should try to use (m/3),k-bent functions as the m-variable component functions of the S-boxes in order to have higher resistance to cubic cryptanalysis.

6.2 Summary of Results and Directions for Future Research

Main results of this thesis can be summarized as follows. We have

1) proven that, each null frequency of the Walsh transform defines at least one covering sequence; however, the number of covering sequences is more than the number of Walsh transform nulls in general;

2) shown that the set of covering sequences which can be calculated from Proposition 3.2 of Carlet and Tarannikov and from our Theorem 3.3 are distinct;

3) obtained a relation between covering sequences of affine equivalent functions and proven that if a function does not have any covering sequence, then its affine equivalent function does not have any either, on the other hand, numbers of covering sequences of affine equivalent Boolean functions do not have to be equal;

4) defined a new class of functions, which we call t,k-affine, using linear codes over the ring Z_8; and given the explicit formula of the t,k-dot product and its properties;

5) defined the t,k-Walsh transform of a Boolean function and shown that it satisfies the Parseval's equation;

6) given the definition of t,k-nonlinearity and related it to t,k-Walsh transform;

7) suggested a new class of bent functions, the t,k-bent functions, which are extensions of k-bent functions and shown that they are affine equivalent to Maiorana McFarland class of bent functions.

Future studies can include the extension of such work to larger rings (or fields) and the search for codes, whose binary images are nonlinear and having better properties than the presently known ones. Suggested cubic cryptanalysis method can be applied to the known cryptosystems, compared with linear and quadratic cryptanalyses in terms of probability biases, and the correctness of our conjecture that "$(m/3),k$-bent functions are strong against cubic cryptanalysis" can be explored more extensively.

REFERENCES

[1] Agievich S., “On the Affine Classification of Cubic Bent Functions”, Cryptology ePrint Archive, Report 2005/044, 2005.

[2] Anderson R., “A5 the GSM Encryption Algorithm”, Scicrypt post, 1994.

[3] Armknecht F. and Krause M., “Algebraic Attacks on Combiners with Memory”, Crypto 2003 (D. Boneh, ed.), Lecture Notes in Computer Science, Springer-Verlag, vol. 2729, pp. 162-175, 2003.

[4] Armknecht F., “A Linearization Attack on the Bluetooth Key Stream Generator”, ePrint Archive, Report /2002/191, presented at the 8th Estonian WinterSchool in Computer Science (EWSCS), 9 pages, 2002.

[5] Ashikmin A. and Barg A., “Minimal Vectors in Linear Codes”, IEEE Transactions on Information Theory IT-44 , no. 5, 2010-2017, 1998.

[6] Beauchamp K.G., “Applications of Walsh and Related Functions with an Introduction to Sequence Functions”, Microelectronics and Signal Processing, Academic Press, London New York, 1984.

[7] Berger T. P., Canteaut A., Charpin P. and Laigle-Chapuy Y., “On Almost Perfect Nonlinear Mappings”, In Proceedings 2005 IEEE International Symposium on Information Theory, ISIT 05, ISBN: 0-7803-9151-9, pp. 705-782, Adelaide, Australia, 2005.

[8] Berman S. D. and Grushko I., “B-functions Encountered in Modular Codes”, Problemy Perdachi Informatsii, pp.10-18, 1981.

[9] Beth T. and Ding C., “On Almost Perfect Nonlinear Permutations”, In Advances in Cryptology EUROCRYPT’93, LNCS, vol. 765, pp. 65-76, Springer-Verlag, 1993.

[10] Biham E. and Shamir A., "Differential Cryptanalysis of DES-like Crytptosystems," Journal of Cryptography, vol. 4, no. 1, pp. 3-72, 1991.

[11] Biryukov A., De Canniμere C., Braeken A. and Preneel B., “A Toolbox for Cryptanalysis: Linear and Affine Equivalence Algorithms”, Eurocrypt 2003 (E. Biham, ed.), Lecture Notes in Computer Science, vol. 2656, Springer-Verlag, pp. 33-55, 2003.

[12] Borges J., Phelps K. T., Rifa J. and Zinoviev V. A., “On Z_4-linear Preparata-like and Kerdock-like Codes”, IEEE Transactions on Information Theory, vol. 49, no.11, pp. 2834–2843, 2003.

[13] Borissov Y., Braeken A., Nikova S. and Preneel B., “On the Covering Radii of Binary Reed-Muller Codes in the Set of Resilient Boolean Functions”, IEEE Transactions on Information Theory IT-51, no. 3, pp.1182-1189, 2005.

[14] Boztas S., “Near-optimal 4ϕ (four-phase) Sequences and Optimal Binary Sequences for CDMA,” Ph.D. Thesis, Univ. of Southern California, Los Angeles, CA, 1990.

[15] Boztas S., Hammons R. and Kumar P. V., “4-Phase Sequences with Near-Optimum Correlation Properties”, ”, IEEE Transactions on Information Theory, vol. 38, no. 3, pp. 1101-1113, 1992.

[16] Braeken A. and Preneel B., “Probabilistic Algebraic Attacks”, 10th IMA International Conference on Cryptography and Coding (N. Smart, ed.), Lecture Notes in Computer Science, Springer-Verlag, 2005.

[17] Braeken A., “Fast Correlation Attacks on Stream Ciphers”, Technical Report, K.U. Leuven, http://www.esat.kuleuven.ac.be/~abiryuko/ Cryptan/lectures.html, 2002.

[18] Braeken A., Borissov Y. and Nikova S., “Classification of Cubic Boolean Functions in 7 Variables”, Twenty-Sixth Symposium on Information Theory in the Benelux (J. Cardinal, N. Cerf, O. Delgrange, and O. Markowitch, eds.), pp. 285-292, 2005.

[19] Braeken A., Borissov Y., Nikova S. and Preneel B., “Classification of Boolean Functions of 6 Variables or Less with respect to Cryptographic Properties”, International Colloquium on Automata, Languages and Programming ICALP 2005 (M. Yung, G.F. Italiano, and C. Palamidessi, eds.), Lecture Notes in Computer Science, vol. 3580, Springer-Verlag, pp. 324-334, 2005.

[20] Braeken A., Borissov Y., Nikova S. and Preneel B., “Classification of Cubic (n−4)-resilient Boolean Functions”, IEEE Transactions on Information Theory, 2005.

[21] Braeken A., Lano J. and Preneel B., “Evaluating the Resistance of Filter and Combiners Against Fast Algebraic Attacks”, Internal Report, 2005.

[22] Braeken A., Nikov V., Nikova S. and Preneel B., “On Boolean Functions with Generalized Cryptographic Properties”, Indocrypt 2004 (A. Canteaut and K. Viswanathan, eds.), Lecture Notes in Computer Science, vol. 3348, Springer-Verlag, pp. 120-135, 2004.

[23] Braeken A., Wolf C. and Preneel B., “Classification of Highly Nonlinear Boolean Power Functions with a Randomised Algorithm for Checking Normality”, Cryptology ePrint Archive, Report 2004/214, 2004.

[24] Brier E. and Langevin P., “Classification of Boolean Cubic Forms of Nine Variables”, IEEE Information Theory Workshop 2003 (J. Boutros and A. Gulliver, eds.), pp. 179-182, 2003.

[25] Braeken A.,”Cryptographic Properties of Boolean Functions and S-boxes”, Ph.D. Thesis, 2006.

[26] Brualdi R.A. and Pless V., “Orphans of the First order Reed-Muller Codes”, IEEE Transaction on Information Theory, no. 2, pp. 399-407, 1990.

[27] Calderbank A. R. and Kantor W. M., “The Geometry of Two-Weight Codes", Bull. London Math. Soc., vol. 118, pp. 97-12, 1986.

[28] Calderbank A. R., Cameron P. J., Kantor W. M. and Seidel J. J., “Z_4-Kerdock Codes, Orthogonal Spreads, and Extremal Euclidean Line-Sets”, Proc. London Math. Soc., vol. 75, pp. 436-480, 1997.

[29] Calderbank A. R., Hammons R., Kumar P. V., Sloane N. J. A. and Solé P., “A Linear Construction for Certain Kerdock and Preparata Codes”, Bull. Amer. Math. Soc., submitted, 1992.

[30] Calderbank A. R., Li W. and Poonen B., “A 2-adic Approach to the Analysis of Cyclic Codes", IEEE Transactions on Information Theory, vol. 43, pp. 977-986, 1997.

[31] Calderbank A. R., Rains E. M., Shor P. W. and Sloane N. J. A., “Quantum Error Correction and Orthogonal Geometry", Phys. Rev. Lett., 78, pp. 405-409, 1997.

[32] Calderbank A. R., Rains E. M., Shor P. W. and Sloane N. J. A., "Quantum Error Correction Via Codes Over GF(4)", IEEE Transactions on Information Theory, vol. 3, pp. 1-40, 1998.

[33] Canteaut A. and Charpin P., "Decomposing Bent Functions", IEEE Transactions on Information Theory, no. 8, pp. 2004-2019, 2003.

[34] Canteaut A. and Trabbia M., "Improved Fast Correlation Attacks Using Parity-Check Equations of Weight 4 and 5", Eurocrypt 2000 (B. Preneel, ed.), Lecture Notes in Computer Science, vol. 1807, Springer-Verlag, pp. 573-588, 2000.

[35] Canteaut A., "Differential Cryptanalysis of Feistel Ciphers and Differential Uniform Mappings", Selected Areas in Cryptography SAC 1997 (C. Adams and M. Just, eds.), Lecture Notes in Computer Science, vol. 1556, Springer-Verlag, pp. 172-184, 1997.

[36] Canteaut A., "Open Problems Related to Algebraic Attacks on Stream Ciphers", Proceedings of the 2005 International Workshop on Coding and Cryptography WCC 2005 (P. Charpin and O. Ytrehus, eds.), Lecture Notes in Computer Science, pp. 1-10, 2005.

[37] Canteaut A., Carlet C., Charpin P. and Fontaine C., "Propagation Characteristics and Correlation-Immunity of Highly Nonlinear Boolean Functions" In Advances in Cryptology, Eurocrypt 2000, pp. 507–522. Number 1807 in Lecture Notes in Computer Science, Springer-Verlag, Berlin, 2000.

[38] Canteaut A., Carlet C., Charpin P., and Fontaine C., "On cryptographic Properties of the Cosets of RM(1;m)", IEEE Transactions on Information Theory IT-47, no. 4, pp. 1494-1513, 2001.

[39] Carlet C. and Mesnager S., "On the supports of the Walsh transforms of Boolean functions", Proceedings of BFCA (First Workshop on Boolean Functions: Cryptography and Applications), 2005.

[40] Carlet C. and Tarannikov Y., “Covering Sequences of Boolean Functions and their Cryptographic Significance”, Designs, Codes and Cryptography, vol. 25, pp. 263-279, 2002.

[41] Carlet C., “Improving the Algebraic Immunity of Resilient and Nonlinear Functions and Constructing Bent Functions”, IEEE Transactions on Information Theory, pp. 420-426, 2000.

[42] Carlet C., “On the Complexity of Cryptographic Boolean Functions”, 6th Conference on Finite Fields and Applications (G.L. Mullen, H. Stichtenoth, and H. Tapia-Recillas, eds.), Springer-Verlag, pp. 53-69, 2001.

[43] Carlet C., “On the Secondary Constructions of Resilient and Bent Functions”, Progress in Computer Science and Applied Logic, pp. 3-28, 2004.

[44] Carlet C., “Two New Classes of Bent Functions”, Eurocrypt 1993 (T. Helleseth, ed.), Lecture Notes in Computer Science, vol. 950, Springer-Verlag, pp. 77-101, 1993.

[45] Carlet C., “Z_{2^k}-linear codes”, IEEE Transactions on Information Theory, vol. 44, no. 4, pp. 1543–1547, 1998.

[46] Charpin P., TietÄavÄainen A., and Zinoviev V., “Cyclic Codes with Minimum Distance d = 3”, Problems of Information Transmission, no. 4, pp. 287-296, 1997.

[47] Dawson E. and Wu C. H., “On the Linear Structures of Symmetric Boolean Functions, vol. 16, pp. 87-102, 1996.

[48] Denev J. D. and Tonchev V. D., “On the Number of Equivalence Classes of Boolean Functions Under a Transformation Group”, IEEE Transactions on Information Theory IT-26, no. 5, pp. 625-626, 1980.

[49] Dillon J., “A survey of bent functions”, Technical Report, NSA Technical Journal, pp. 191-215, 1972.

[50] Dillon J., “Elementary Hadamard Difference Sets”, Ph.D. Thesis, University of Maryland, 1974.

[51] Dobbertin H., “Construction of Bent Functions and Balanced Boolean Functions with High Nonlinearity”, Fast Software Encryption FSE 1994 (B. Preneel, ed.), Lecture Notes in Computer Science, vol. 1008, Springer-Verlag, pp. 61-74, 1994.

[52] Dobbertin H., “One-to-one Highly Nonlinear Power Functions on GF(2^n)”, Applicable Algebra in Engineering, Communication, and Computation 9 , pp. 139-152, 1998.

[53] Dobbertin H., “Ten Problems on Extremely Nonlinear Boolean Functions”, Technical Report, Dagstuhl, 1998.

[54] Evertse J. H., “Linear Structures in Block Ciphers”, Eurocrypt 1987 (D. Chaum and W. L. Price, eds.), Lecture Notes in Computer Science, vol. 304, Springer-Verlag, pp. 249-266, 1987.

[55] Forney G. D., Sloane N. J. A. and Trott M. D., “The Nordstrom-Robinson Code is the Binary Image of the Octacode”, Inform. Control, pp. 1-11, 2001.

[56] Fuller J. and Millan W., “Linear Redundancy in S-box”, Fast Software Encryption, pp. 74-86, 2003.

[57] Golic J. D., “Correlation via Linear Sequential Circuit Approximation of Combiners with Memory,” in Advances in Cryptology—Eurocrypt ’92 (Lecture Notes in Computer Science). Berlin, Germany: Springer-Verlag, vol. 658, pp. 113–123, 1993.

[58] Golic J. D., “Vectorial Boolean Functions and Induced Algebraic Equations”, IEEE Transactions on Information Theory, vol.52, no. 2, pp. 528-537, 2006.

[59] Hammons R., Kumar P. V., Calderbank A. R., Sloane N. J. A. and Solé P., “The Z_4-linearity of Kerdock, Preparata, Goethals and Related Codes”, preprint.

[60] Harrison M. A., “On the Classification of Boolean Functions by the General Linear and Affine Group”, Journal of the Society for industrial and applied mathematics, no. 3, pp.284-299, 1964.

[61] Hawkes P. and Rose G., “Rewriting Variables: The complexity of Fast Algebraic Attacks on Stream Ciphers”, Crypto 2004 (M. Franklin, ed.), Lecture Notes in Computer Science, vol. 3152, Springer-Verlag, pp. 390-406, 2004.

[62] Helleseth T. and Sandberg D., “Some Power Mappings with Low Differential Uniformity”. Applicable Algebra in Engineering, Communication and Computing, Springer-Verlag, vol.8, no. 5, pp. 363-370, 1997.

[63] Helleseth T. and Zinoviev V., “On Z_4-linear Goethals Codes and Kloosterman Sums”, Designs, Codes and Cryptography, vol. 17, pp. 268-288, 1999.

[64] Helleseth T., Kumar P. V. and Shanbhagl A. G., “New Codes with the Same Weight Distributions as the Goethals Codes and the Delsarte-Goethals Codes”,

Norwegian Research Council under Grant Numbers 107542/410 and 107623/420 and the National Science Foundation under Grant Number NCR-9016077.

[65] Helleseth T., Kumar P. V., “The Algebraic Decoding of the Z_4−linear Goethals Codes”, IEEE Transactions on Information Theory, vol. 41, pp. 2040–2048, 1995.

[66] Honda T., Satoh T., Iwata T., and Kurosawa K., “Probabilistic Higher Order Differential Attack and Higher Order Bent Functions”, Selected Areas in Cryptography SAC 1997 (C. Adams and M. Just, eds.), Lecture Notes in Computer Science, vol. 1556, Springer-Verlag, pp. 64-72, 1997.

[67] Hou X. D., “Covering Radius of the Reed-Muller Code R(1; 7) - a Simpler Proof”, Journal of Theoretical Theory, pp. 337-341, 1996.

[68] Hou X. D., “Cubic Bent Functions”, Discrete Mathematics, no. 1, pp. 149-161, 1998.

[69] Kasami T., “The Weight Enumerators for Several Classes of Subcodes of the Second Order Binary Reed-Muller Codes”, Information and Control, pp. 369-394, 1971.

[70] Kocher P., “Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems”, Crypto 1996 (N. Koblitz, ed.), Lecture Notes in Computer Science, vol. 1109, Springer-Verlag, pp. 104-113, 1996.

[71] Krotov D. S., “On Z_2^k-Dual Binary Codes”, available at http://arxiv.org/abs/0710.0198.

[72] Krotov D. S., “Z_4-linear Hadamard and Extended Perfect Codes”, Proc. of the Int. Workshop on Coding and Cryptography, pp. 329–334, 2001.

[73] Krotov D. S., "Z_4-linear Perfect Codes", Discrete Analysis and Operation Research, vol. 7, pp. 78-90, 2000 (in Russian). English translation is available at http://arxiv.org/abs/0710.0198.

[74] Kurosawa K., Iwata T. and Yoshiwara T., "New Covering Radius of Reed-Muller Codes for t-resilient Functions", Selected Areas in Cryptography SAC 2001 (S. Vaudenay and A.M. Youssef, eds.), Lecture Notes in Computer Science, vol. 2259, Springer-Verlag, pp. 75-86, 2001.

[75] Lai X., "Additive and Linear Structures of Cryptographic Functions", Fast Software Encryption FSE 1994 (B. Preneel, ed.), Lecture Notes in Computer Science, vol. 1008, Springer-Verlag, pp. 75-85, 1994.

[76] MacWilliams F. J. and Sloane N. J. A., "The Theory of Error-Correcting Codes", Elsevier Science Publisher, 1991.

[77] Maitra S. and Pasalic E., "Further Constructions of Resilient Boolean Functions with Very High Nonlinearity", IEEE Transactions on Information Theory IT-48 , pp.1825-1834, 2002.

[78] Maitra S. and Sarkar P., "Maximum Nonlinearity of Symmetric Boolean Functions on Odd Number of Variables", IEEE Transactions on Information Theory IT-48, no. 9, pp. 2626-2630, 2002.

[79] Matsui M., "Linear Cryptanalysis Method for DES Cipher", Eurocrypt 1993, vol. 765, pp. 386-397, 1994.

[80] McFarland R., "A Family of Noncyclic Difference Sets", Journal of Combinatorial Theory, 1965.

[81] McFarland R., “A Family of Noncyclic Difference Sets”, Journal of Combinatorial Theory, pp. 1-10, 1973.

[82] Meier W. and Staffelbach O., “Correlation Properties of Combiners with Memory in Stream Ciphers”, J. Cryptol., vol. 5, no. 1, pp. 67–86, 1992.

[83] Meier W. and Staffelbach O., “Nonlinearity Criteria for Cryptographic Cunctions”, Eurocrypt 1989 (J.-J. Quisquater and J. Vandewalle, eds.), Lecture Notes in Computer Science, vol. 434, Springer-Verlag, pp. 549-562, 1989.

[84] Meier W., Pasalic E., and Carlet C., “Algebraic Attacks and Decomposition of Boolean Functions”, Eurocrypt 2004 (C. Cachin and J. Camenisch, eds.), Lecture Notes in Computer Science, vol. 3027, Springer-Verlag, pp. 474-491, 2004.

[85] Nechaev A. A., “Kerdock Code in a Cyclic Form”, Discr. Mat. (USSR) 1, no. 4, pp. 123 -139 (in Russian). English translation: Discrete Math. and Appl., vol. 1, no. 4, pp. 365-384, 1999.

[86] Nordstrom A. W. and Robinson J. P., “An Optimum Nonlinear Code”, Inform. Control, vol.11, pp. 613-616, 1967.

[87] Pasalic E., “On Algebraic Immunity of Maiorana-McFarland Like Functions and Applications of Algebraic Attacks to Some Stream Cipher Schemes”, IEEE Transactions on Information Theory, pp. 241-246, 1998.

[88] Pasalic E., Maitra S., Johansson T. and Sarkar P., “New Constructions of Resilent and Correlation Immune Boolean Functions Achieving Upper Bounds on Nonlinearity”, In Proceedings of the Workshop on Cryptography and Coding Theory, Paris, 2001. Electronic Notes in Discrete Mathematics, Volume 6, Elsevier, Amsterdam, 2000.

[89] Pieprzyk J., “Bent Permutations”, Proceedings of First International Conference on Finite Fields, Coding Theory and Advances in Communication and Computing, 1992.

[90] Pless V., Solé P. and Qian Z., “Cyclic Self-dual Z_4−Codes”, Finite Fields and Applications, vol. 3, pp. 48-69, 1999.

[91] Preneel B., “Analysis and Design of Cryptographic Hash Functions”, Ph.D Thesis, KU Leuven (Belgium), 1993.

[92] Qu C., Sebery J. and Pieprzyk J., “Homegeneous Bent Functions”, 1999.

[93] Roth R. M. and Siegel P. H., “Lee-Metric BCH Codes and their Application to Constrained and Partial-Response Channels”, IEEE Transactions on Information Theory, vol. 40, no. 4, pp. 1083-1096, 1994.

[94] Rothaus O.S., “On Bent Functions”, Journal of Combinatorial Theory (A), pp. 300-305, 1976.

[95] Sarkar P. and Maitra S., “Cross-Correlation Analysis of Cryptographically Useful Boolean Function s and S-Boxes”, Theory of Comput. Systems, vol. 35, pp. 39–57, Springer-Verlag, 2002.

[96] Shannon C.E., “Communication Theory of Secrecy Systems”, Bell Systems Technical Journal, vol. 28, pp. 656–715, 1949.

[97] Siegenthaler T., “Correlation-Immunity of Nonlinear Combining Functions for Cryptographic Applications”, IEEE Transactions on Information Theory, IT-30(5): pp.776–780, September 1984.

[98] Snover S. L., “The Uniqueness of the Nordstrom-Robinson and the Golay Binary Codes”, Ph.D. Dissertation, Math. Dept., Michigan State Univ., 1973.

[99] Solé P., “A Quaternary Cyclic Code, and a Family of Quadriphase Sequences with Low Correlation Properties”, Springer Lecture Notes on Computer Science, vol. 388, pp. 193–201, 1988.

[100] Strazdins I., “Universal Affine Classification of Boolean Functions”, Acta Applicandae Mathematicae, pp.147-167, 1997.

[101] Tarannikov Y., “On Resilient Boolean Functions with Maximal Possible Nonlinearity”, Cryptology ePrint archive (http://eprint.iacr.org/), Report 2000/005 (2000), 18 pp.; Proceedings of Indocrypt 2000, Lecture Notes in Computer Science, vol. 1977, Springer-Verlag, pp. 19–30, 2000.

[102] Tarannikov Y., Korolev P. and Botev A., “Autocorrelation Coefficients and Correlation Immunity of Boolean Functions”, Eurocrypt 2000 (B. Preneel, ed.), Lecture Notes in Computer Science, vol. 1807, Springer-Verlag, pp. 460-479, 2000.

[103] Tilborg H., “On Weight in Codes”, Ph.D. Thesis, Technische Universiteit Eindhoven, 1976.

[104] Tokareva N. N., “Bent Functions with Stronger Nonlinear Properties: *k*-bent Functions”, vol. 14 of Discrete Analysis and Operation Research, no. 4, pp. 76-102, 2007. (in Russian). English translation will be available soon in Journal of Applied and Industrial Mathematics and at www.math.nsc.ru/»tokareva.

[105] Tokareva N. N., “*k*-Bent Functions and Quadratic Cryptanalysis of Block Ciphers”, BFCA'08, pp. 1-16, 2008.

[106] Tokareva N. N., “Method of Quadratic Cryptanalysis for Block Ciphers”, 2008. Submitted to Problems of Information Transmission.

[107] Tokareva N. N., “On k-bent Functions”, SIBECRYPT’2007, pp. 4–7, 2007.

[108] Tokareva N. N., “The Hierarchy of Classes of Bent Functions with Multiple Nonlinearity”, Sixth Scientific School on Discrete Mathematics and its Applications, Keldysh Institute of Applied Mathematics, pp. 16-20, 2007 (in Russian).

[109] Tsai C. and Sadowska M., “Boolean Functions Classification via Fixed Polarity Reed-Muller Forms”, IEEE Transactions on Computers, vol. 46, no. 2, pp. 173-186, 1997.

[110] Wolfmann J., “Binary Cyclic Codes which are Z_4-cyclic codes”, ISIT’2001, pp. 176-178, 2001.

[111] Wolfmann J., “Binary Images of Cyclic Codes over Z_4”, IEEE Transactions on Information Theory, vol. 47, no. 5, pp.1773-1779, 2001.

[112] Wolfmann J., “Negacyclic and cyclic codes over Z_4”, IEEE Transactions on Information Theory, vol. 45, pp. 2527-2532, 1999.

[113] Wolfmann J., “Z_4-version of the Binary Maiorana-McFarland Bent Functions”, IEEE Transactions on Information Theory, pp. 120-126, 1998.

[114] Xiao G. and Massey J. L., “A Spectral Characterization of Correlation-Immune Combining Functions”, IEEE Transactions on Information Theory, vol. 3, pp. 569–571, 1988.

[115] Yang K. and Helleseth T., “Kerdock Codes over Z_4 and Their Application to Designs”, ISlT 1998, pp. 398-399, 1998.

[116] Zhang M. and Chan A., “Maximum Correlation Analysis of Nonlinear S-boxes in Stream Ciphers”, In Advances in Cryptology - CRYPTO 2000, pp. 501–514, Springer-Verlag, Berlin, 2000.

[117] Zhang X. M. and Zheng Y., “GAC—the criterion for global Avalanche Characteristics of Cryptographic Functions”, Journal for Universal Computer Science, vol 5, pp. 316–333, 1995.

[118] Zheng Y. and Zhang X. M., “Relationships Between Bent Functions and Complementary Plateaued Functions”, International Conference on Information andCommunications Security (ICICS) 1999 (V. Varadharajan and Y. Mu, eds.), Lecture Notes in Computer Science, vol. 1726, pp. 60-75, 1999.

VITA

Güzin Yıldırım Kurnaz was born in Denizli, Turkey in 1975. She received her B.Sc. degree with Honors and M.Sc degree from Middle East Technical University (METU), Department of Electrics and Electronics Engineering in 1997 and 1999, respectively. She is working towards Ph.D. degree at METU, Department of Electrics and Electronics Engineering since 2001.

She worked as digital hardware design engineer from November 1996 to June 2002. She is working for TÜBİTAK-SAGE since 2006. Her areas of interest are digital hardware design and cryptology.

Her publications are:

Yıldırım G. and Yücel M.D., "Blok Şifreler için Doğrusal Olmama Ölçütü", ELEKO'2000 Elektrik-Elektronik-Bilgisayar Müh Sempozyumu Bildiriler Kitabı, Bursa, 2000.

Yıldırım G., "New Findings on Covering Sequences of Boolean Functions", ISC'2007, pp. 35-46, 2007.

Her papers that are submitted:

Yıldırım G. and Yücel M. D., "Relations Between Walsh Transform Null Frequencies and Covering Sequences", submitted to Designs, Codes and Cryptography.

Her papers to be submitted:

Yıldırım G. and Yücel M. D., “Covering sequences of Affine Equivalent Boolean Functions”, to be submitted to Designs, Codes and Cryptography.

Yıldırım G. and Yücel M. D., “A New Class of Bent Functions Depending on Z_8-linear Codes and its Affine Equivalency to Maiorana McFarland Class”, to be submitted to IEEE Transactions on Information Theory, 2009.

Master Thesis:

Yıldırım G., “A Novel Measure of Nonlinearity Criteria for Boolean Function”, Thesis supervisor: Melek Diker Yücel, Middle East Technical University, December 2000.