ON CRYPTOGRAPHICALLY SIGNIFICANT WEIGHTWISE (ALMOST) PERFECTLY BALANCED BOOLEAN FUNCTIONS

By

KRISHNA MALLICK

MATH11201904002

National Institute of Science Education and Research, Bhubaneswar

A thesis submitted to the

Board of Studies in Mathematical Sciences

In partial fulfillment of requirements

for the Degree of

DOCTOR OF PHILOSOPHY

of

HOMI BHABHA NATIONAL INSTITUTE

December, 2025

Homi Bhabha National Institute

Recommendations of the Viva Voce Committee

As members of the Viva Voce Committee, we certify that we have read the dissertation prepared by **Krishna Mallick** entitled **"On Cryptographically Significant Weightwise (Almost) Perfectly Balanced Boolean Functions"** and recommend that it may be accepted as fulfilling the thesis requirement for the award of Degree of Doctor of Philosophy.

S. No.	Doctoral Committee	Name	Signature	Date	InPerson/ Online
1.	Chairman	Prof. Binod Kumar Sahoo	bk Sahoo	16/12/25	In Person
2.	Guide/ Convener	Dr. Deepak Kumar Dalai	Dalai	16/12/25	In Person
3.	Co-Guide (if any)				
4.	Examiner	Prof. Sourav Mukhopadhyay	Sourav mukh	16/12/25	In Person
5.	Member 1	Dr. Kamal Lochan Patra	KLPatra	16/12/25	In Person
6.	Member 2	Dr. Aritra Banik		16/12/25	In Person
7.	Member 3	Prof. Santanu Sarkar	Santanu	Sarkar 16/12/2025	Online

Final approval and acceptance of this thesis is contingent upon the candidate's submission of the final copies of the thesis to HBNI.

I/We hereby certify that I/we have read this thesis prepared under my/our direction and recommend that it may be accepted as fulfilling the thesis requirement.

Date : 16/12/25

Place : NISER, Bhubaneswar

Signature
Co-guide (if any)

Dalai
Signature
Guide

STATEMENT BY AUTHOR

This dissertation has been submitted in partial fulfillment of requirements for an advanced degree at Homi Bhabha National Institute (HBNI) and is deposited in the Library to be made available to borrowers under rules of the HBNI.

Brief quotations from this dissertation are allowable without special permission, provided that accurate acknowledgement of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the Competent Authority of HBNI when in his or her judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

Krishna Mallick

DECLARATION

I, Krishna Mallick, declare that the research works presented in this thesis have been carried out by me. The work is original and has not been submitted earlier as a whole or in part for a degree / diploma at this or any other Institution/University. I authorize the National Institute of Science Education and Research (NISER) to lend this thesis to other institutions or individuals for the purpose of scholarly research.

Krishna Mallick

Krishna Mallick

CERTIFICATION ON ACADEMIC INTEGRITY

Undertaking by the Student

1. I **Krishna Mallick**, HBNI Enrollment Number **MATH11201904002** hereby undertake that the Thesis, titled " **On Cryptographically Significant Weightwise (Almost) Perfectly Balanced Boolean Functions**" is prepared by me and is the original work undertaken by me.
2. I also hereby undertake that this document has been duly checked through a plagiarism detection tool and the document is found to be plagiarism free as per the guidelines of the Institute/ UGC.
3. I am aware and undertake that if plagiarism is detected in my thesis at any stage in the future, suitable penalty will be imposed as per the guidelines of the Institute/ UGC.

Krishna Mallick
16/12/2025
Signature of the Student with date

Endorsed by the Thesis Supervisor:

I certify that the thesis written by the Researcher is plagiarism free as mentioned above by the student.

DK Dalai
16/12/2025

Signature of the Thesis Supervisor with date
Name : Deepak Kumar Dalai
Designation : Associate Professor
Department/ Centre : School of Mathematical sciences
Name of the CI/ OCC : NISER, Bhubaneswar

List of Publications arising from the thesis

Journal

1. A class of weightwise almost perfectly balanced Boolean functions, D. K. Dalai and K. Mallick, *Advances in Mathematics of Communications*, 2024, Vol 18(2): 480-504. doi: 10.3934/amc.2023048.

Conferences

1. Constructing WAPB Boolean functions from the direct sum of WAPB Boolean functions. D.K. Dalai and K. Mallick, *Progress in Cryptology- INDOCRYPT 2024*, Lecture Notes in Computer Science (LNCS), Vol 15495, page: 188-209, 2025, Springer. doi: https://doi.org/10.1007/978-3-031-80308-6-9.

2. On the direct sum of weightwise almost perfectly balanced Boolean functions, D.K. Dalai and K. Mallick, In *9th International Workshop on Boolean Functions and their Applications (BFA), 2024.*

3. A class of weightwise almost perfectly balanced Boolean functions with high weightwise nonlinearity, D.K. Dalai and K. Mallick, In *8th International Workshop on Boolean Functions and their Applications (BFA), 2023*, *Cryptology ePrint Archive, Paper 2024/422*, 2024.

4. A class of weightwise almost perfectly balanced Boolean functions, D.K. Dalai and K. Mallick, In *ALgebraic and combinatorial methods for COding and CRYPTography (ALCOCRYPT), 2023.*

Communicated

1. Weightwise almost perfectly balanced Boolean functions, construction from a permutation group action view, D.K. Dalai, K. Mallick and P. Meaux, *Cryptology ePrint Archive, Paper 2024/2068*, 2024.

Krishna Mallick

Krishna Mallick

Dedicated to . . .

I dedicate this thesis to the cherished memory of my beloved grandparents Jeje, Jejema, and Badabapa, whose unconditional love, blessings, and faith in me gave me the strength to persevere. I also fondly remember my Kaka, whose presence and affection were deeply meaningful to me. I lost all of them during my PhD, but their love and memories have continued to inspire me everyday of this journey. This work is a heartfelt tribute to their lasting influence on my life.

ACKNOWLEDGEMENTS

This thesis presents the work and results of my PhD journey in the School of Computer Sciences at the National Institute of Science Education and Research (NISER), Bhubaneswar. The journey started in 2019. These years have provided me with valuable opportunities to explore the field of cryptography and develop a deeper interest in cryptographic Boolean functions. This journey has been an enriched transformation in both academic and personal terms, made possible only through the generous support, motivation, and understanding of many individuals to whom I owe my deepest gratitude.

I would first like to express my sincere gratitude to my PhD supervisor, Dr. Deepak Kumar Dalai, for giving me the opportunity to pursue my doctoral research under his guidance. Throughout this journey, he has been an exceptional mentor for his patient, thoughtful guidance, and genuine willingness to help wherever he could. His constant support and guidance have not only shaped the direction of this thesis but also influenced my approach to research. In addition to his academic guidance, he played a crucial role in supporting me in many aspects during PhD and made the experience far more manageable. For all of this, I remain deeply grateful.

I am sincerely grateful to the members of my doctoral committee, Dr. Binod Kumar Sahoo, Dr. Kamal Lochan Patra, Dr. Santanu Sarkar, and Dr. Aritra Banik, for their valuable time, critical reviews, and suggestions throughout my thesis work. Their feedback has greatly contributed to improving the quality of this thesis.

I am also deeply grateful to all the faculty members of the School of Computer Sciences and the School of Mathematical Sciences, who taught me various courses during my coursework at NISER. Their teaching and guidance have laid a strong foundation and broadened my understanding of the field.

I extend my thanks to the administrative staff of NISER for their assistance during my Ph.D. tenure. I am grateful to NISER and the Department of Atomic Energy (DAE), Government of India, for providing the financial support through a fellowship for pursuing my PhD.

My sincere thanks to the Homi Bhabha National Institute, Mumbai, for awarding me an international travel grant to attend and present my research work at ALCOCRYPT-2023, held in Marseille, France. I also thank the Association for Computing Machinery (ACM) for the travel grant that helped me participate in BFA-2023 in Voss, Norway.

I would like to thank my collaborator, Dr. Pierrick Méaux for engaging in fruitful discussions and contributing to a meaningful collaborative research paper.

I am also grateful to my M.Sc. teachers, Dr. Pradeep K. Parida and Dr. Hrisikesh Mahato, for their support and for providing me with the recommendation letters that helped me in joining on this doctoral journey. I fondly remember Late Dr. B. B. Bhattacharya, whose inspiring teaching first sparked my curiosity and love for Cryptography.

I would like to extend my warmest thanks to my friends at NISER: Rasmita Apa, Swati, Annada, Rucha, Ankit, Susobhan, Anuvartika, Saswati, Rupkatha, and Pinky for making my PhD journey truly memorable. From sharing self-cooked meals to evening walks and countless cups of tea, their presence added joy to my PhD life. I am thankful to Annada once again for his support in helping me for Python and GitHub, which proved immensely helpful in my work.

I am truly thankful to my dearest friends Dinesh, Anjali, Shipra and Chiku for always being there to listen me through the most challenging phases of this journey. I am grateful to Dinesh, not just for his help with academic doubts, but for being a friend I could always count on.

Lastly and most importantly, I am deeply indebted to my family for their love and patience. I am deeply grateful to my Baba, whose constant support and encouragement words have uplifted me from the very beginning and carried me through every stage of this journey. I am also grateful to my siblings for their unconditional love and care.

Finally, I thank the Almighty for blessing me with strength throughout this journey.

ABSTRACT

Boolean functions play a fundamental role in the design of symmetric key primitives, particularly in stream ciphers, which have become significant in lightweight cryptographic applications due to their low computational complexity. In EUROCRYPT 2016, Méaux et al. introduced a stream cipher, FLIP, which is based on a filter permutator, where the input to the filtering Boolean function is restricted to constant Hamming weight vectors $x \in \mathbb{F}_2^n$. In this thesis, we focus on the construction of Weightwise Almost Perfectly Balanced (WAPB) and Weightwise Perfectly Balanced (WPB) Boolean functions, which exhibit (almost) balance over the sets $\mathsf{E}_{k,n} = \{x \in \mathbb{F}_2^n : \mathsf{w_H}(x) = k\}$ for all $0 \le k \le n$. These functions are of particular interest in the context of FLIP cipher framework.

The following provides a brief description of our work on the construction and analysis of the WAPB/WPB Boolean functions.

1. We present several constructions of WAPB Boolean functions based on Siegenthaler's method. We introduce a new class of WAPB Boolean functions, known as *complementary weightwise almost perfectly balanced (CWAPB)* Boolean functions, and identify the necessary and sufficient conditions under which the function is special WAPB (SWAPB) as defined by Gini and Méaux in their INDOCRYPT 2022 paper.

   Specifically, we propose a method for constructing a class of n variable WAPB functions by extending the support of a known n_0 variable WAPB Boolean function, where $n = n_0 2^m$ for some integer m, with n_0 being odd. This approach, combined with an elegant construction of WPB functions proposed by Mesnager and Su, gives a generalized framework for constructing WAPB functions that is applicable for arbitrary n.

2. For two Boolean functions $f : \mathbb{F}_2^m \to \mathbb{F}_2$ and $g : \mathbb{F}_2^n \to \mathbb{F}_2$, we define the direct sum $h(x, y) = f(x) + g(y)$ as a Boolean function over $\mathbb{F}_2^{m+n}$ for $x \in \mathbb{F}_2^m$ and $y \in \mathbb{F}_2^n$. We study the direct sum construction of WAPB and WPB Boolean functions and establish a general condition under which the direct sum h results in a WAPB/WPB Boolean function. Given f and g each being either WAPB or WPB, we investigate two cases under which h is WAPB or WPB. Our findings refine the earlier result by

Carlet et al. on the construction of WPB functions and compute the weight of h over $\mathsf{E}_{k,n}$ for $k \in [1, n-1]$ earlier proved by Zhu et al. that the direct sum of several WPB functions.

We propose a recursive construction of WPB functions based on direct sum and establish an upper bound to the algebraic immunity of the resulting functions. The constructed functions also exhibit high nonlinearity over $\mathbb{F}_2^n$. Furthermore, we define another subclass of WAPB functions called *alternating WAPB (AWAPB)*, which enable a recursive direct sum construction method for generating WAPB functions.

3. We propose a general construction method for a class of WAPB Boolean functions based on the action of a cyclic permutation group $P = \langle \pi \rangle$, where $\pi \in \mathbb{S}_n$ is a permutation on n elements, acting on $\mathbb{F}_2^n$. In particular, we studied the WAPB/WPB Boolean functions generated due to the action of two significant permutation groups, $\langle \psi \rangle$ and $\langle \sigma \rangle$, where ψ is a distinct binary-cycle permutation and σ is a rotation. When $n = 2^m$ for $m > 0$, a particular case of this construction is a WPB Boolean function in 2^m variables, proposed by Liu and Mesnager in Design, Codes and Cryptography, 2019. We evaluate the nonlinearity and weighted nonlinearities of the functions obtained from this construction, and as a result, the derived bounds improve upon those established by Liu and Mesnager. We theoretically analyze the cryptographic properties of the WAPB functions derived from these permutations and experimentally evaluate their nonlinearity parameters for n between 4 and 10.

Contents

Summary

Boolean functions are fundamental component to the design of symmetric key cryptographic primitives, both in stream ciphers and block ciphers. The cryptographic properties of Boolean functions have been defined over $\mathbb{F}_2^n$. However, in stream cipher FLIP, the input to the filtering function is restricted to $\mathsf{E}_{\frac{n}{2},n} \subseteq \mathbb{F}_2^n$, where $\mathsf{E}_{\frac{n}{2},n} = \{x \in \mathbb{F}_2^n : \mathsf{w}_\mathsf{H}(x) = \frac{n}{2}\}$. This shift needs the development and analysis of Boolean functions that exhibit strong cryptographic properties over such restricted domains.

This thesis focuses on the study and constructions of two classes of Boolean functions: Weightwise Perfectly Balanced (WPB) and Weightwise Almost Perfectly Balanced (WAPB) Boolean functions. A Boolean function is said to be WPB if it is balanced on $\mathsf{E}_{k,n} = \{x \in \mathbb{F}_2^n : \mathsf{w}_\mathsf{H}(x) = k\}$ for all $k \in [1, n-1]$, and the existence of such functions is limited to cases where n is a power of 2. Therefore, WAPB functions have been introduced in the literature that are almost balanced in each subset $\mathsf{E}_{k,n}$ for all $k \in [0, n-1]$. Since it is challenging in optimize all cryptographic criteria of Boolean function over $\mathbb{F}_2^n$, it is crucial to study trade offs especially when the function must be balanced or almost balanced on each subset $\mathsf{E}_{k,n}$ for all k.

We begin with the definitions of WPB and WAPB Boolean functions and then investigate several methods of constructing such functions with better cryptographic criteria such as their weightwise nonlinearity and weightwise algebraic immunity. We generalize WPB constructions by Mesnager and Su in [77], and Liu and Mesnager in [**?**] to arbitrary n. We introduce several secondary and recursive constructions based on Siegenthaler's construction over restricted domain, which are further extended to construct WAPB Boolean functions by lifting the support of lower dimensional WAPB functions. We discuss the cryptographic properties of such resultant functions.

A key contribution of this thesis is the use of group actions (particularly, cyclic subgroups of the symmetric group) to construct WAPB Boolean functions with strong nonlinearity and weightwise nonlinearity. We establish improved lower bounds for these parameters. Additionally, we study the direct sum of WAPB and WPB functions, initially

analyzed by Carlet et al. in [20] for WPB case. In this thesis, we identify those conditions under which the resulting function remain to be WAPB and WPB. Several propositions are provided to demonstrate balancedness, nonlinearity, and algebraic immunity in the direct sum framework.

Experimental results validate our theoretical findings and show that the constructed WAPB functions exhibit improved cryptographic properties compared to existing functions, especially for n not a power of 2. We provide a detailed comparison with known results and upper bounds for weightwise nonlinearity.

In conclusion, this thesis contributes both novel constructions for WAPB Boolean functions that are useful in lightweight cryptographic applications such as FLIP.

List of Figures

List of Tables

Chapter 1

Introduction

The research presented in this thesis focuses on the study of Boolean functions and their critical role in modern cryptography, particularly in the framework of symmetric key encryption. To understand the significance of Boolean functions in cryptographic design, it is important to consider how cryptography has evolved from its early origins as a tool for securing messages in puzzles and military communication to its current role in protecting digital information.

In this chapter, we provide a concise overview of the fundamental concepts of cryptography and trace its evolution into modern cryptographic practices. We introduce various cryptographic primitives and discuss their roles in achieving security objectives such as confidentiality, integrity, and authenticity.

Subsequently, we focus on stream ciphers, a class of symmetric key encryption schemes, by describing their general structure and also emphasizing the significant role of the Boolean function, which is a map from $\{0,1\}^n \to \{0,1\}$, in their design. Special attention is paid to the use of Boolean functions in constructing secure stream ciphers, particularly in the context of fully homomorphic encryption. Additionally, we highlight the importance of Boolean functions in the structure of block ciphers.

1.1 Motivation

Classical cryptography was initially developed to enable two individuals to communicate securely over insecure channels, ensuring that any eavesdropper capable of monitoring all communication between them would be unable to extract the original message, known as the *plaintext*, from the transmitted content, referred to as the *ciphertext*. Transformation of plaintext into ciphertext is called encryption and is ensured by a *cipher*. One of the earliest known ciphers was the Scytale cipher, used by the Spartans in ancient Greece in the 7th century BC. The famous Caesar cipher, named after the Roman general, politician Julius Caesar, was used in 46th BC to communicate with his generals during military campaigns securely. Cryptography was crucial during World War I and World War II in securing military communications and gathering intelligence. Notable examples include the German military's use of the Enigma [55] and Lorenz cipher machines [32]. In 1948, Claude Shannon laid the theoretical foundation through two landmark papers on information theory [Sh48] and cryptography [Sh49], both of which have a significant influence on modern cryptography and cryptanalysis. Since then, cryptography has evolved significantly with the advent of computers and digital systems. Now, it goes beyond secret communication to include advanced mathematical methods for securing digital data, protecting computational systems, and enabling secure distributed computing against adversarial threats.

The primary objective of cryptography is to protect information from unauthorized access, modification, and impersonation. This is achieved through the following three foundational security principles mentioned in [76, Chapter 1]:

- **Confidentiality:** It ensures that messages exchanged during communication can only be read by intended recipients. Even if an active or passive adversary, who has access to the transmission channel, is unable to derive any meaningful information about the content of messages exchanged between authorized recipients.

- **Integrity:** It ensures that the content of exchanged messages during a communication cannot be unauthorizedly modified by an active adversary who has access to the transmission channel. If an active adversary attempts to modify the content of the messages, such tampering can be detected.

- **Authenticity:** It ensures that the message truly originates from the person with whom we are communicating and that this person is indeed authorized for communication. In otherword, it prevents an adversary from impersonating a legitimate source of messages to any of the authentic communication partners.

In today's world, cryptography is indispensable, serving a crucial role in a wide range of activities such as protecting users' privacy on social networking platforms, enabling secure e-commerce transactions with credit cards, supporting digital currencies, preventing tampering of legal and financial documents, verifying software updates, and safeguarding private data stored in online platform.

Cryptographic systems rely on various fundamental building blocks known as cryptographic primitives. Among the most essential primitives are: *symmetric key encryption*, *asymmetric key encryption*. These primitives differ primarily in their use of encryption and decryption keys.

Asymmetric key encryption, also known as public key encryption, where encryption and decryption use two different keys. The receiver generates a pair of keys (pk, sk), where pk denotes the *public key* and sk the *secret key*. The public key pk is openly shared and used by any sender to encrypt messages. Only the receiver can decrypt the received ciphertext using its secret key sk. Hence, asymmetric key encryption enables multiple senders to communicate privately with a single receiver. In contrast, symmetric key encryption enables private communication only between two parties who share the same key for the communication in advance. The first ground-breaking work was by W. Diffie and M. Hellman in [33], for asymmetric key cryptography by introducing the concept of key exchange over a public channel.

Symmetric key encryption, also known as private key encryption, involves a single key that is shared between the communicating parties. This shared key is used for both encryption by the sender and decryption by the receiver. In 1883, Auguste Kerckhoffs, in [60], explicitly stated that a cipher needs to be secure as long as the encryption key remains secret, even if the adversary has complete knowledge of the cipher algorithm. This concept is now known as Kerckhoffs' principle and it asserts that the security of symmetric encryption relies solely on the confidentiality of the shared secret key. As a result, anyone who knows the key can compromise the confidentiality of the encryption scheme. Thus, the main challenge is *key exchange problem*, which arises from the fact that the sender and receiver need to agree on a secret key before establishing secure communication, but there is no secure channel available for this exchange.

In [94], Claude Shannon introduced the foundational concepts of confusion and diffusion, which have had a profound impact on the design and security of symmetric ciphers. There exist two classes of symmetric ciphers: *block ciphers* and *stream ciphers*, which are based on the process of encryption of plaintext. A *block cipher* is a specific type of symmetric encryption scheme that operates on fixed-size blocks of the plaintext. The plaintext message m is divided into blocks of size t (e.g., 64 or 128 bits), denoted as $m_1, m_2, \cdots$ and also known as a message block. If the message length is not a multiple of the block size t, then it is typically padded to fill the last message block. Each block is encrypted individually using a secret key k. Mathematically a block cipher encryption can be seen as: for a fixed k, and for each block $m_i \in \mathcal{X}$, the encryption function is defined as $f_k : \mathcal{X} \to \mathcal{X}$, which transform each block m_i to $f_k(m_i)$. Thus, f_k is bijective for a fixed k. Hence, at the receiver end, m can be recovered using the decryption function, which is the inverse of f_k. The block cipher literature is well developed, and one may refer to [62]. One of the earliest widely adopted block ciphers is *Data Encryption Standard (DES)* developed by IBM in the 1970s. However, by the 1990s, the 56-bit key size had become vulnerable to brute-force attacks. As a result, DES was replaced by more robust algorithms such as *Triple DES (3DES)* by Merkle and Hellman in 1981 and later in 2001, the *Advanced Encryption Standard (AES)* or, proposed as Rijndael [25] was adopted by NIST, which supports much larger key sizes (128, 192, or 256 bits) and block size of 128 bits, offering significantly stronger resistance against brute-force attacks. Other popular block ciphers are CAMELLIA [block size: 128 bits; key size: 128, 192 or, 256 bits] jointly developed by NTT and Mitsubishi Electric Corporation, RC6 [bock size: 128 bits; key size: 128, 192 or, 256 bits] [84], SERPENT [block size: 128 bit; key size: 128, 192 or, 256 bits] [9], CAST-256 [block size: 128 bits; key size: upto 256] [1], TWOFISH [block size: 128 bits; key size: 256] [91]. Most of the block ciphers employ a substitution-box, commonly referred to as an S-box, to perform substitution operations. An S-box takes an input of m-bits and produces an output of n-bits, which can formally described as (m, n)-Boolean function or a vectorial Boolean function. In block cipher design, a very popular structure is substitution-permutation networks (SPN), where the permutation boxes (or P-boxes or a (n, n)-Boolean functions) together with S-boxes are used to make the relation between plaintext and the ciphertext difficult to understand. It can be observed that an (m, n)-Boolean function consists of n individual Boolean functions, each taking m input bits. Specifically, for an input binary vector $(x_1, x_2, \cdots, x_m)$, the output is a binary vector $(y_1, y_2, \cdots, y_m)$ and the coordinates $y_1, y_2, \cdots, y_n$ are the outputs of Boolean functions evaluated over $(x_1, x_2, \cdots, x_m)$. For additional information on vectorial Boolean functions and their cryptographic criteria, refer to [19, Chapter 2].

Stream ciphers are a class of symmetric key encryption schemes that encrypt the plaintext messages one bit (or, a small block like 4-bits, a byte) at a time. Let $m_1, m_2, m_3 \cdots \in \mathcal{X}$ be the plaintext sequence, where $\mathcal{X}$ typically represents the binary set $\{0, 1\}$. A keystream $k_1, k_2, k_3, \cdots$ is generated from a short initial secret value k, referred as secret key, which is assumed to be securely shared beforehand between the communicating parties. For each

i, the encryption is performed by a function $f_{k_i} : \mathcal{X} \rightarrow \mathcal{X}$, defined as

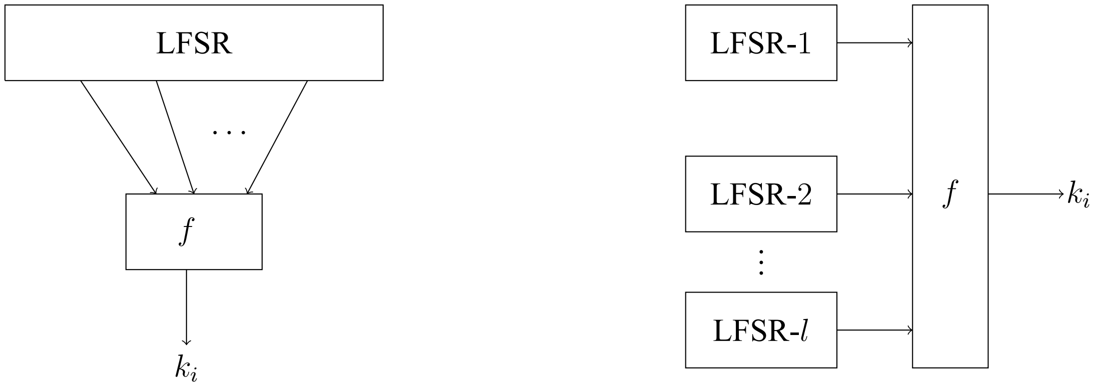
$$f_{k_i}(m_i) = m_i \oplus k_i$$

where $\oplus$ denotes the bitwise XOR (i.e., eXclusive OR or, addition $\mod 2$). Consequently, stream cipher takes the plaintext string $m_1, m_2, m_3 \cdots \in \mathcal{X}$ and outputs a ciphertext string $c_1, c_2, c_3, \cdots \in \mathcal{X}$, where $c_i = f_{k_i}(m_i)$. Similarly, decryption involves applying the inverse of f_{k_i} (here, f_{k_i} itself) to recover the plaintext string from the ciphertext string. The security of the stream cipher depends on the unpredictability of the keystream. A sequence is considered unpredictable, if it exhibits randomness. If the keystream $k_1, k_2, k_3, \cdots$ is a truly random bits, the resulting encryption scheme (using f_{k_i}) is known as *Vernam Cipher* or, *one time pad* [57, pp.32-34]. The one time pad achieves perfect secrecy, as defined in Shannon's work [94]. In particular, Shannon proved that a necessary condition for a symmetric-key encryption is that the key must be at least as long as the message. This condition establishes a lower bound on the key length to provide perfect secrecy. However, generating truly random bits is both practically difficult and computationally inefficient. To address such a challenge, *pseudorandom number generators (PRNGs)* are employed. A *PRNG* is a determistic algorithm that has one or more inputs called *seeds* or *initial vectors*, and it outputs a sequence of values known as *pseudorandom sequence*, that appear to be random according to specified statistical tests provided by NIST [88]. This sequence is as good as a random sequence, as long as we consider only a polynomial time observer. We call a PRNG a cryptographic PRNG if the output is unpredictable, given that the seed or initial vector is unknown. For further mathematical definitions one can refer [57, Chapter-2]. In a stream cipher, the keystream should behave like a pseudorandom sequence, which is XOR-ed with the plaintext sequence to produce the ciphertext. Hence, a major challenge in stream ciphers is keystream generation and secret key distribution. For example: SALSA20 [8] and ChaCha20 [7], are two stream ciphers that are currently considered secure and can also be adapted into cryptographic PRNG.

The primary motivation for dedicated stream ciphers is their efficiency, particularly in hardware-constrained environments where minimizing gate count and low power consumption is needed. There are different constructions of stream ciphers with their advantages and disadvantages in terms of complexity, speed, security, and hardware implementation. Stream ciphers are widely used in wireless communications, Internet of Things (IoT), and cloud computing, where lightweight and high-speed encryption are essential. For example, ChaCha20 [7] is used in TLS 1.3 [64], IPsec [81], and multiple other protocols. Similarly, E0 stream cipher [42] was used in the Bluetooth system, and SNOW 3G [35], ZUC [89] stream ciphers in the 3GPP encryption algorithm. In 2004, Europe launched the ECRYPT project called eSTREAM [85], which was solely dedicated to the development of stream ciphers. However, out of 34 proposed ciphers with diverse designs, 7 stream ciphers were selected based on two catagories: one for high speed software applications such as Salsa20/12 [8], SOSEMANUK [6], Rabbit [10], HC-128 [103] and other for hardware-constrained environment such as Grain-v1 [51], Trivium [15], Mickey-v2 [5]. For further informations regarding stream ciphers refer to [76, Chapter 6], [57, Chapter 6], [56].

There are several designs of stream ciphers: Linear-Feedback shift register (LFSR) based stream ciphers [61], Nonlinear-Feedback shift register (NFSR) based stream ciphers [61], block cipher based stream ciphers [87], sponge structure stream ciphers [56] and many more. LFSRs (figure-1.1) are one of the fundamental building blocks used in the design of stream ciphers, particularly for generating long pseudorandom sequences with good statistical properties. Most PRNGs consist of one or more bit-unit LFSRs, and to obtain a cryptographically secure keystream, the output of these LFSRs is either filtered or combined using a nonlinear Boolean function [19].

Several finalist ciphers from the eSTREAM project, such as the Grain family and Trivium, employ Boolean functions as either filter or combiner functions in their design. In this



Filter generator

Combiner generator

Figure 1.1: LFSR-Based Keystream Generators

context, a major challenge lies in constructing cryptographic Boolean functions that at the same time exhibit optimal properties such as balancedness, high algebraic degree, high nonlinearity, higher order correlation immunity, and strong algebraic immunity. These cryptographic criteria are essential to resist against various cryptographic attacks, including linear attack [102], algebraic attack [23], correlation attack [95], and fast algebraic attack [22], etc. Therefore, the analysis of such cryptographic Boolean functions remains a challenging and significant area of research in modern cryptography and information security.

In most existing studies, the Boolean functions used as filter functions in stream cipher constructions are assumed to be defined over the full input space $\{0,1\}^n$, without any restrictions on the input domain. The field of research concerning Boolean functions is both extensive and well-developed, yet it continues to present numerous open problems and unresolved questions that remain active areas of investigation. In the 1960s, Oscar Rothaus introduced a special class of Boolean functions known as *bent functions* [86], which achieve the maximum possible nonlinearity defined over $\{0,1\}^n$, even n. Patterson and Wiedemann [82] in 1983, provided a class of Boolean functions on 15 variables having nonlinearity greater than the quadratic bound (for odd n, discussed in Chapter-2, subsection 2.2.5). Hence, using [102], it is possible to show that for odd $n \geq 15$, it is possible to have a

Boolean functions having nonlinearity $2^{n-1} - 2^{\frac{n-1}{2}} + 20 \cdot 2^{\frac{n-15}{2}}$. So the question is for $9, 11$ and 13 though for odd $n \leq 7$, the maximum nonlinearity of a Boolean function attains the quadratic bound. Then, Kavut et [59] proved that there exist Boolean functions on odd n-variables having nonlinearity $> 2^{n-1} - 2^{\frac{n-1}{2}}$ if and only if $n > 7$.

A recently introduced stream cipher *FLIP*, which is also discussed in Section 2.3, is based on a filter permutator proposed in [75] in 2016. The structure of the cipher consists of a constant key register, a permutation generator, a pseudorandom number generator (PRNG), and a filter function, which can be seen in the figure 2.4.1. At initialization, a random short key, which is also served as a secret key, is loaded into the key register. In each cycle, the pseudorandom number generator (PRNG) generates a bit, according to which the pseudorandom permutation generates a bit permutation. This permutation is then applied to the key stored in the key register. The resulting permuted key bits are subsequently input to a nonlinear filtering function, which produces a single keystream bit as output. In [20], Méaux et al. proposed several instances of FLIP with the key lengths n of 530, 662, 1394 and 1704-bits, where the Hamming weight of the input to the function is $\frac{n}{2}$. The value of n is chosen to satisfy the security requirement $\binom{n}{\frac{n}{2}} \geq 2^{\lambda}$, where λ denotes the security parameter such that 2^{λ} is the number of operations required for a brute-force attack. From the perspective of the current analysis, the filter function used in this kind of structure needs further investigation.

This thesis is motivated by the underlying structure of the FLIP cipher, with a primary focus on the study of Boolean functions that are instrumental in the design of nonlinear filter functions in such a framework. As previously mentioned, it is crucial for any stream cipher to ensure that the keystream exhibits strong statistical properties, and this is why Boolean functions used in stream ciphers as filter or combiner functions are to be balanced. However, in the case of FLIP, the Hamming weight of all inputs to the filter function is fixed and equal to that of the secret key, as only bit permutations are applied to the key register during each cycle. Therefore, it is necessary to construct Boolean functions that are balanced on all subsets of $\{0,1\}^n$ consisting of vectors with fixed Hamming weights k, for $1 \leq k \leq n-1$. Such Boolean functions are known as *weightwise perfectly balanced(WPB)* or *weightwise almost perfectly balanced (WAPB)* depending on the form of n, whether $\binom{n}{k}$ is even or odd. Claude Carlet, Pierrick Méaux and Yann Rotella first introduced these terms in [20].

Several open questions still remain to be addressed.

Q1. [20] Determine the number of monomials in the algebraic normal form (ANF) as defined in definition 2.2.1, of a WPB Boolean function. This question can similarly be extended to the ANF of a WAPB Boolean function.

Q2. [20] Tightness in the upper bound of the weightwise nonlinearity $\mathsf{NL}_k(f)$, *i.e.* $\mathsf{NL}_k(f) \leq \frac{1}{2}\left[\binom{n}{k} - \sqrt{\binom{n}{k}}\right]$ for WPB or WAPB Boolean functions. The notion of tightness can only be explored for those values of n and k where the binomial coefficient $\binom{n}{k}$ is a perfect square. In other cases, it remains to be investigated whether the floor value of this expression provides a tight bound.

Q3. [19] Determine all nonquadratic bent functions whose restrictions to the set of binary vectors of length n and Hamming weight k have null nonlinearity.

1.2 Contribution of the Thesis

The primary contributions of this thesis are based on the study and analysis of Boolean functions with specific cryptographic properties relevant to the design of the filter function for stream ciphers such as FLIP [75]. Motivated by the structure of such ciphers, we focus on the construction and analysis of weightwise almost perfectly balanced (WAPB) and weightwise perfectly balanced (WPB) Boolean functions. The specific contributions of our work into the thesis are outlined as follows:

I. **A class of weightwise almost perfectly balanced Boolean functions [29].**

We introduce a new class of WAPB Boolean functions, known as *complementary weightwise almost perfectly balanced (CWAPB)* Boolean functions, and identify the conditions under which the function is special WAPB (SWAPB) defined in [44].

We also propose a construction method for a class of n-variable WAPB Boolean functions from the known support of an n_0-variable WAPB Boolean function where $n_0 < n$. This is a generalization of the construction of a WPB Boolean function by Mesnager and Su [77].

Furthermore, we present a modified class of WAPB Boolean functions derived from the above generalized construction. The modified functions exhibit significantly improved nonlinearity and weightwise nonlinearity.

II. **On some constructions of weightwise almost perfectly balanced Boolean functions [28].**

We present a construction on WAPB Boolean functions by perturbing the support vectors of a highly nonlinear function in the construction presented in [29]. The nonlinearity and weightwise nonlinearities of the modified functions improve substantially.

III. **Constructing WAPB Boolean Functions From the Direct Sum of WAPB Boolean Functions [30].**

We study the construction of WAPB and WPB Boolean functions by the direct sum defined in Definition 2.2.9 of two WAPB/WPB Boolean functions. A general result in this direction is established. We have presented some cases when the direct sum results in a WAPB/WPB Boolean function.

Several new constructions of WAPB/WPB Boolean functions are introduced in this context. Some results on the direct sum of WAPB/WPB Boolean functions presented by Carlet et al. [20] and Zhu et al. [106] are direct consequences of our findings.

IV. **Weightwise Almost Perfectly Balanced Functions, Construction From A Permutation Group Action View [31].**

We explore two significant permutation groups, $\langle\psi\rangle$ and $\langle\sigma\rangle$, where ψ is a distinct binary-cycle permutation and σ is a rotation. We propose a general method to construct a class of WAPB Boolean functions using the action of a cyclic permutation group on $\mathbb{F}_2^n$. This class generalizes the Weightwise Perfectly Balanced (WPB) Boolean function construction by Liu and Mesnager [66] to any n. We theoretically analyze the cryptographic properties of the WAPB functions derived from these permutations and experimentally evaluate their nonlinearity parameters for n between 4 and 10. We obtain a lower bound on nonlinearity and weightwise nonlinearities for the constructions and as a result, the derived bounds improve upon those established in [66].

1.3 Organization of the Thesis

1. In *Chapter* 2, we discuss the necessary definitions and notation related to Boolean functions that are used throughout the thesis. Fundamental cryptographic criteria such as algebraic degree, balancedness, nonlinearity, and algebraic immunity are discussed. The chapter concludes with the design principles of a stream cipher FLIP.

2. In *Chapter* 3, we define the concept of weightwise perfectly balanced (WPB) and weightwise almost perfectly balanced (WAPB) Boolean functions. We revisit the cryptographic properties of Boolean functions when restricted to a set of constant Hamming weight vectors.

3. In *Chapter* 4, we present several fundamental results related to binary Krawchouk polynomials in the case where $q = 2$. Specifically, we compute the minimum value of the binary Krawchouk polynomial of a fixed degree k. Towards the end of the chapter, a graph theoretic importance of our findings are discussed.

4. In *Chapter* 5, we explore several secondary constructions of WAPB and WPB Boolean functions. A recursive approach is proposed to generate a class of WAPB Boolean functions from a known WAPB function. This construction combines the technique of Siegenthaler's method and the support of the WPB function proposed by Mesnager and Su [77]. We introduce a modified class of WAPB Boolean functions derived from the generalized construction, which exhibits improved nonlinearity and weightwise nonlinearities. The chapter concludes with experimental results and comparative tables for various values of n.

5. In *Chapter* 6, we establish a general expression that characterizes when the direct sum of two WAPB Boolean functions gives a WAPB Boolean function. Specifically, we identify two cases where direct sum results in either a WAPB or a WPB. Additionally, we propose a recursive construction based on direct sum to construct a WPB Boolean function and analyze its cryptographic properties, including nonlinearity and algebraic immunity.

6. In *Chapter* 7, we explore two significant permutation groups, $\langle\psi\rangle$ and $\langle\sigma\rangle$, where ψ is a distinct binary-cycle permutation and σ is a rotation as defined in Section 7.1.1. We theoretically analyze the cryptographic properties of the WAPB functions derived from these permutations and conduct experimental evaluations of their nonlinearity and weightwise nonlinearity for n ranging from 4 to 10. Both the class of WAPB Boolean functions derived from the permutation group cover the WPB Boolean functions by Liu and Mesnager in [66]. Furthermore, this chapter establishes improved

lower bounds on nonlinearity and k weightwise nonlinearities of functions.

7. In *Chapter* 8, we outline several open problems and future research directions that emerge from the findings in this thesis.

Chapter 2

Preliminaries

2.1 Introduction

Boolean functions are essential in both cryptography and error-correcting codes. In cryptographic applications, they serve as a core component in the design and analysis of symmetric-key primitives, including stream ciphers and block ciphers. This chapter presents the necessary definitions and notations related to Boolean functions that will be used throughout the thesis. We begin by defining a Boolean function and discussing various representations. Subsequently, we provide the definitions of the key cryptographic criteria, including algebraic degree, balancedness, nonlinearity, and algebraic immunity. Additionally, we highlight several results concerning these cryptographic properties and see how these properties work in specific classes of Boolean functions.

The binary field $\mathbb{F}_2$, also denoted $GF(2)$, is a finite field consisting of two elements 0 and 1. The operations defined over $\mathbb{F}_2$ are binary addition and binary multiplication, both defined by modulo 2 arithmetic. Specifically, Binary addition "$+$" corresponds to the XOR "$\oplus$" and binary multiplication "$\cdot$" corresponds to AND "$\wedge$" operations. Let $\mathbb{F}_2^n$ be the n-dimensional vector space over $\mathbb{F}_2$. We express every vector $x \in \mathbb{F}_2^n$ as per the standard basis of $\mathbb{F}_2^n$ and write a vector x as $(x_1, x_2, \cdots, x_n)$ in the usual coordinate form or as $x_1x_2 \ldots x_n$ in the binary string form. For any two vectors $x = (x_1, x_2, \cdots, x_n)$ and $y = (y_1, y_2, \cdots, y_n)$ in $\mathbb{F}_2^n$, the inner product usually denoted as "$\cdot$" is defined as $x \cdot y = x_1 \cdot y_1 + x_2 \cdot y_2 + \cdots + x_n \cdot y_n$ in $\mathbb{F}_2$. A natural total order on $\mathbb{F}_2^n$ is lexicographic ordering.

For any vectors $x = (x_1, x_2, \cdots, x_n)$ and $y = (y_1, y_2, \cdots, y_n)$ in $\mathbb{F}_2^n$, we say that $x < y$ is in lexicographic order, if there exists an index $k \in [1, n]$ such that $x_1 = y_1, x_2 = y_2, \ldots, x_{k-1} = y_{k-1}$ and $x_k < y_k$.

We denote $[i, j] = \{i, i+1, \ldots, j\}$ for two integers i, j with $i \leq j$. The notation, $0^n = (0, 0, \ldots, 0)$ and $1^n = (1, 1, \ldots, 1)$.

2.2 Boolean Function

A *Boolean function* in n-variable is a function from $\mathbb{F}_2^n$ to $\mathbb{F}_2$. The set of all n-variable Boolean functions is denoted as $\mathcal{B}_n$ and hence, the cardinality of $\mathcal{B}_n$ is 2^{2^n}. Here, n denotes the number of variables or, the input bits to the function f.

For positive integers n, m, an (n, m)*-function* or *vectorial Boolean function* is a function $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$. An (n, m)-function can be viewed as

$$F(x_1, x_2, \cdots, x_n) = (f_1(x_1, x_2, \cdots, x_n), f_2(x_1, x_2, \cdots, x_n), \cdots, f_m(x_1, x_2, \cdots, x_n)),$$

where each $f_i : \mathbb{F}_2^n \to \mathbb{F}_2$ is an n-variable Boolean function for $i \in [1, m]$. Similarly, a function from $\mathbb{F}_2^n$ to $\mathbb{R}$ is said to be a pseudo-Boolean function.

The *support* of vector $x \in \mathbb{F}_2^n$ is defined as the set $\mathsf{supp}(x) = \{i \in \{1, 2, \ldots, n\} : x_i \neq 0\}$. The *Hamming weight* of x, denoted as $\mathsf{w_H}(x)$, is the number of nonzero coordinates in x *i.e.*, $\mathsf{w_H}(x) = |\mathsf{supp}(x)|$. Similarly, the Hamming distance between two vectors x and y in $\mathbb{F}_2^n$, denoted as $\mathsf{d_H}(x, y)$, is defined as $\mathsf{d_H}(x, y) = |\{i \in [1, n] : x_i \neq y_i\}| = \mathsf{w_H}(x + y)$. A vector $x \in \mathbb{F}_2^n$ is called a true point of a Boolean function f if $f(x) = 1$. The *support of function* f is the set $\mathsf{supp}(f) = \{x \in \mathbb{F}_2^n : f(x) = 1\}$, and the Hamming weight of f, denoted as $\mathsf{w_H}(f)$ is given by $\mathsf{w_H}(f) = |\mathsf{supp}(f)|$. The *Hamming distance* between two functions f and g, denoted as $\mathsf{d_H}(f, g)$ is defined as

$$\mathsf{d_H}(f, g) = |\{x \in \mathbb{F}_2^n : f(x) \neq g(x)\}| = \mathsf{w_H}(f \oplus g),$$

where $f \oplus g$ denote the bitwise XOR of f and g.

x_1	x_2	x_3	$f(x)$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

Table 2.1: Truth Table representation of a function $f \in \mathcal{B}_3$.

2.2.1 Representations of Boolean functions

Boolean functions can be expressed in multiple equivalent forms. The following three representations are most commonly used in cryptography and coding theory.

Truth table: A classical representation of a Boolean function is its truth table, also known as the look-up table representation. This representation can be viewed as a binary string of length 2^n if the vectors of ordered by a total ordering of $\mathbb{F}_2^n$. We use lexicographical order for this purpose. Hence, the truth table representation $f \in \mathcal{B}_n$ is given by

$$f = (f(0, 0, \ldots, 0), f(0, 0, \ldots, 1), \ldots, f(1, 1, \ldots, 1)), \tag{2.1}$$

where inputs are ordered lexicographically. This representation of a Boolean function $f \in \mathcal{B}_n$ corresponds to a vector in $\mathbb{F}_2^{2^n}$.

Example 2.2.1. A truth table representation of a 3-variable Boolean function $f : \mathbb{F}_2^3 \to \mathbb{F}_2$ given in the Table 2.2.1 is 01101010 following the lexicographically ordering.

Algebraic normal form: The Algebraic normal form (in short, ANF) representation is an n-variable polynomial representation over $\mathbb{F}_2$ in the ring $\mathbb{F}_2[x_1, x_2, \ldots, x_n]/ < x_1^2+x_1, x_2^2+$

$x_2, \ldots, x_n^2 + x_n >$ of the form

$$\begin{aligned} f(x) &= \sum_{I \subseteq [1,n]} a_I \left(\prod_{i \in I} x_i \right) \\ &= a_0 + \sum_{i=1}^{n} a_i x_i + \sum_{1 \le i < j \le n} a_{i,j} x_i x_j + \ldots + a_{1,n} x_1 x_2 \cdots x_n. \end{aligned} \tag{2.2}$$

for $a_0, a_1, a_2, \cdots, a_{1,2,\cdots,n} \in \mathbb{F}_2$. The above representation in Equation (2.2) can also be written in the form $f(x) = \sum_{(u_1,u_2,\ldots,u_n) \in \mathbb{F}_2^n} a_u x_1^{u_1} x_2^{u_2} \cdots x_n^{u_n}$, where the coefficients a_u are from $\mathbb{F}_2$. As $x_i^2 = x_i$, every variable x_i in $f(x)$ appears with exponents either 0 or 1.

Any polynomial $p(x) \in \mathbb{F}_2[x_1, x_2, \ldots, x_n]/ < x_1^2 + x_1, x_2^2 + x_2, \ldots, x_n^2 + x_n >$ defined in Equation 2.2, defines a map from $\mathbb{F}_2^n$ to $\mathbb{F}_2$, hence a Boolean function. Since, there are 2^{2^n} polynomials in this ring, implies a 1-1 correspondance between $\mathcal{B}_n$ and $\mathbb{F}_2[x_1, x_2, \ldots, x_n]/ < x_1^2 + x_1, x_2^2 + x_2, \ldots, x_n^2 + x_n >$.

From a truth table of a Boolean function $f \in \mathcal{B}_n$, the ANF of f can be computed as

$$f(x) = \sum_{a \in \mathbb{F}_2^n} f(a)\delta_a(x) = \sum_{a \in \mathsf{supp}(f)} f(a)\delta_a(x) \tag{2.3}$$

where δ_a is the Kronecker symbol at $a \in \mathbb{F}_2^n$ and equals to $\delta_a(x) = \prod_{i=1}^{n}(x_i + a_i + 1)$.

Example 2.2.2. The ANF of the function in Example 2.2.1 can be computed using Equation 2.3 as follows. Here, $\mathsf{supp}(f) = \{001, 010, 100, 110\} \subseteq \mathbb{F}_2^3$. Then $\delta_a(x)$ for $a \in \mathsf{supp}(f)$ are $(x_1+1)(x_2+1)x_3$, $(x_1+1)x_2(x_3+1)$, $x_1(x_2+1)(x_3+1)$ and $x_1x_2(x_3+1)$, respectively. Therefore, the ANF of the function is $f(x_1, x_2, x_3) = (x_1+1)(x_2+1)x_3 + (x_1+1)x_2(x_3+1) + x_1(x_2+1)(x_3+1) + x_1x_2(x_3+1) = x_1x_2 + x_1 + x_2 + x_3$.

Trace Representation: Let $\mathbb{F}_{2^n}$, also denoted as $GF(2^n)$, be the finite field with 2^n elements, which is an extention of $\mathbb{F}_2$. For every n, such a field exists and is unique up to isomorphism. Since $\mathbb{F}_{2^n}$ is a n-dimentional vector space over $\mathbb{F}_2$, we can construct a vector space isomorphism ϕ from $\mathbb{F}_2^n$ to $\mathbb{F}_{2^n}$. For a vector $x = (x_1, x_2, \cdots, x_n)$ in $\mathbb{F}_2^n$, this map is defined by $\phi(x) = x_1\alpha_1 + x_2\alpha_2 + \cdots + x_n\alpha_n$, where $\{\alpha_1, \alpha_2, \cdots, \alpha_n\}$ is a $\mathbb{F}_2$-basis of $\mathbb{F}_{2^n}$. In particular, we choose $\{\alpha_1, \alpha_2, \cdots, \alpha_n\}$ as a normal basis $\{\alpha, \alpha^2, \cdots, \alpha^{2^{n-1}}\}$ for $\alpha \in \mathbb{F}_{2^n}$ and decompose $x = x_1\alpha + x_2\alpha^2 + \cdots + x_n\alpha^{2^{n-1}}$ for $(x_1, x_2, \cdots, x_n) \in \mathbb{F}_2^n$.

Definition 2.2.1. The cyclotomic classes of 2 modulo $2^n - 1$ are defined as

$$C(j) = \{j2^i \mod (2^n - 1) : i = 0, 1, \cdots, o(j) - 1\},$$

where $o(j)$ is the smallest positive integer such that $j2^{o(j)} \equiv j \mod (2^n - 1)$. It is clear that $o(j)$ is the size of the cyclotomic class containing j.

The smallest element in each cyclotomic class $C(j)$ is called the coset leader of the class.

Example 2.2.3. Let $n = 4$. Then for $j \in [0, 2^n - 2]$, the cyclotomic classes 2 modulo 15 are $C(0) = \{0\}$, $C(1) = \{1, 2, 4, 8\}$, $C(3) = \{3, 6, 12, 9\}$, $C(5) = \{5, 10\}$ and $C(7) = \{7, 14, 11, 13\}$.

Proposition 2.2.1. *The size of the cyclotomic classes $C(j)$ divides n, i.e. $o(j)|n$.*

Definition 2.2.2. Let $r|n$. For $\alpha \in \mathbb{F}_{2^n}$, the trace $Tr_r^n(\alpha)$ of α over $\mathbb{F}_{2^r}$ is defined by

$$Tr_r^n(\alpha) = \alpha + \alpha^{2^r} + \alpha^{2^{2r}} + \cdots + \alpha^{2^{n-r}}.$$

For $r = 1$, the trace map $Tr_1^n(\alpha) = \alpha + \alpha^2 + \alpha^{2^2} + \cdots + \alpha^{2^{n-1}}$ over $\mathbb{F}_{2^n}$ is called absolute trace map.

Proposition 2.2.2. *Every n-variable Boolean function f can be represented in the form*

$$f(x) = Tr_1^n(\sum_{i=0}^{2^n-1} \beta_i x^i), \tag{2.4}$$

where $\beta_i \in \mathbb{F}_{2^n}$.

The representation in Equation 2.4 can be transformed to the following representation. Let $\Gamma(n)$ be the set of all the coset leaders of the cyclotomic classes $C(j)$ of 2 modulo $2^n - 1$. Then,

$$f(x) = \sum_{j \in \Gamma(n)} Tr_1^{o(j)}(\beta_j x^j) + \beta_{2^n-1} x^{2^n-1} \tag{2.5}$$

where $\beta_j \in \mathbb{F}_{2^{o(j)}}$ for all $j \in \Gamma(n)$, $\beta_{2^n-1} \in \mathbb{F}_2$.

The representation in Equation 2.4 is called an *absolute trace representation* of f, and the representation in Equation 2.5 is called the *subfield trace representation* of f.

Proposition 2.2.3. *The set of all n-variable linear functions has the trace form $f(x) = Tr_1^n(ax)$ for a runs through the finite field $\mathbb{F}_{2^n}$.*

2.2.2 Algebraic degree

Definition 2.2.3. The *algebraic degree* (or simply, degree) of the ANF of a Boolean function, denoted by $\deg(f)$, is defined as

$$\deg(f) = \max_{I \subseteq [1,n]} \{|I| : a_I \neq 0\},$$

where a_I is the coefficient of the monomial $\prod_{i \in I} x_i$ in the ANF of f (see Equation 2.2). That is, the number of variables in the highest order monomial in the ANF with a nonzero coefficient.

Consequently, the $\deg(f)$ is well defined for every Boolean function due to the existence and uniqueness of the ANF of Boolean functions. For $f, g \in \mathcal{B}_n$, $\deg(f + g) \leq \max\{\deg(f), \deg(g)\}$. For example: $f(x_1, x_2, x_3) = x_1x_2x_3 + x_1x_3 + 1$ and $g(x_1, x_2, x_3) = x_1x_2x_3$. Here $\deg(f) = 3$ and $\deg(g) = 3$ but $\deg(f + g) = 2$. Similarly, $\deg(fg) \leq \deg(f) + \deg(g)$.

A Boolean function f is said to be *affine* if and only if $\deg(f) \leq 1$ *i.e.*, it can be expressed in the form $l_{a,b}(x) = a \cdot x + b$ where $a \in \mathbb{F}_2^n$ and $b \in \mathbb{F}_2$. If $b = 0$, the function is said to be *linear i.e.*$l_a(x) = a \cdot x$ for $a \in \mathbb{F}_2^n$. In particular, constant functions are of degree 0 and are therefore also considered affine. Let $\mathcal{A}_n$ be the set of all affine functions, then $|\mathcal{A}_n| = 2^{n+1}$.

2.2.3 Walsh-Hadamard Transform

Definition 2.2.4. The *Walsh-Hadamard transform* of any Boolean function $f \in \mathcal{B}_n$ is an integer-valued function $W_{\hat{f}} : \mathbb{F}_2^n \to \mathbb{Z}$ defined by

$$W_{\hat{f}}(a) = \sum_{x \in \mathbb{F}_2^n} f(x)(-1)^{a \cdot x} \tag{2.6}$$

where $a \cdot x = a_1x_1 + a_2x_2 + \cdots + a_nx_n$, the inner product in $\mathbb{F}_2^n$.

The function f can be recovered from the Walsh-Hadamard transform by the *inverse Walsh-Hadamard transform*

$$f(x) = \frac{1}{2^n} \sum_{w \in \mathbb{F}_2^n} W_{\hat{f}}(w)(-1)^{w \cdot x}. \tag{2.7}$$

Definition 2.2.5. The *Walsh transform* of $f \in \mathcal{B}_n$ is the function $W_f : \mathbb{F}_2^n \to \mathbb{Z}$ defined by

$$W_f(a) = \sum_{x \in \mathbb{F}_2^n} (-1)^{f(x)+a \cdot x} \tag{2.8}$$

We say, $W_f(a)$ for $a \in \mathbb{F}_2^n$ is the Walsh coefficient of f at a and $\{W_f(a) : a \in \mathbb{F}_2^n\}$ is the *Walsh spectrum* of f or spectral coefficients of f.

Example 2.2.4. The Walsh spectrum of f, presented in Example 2.2.1, is presented in the following table:

u	000	001	010	011	100	101	110	111
$W_f(a)$	0	0	0	0	-4	4	4	4

Table 2.2: Walsh spectrum of f in Example 2.2.1.

The following are some key identities on the Walsh transform of a function $f \in \mathcal{B}_n$.

Proposition 2.2.4. *Let $f \in \mathcal{B}_n$. Then*

1. $\sum_{a\in\mathbb{F}_2^n} W_f(a) = 2^n(-1)^{f(0)}$.
2. ***Parseval relation:*** $\displaystyle\sum_{a\in\mathbb{F}_2^n} W_f(a)^2 = 2^{2n}$.
3. ***Relation among Walsh-Hadamard transform and Walsh transform:***
$$W_f(a) = \begin{cases} 2^n - 2W_{\hat{f}}(a) & \textit{if } a = 0^n \\ -2W_{\hat{f}}(a) & \textit{if } a \neq 0^n. \end{cases}$$
4. $2^{\frac{n}{2}} \leq \max_{a\in\mathbb{F}_2^n} |W_f(a)| \leq 2^n$.

Proposition 2.2.5. *If $f \in \mathcal{B}_n$ then*

$$W_f(a) = \begin{cases} 2^n - 2\mathsf{w}_\mathsf{H}(f) & \textit{for } a = 0^n \\ 2^n - 2\mathsf{w}_\mathsf{H}(f + l_a) & \textit{for } a \neq 0^n \end{cases}$$

where $l_a(x) = a \cdot x$ for $x \in \mathbb{F}_2^n$.

Definition 2.2.6. The *bias* of an n-variable Boolean function f is defined by

$$Pr[f(x) = 1] = \frac{\mathsf{w}_\mathsf{H}(f)}{2^n}.$$

For $a \in \mathbb{F}_2^n$, the bias of the Boolean function $f(x) + a \cdot x$ is $Pr[f(x) + a \cdot x = 1] = \frac{\mathsf{w}_\mathsf{H}(f(x)+a\cdot x)}{2^n} = \frac{1}{2}\left(1 - \frac{W_f(a)}{2^n}\right)$. It can be observed that the smaller the value of $W_f(a)$ results the higher bias of $f(x) + a \cdot x$ and that implies the greater deviation of $f(x)$ from the linear function $l_a(x) = a \cdot x$.

2.2.4 Balancedness

A Boolean function $f \in \mathcal{B}_n$ is *balanced* if the Hamming weight $\mathsf{w}_\mathsf{H}(f) = 2^{n-1}$. This indicates that f outputs 1 for exactly half of the inputs in $\mathbb{F}_2^n$. In the design of stream ciphers, Boolean functions are often required to be balanced, to ensure a uniform distribution of 0 and 1 in the output. The total number of balanced Boolean functions in n variables is $\binom{2^n}{2^{n-1}}$.

Proposition 2.2.6. *1. If $f \in \mathcal{B}_n$ is balanced, then* $\deg(f) \leq n-1$.

2. For any $g \in \mathcal{B}_{n-1}$, $g(x_1, x_2, \ldots, x_{n-1}) + x_n$ in n variables is balanced.

3. Any non-constant affine function is balanced.

4. A Boolean function $f \in \mathcal{B}_n$ is balanced if and only if $W_f(0^n) = 0$.

2.2.5 Nonlinearity

Definition 2.2.7. The *nonlinearity* of $f \in \mathcal{B}_n$ is defined as

$$\mathsf{NL}(f) = \min_{l_a \in \mathcal{A}_n} \mathsf{d_H}(f, l_a)$$

where $\mathcal{A}_n = \{a \cdot x + b : a \in \mathbb{F}_2^n \text{ and } b \in \mathbb{F}_2\}$ is the set of all affine functions.

The nonlinearity of a Boolean function is one of the fundamental cryptographic properties. If a Boolean function is used in a cryptographic algorithm, it must have high nonlinearity to protect against the best affine approximation attacks [76]. In this subsection, we provide some results on the nonlinearity of Boolean functions.

Lemma 2.2.7. *Let $f \in \mathcal{B}_n$. Then, for $a \in \mathbb{F}_2^n$ and $b \in \mathbb{F}_2$,*

$$\mathsf{d_H}(f, l_{a,b}) = \mathsf{w_H}(f + l_{a,b}) = 2^{n-1} + (-1)^b \frac{1}{2} W_f(a).$$

From Definition 2.2.7 and Lemma 2.2.7, we have the relation between NL and $W_f(a)$ as follows.

Proposition 2.2.8. *For $f \in \mathcal{B}_n$,* $\mathsf{NL}(f) = 2^{n-1} - \frac{1}{2} \max_{a \in \mathbb{F}_2^n} |W_f(a)|$.

Hence, the nonlinearity of a Boolean function depends on the maximum absolute value of its Walsh spectrum. Using the results in Proposition 2.2.4, we have the following bounds on nonlinearity.

Proposition 2.2.9. *Let $f \in \mathcal{B}_n$. Then $0 \leq \mathsf{NL}(f) \leq 2^{n-1} - 2^{\frac{n}{2}-1}$.*

Definition 2.2.8. A Boolean function $f \in \mathcal{B}_n$ is called *bent* if the nonlinearity of f reaches the upper bound *i.e.*, $\mathsf{NL}(f) = 2^{n-1} - 2^{\frac{n}{2}-1}$.

Equivalently, it can be stated that a Boolean function $f \in \mathcal{B}_n$ is bent if and only if $|W_f(a)| = 2^{\frac{n}{2}}$ for all $a \in \mathbb{F}_2^n$. As $W_f(a)$ is an integer-valued function, a bent function exists only when n is even. There always exists a bent function for every even positive integer n; for example, $f(x) = x_1x_2 + x_3x_4 + \cdots + x_{n-1}x_n$ is a bent function [86].

Since $|W_f(0^n)| = 2^{\frac{n}{2}}$ for a bent Boolean function $f \in \mathcal{B}_n$, bent Boolean functions are not balanced, although they have the highest nonlinearity. The following result provides the upper and lower bounds for the balanced Boolean functions [69, 58, 70].

Proposition 2.2.10. *Let $f \in \mathcal{B}_n$, $(n \geq 3)$ be balanced. Then the nonlinearity of f is given by*

$$\mathsf{NL}(f) \leq \begin{cases} 2^{n-1} - 2^{\frac{n}{2}-1} - 2 & \text{if } n \text{ is even} \\ \lfloor\lfloor 2^{n-1} - 2^{\frac{n}{2}-1} \rfloor\rfloor & \text{if } n \text{ is odd} \end{cases}$$

where $\lfloor\lfloor x \rfloor\rfloor$ denotes the maximum even integer less than or equal to x.

Construction of nonlinear Boolean functions is often based on extending the number of variables from some known Boolean functions of a smaller number of variables. Proposition 2.2.11 and Proposition 2.2.12 present the nonlinearity of two simple ways of construction.

Proposition 2.2.11. *Let $f \in \mathcal{B}_n$ and $g \in \mathcal{B}_{n+1}$ such that $g(x, x_{n+1}) = f(x) + cx_{n+1}$, where $c \in \mathbb{F}_2, x \in \mathbb{F}_2^n$ and $x_{n+1} \in \mathbb{F}_2$. Then, $\mathsf{NL}(g) = 2\mathsf{NL}(f)$.*

Proposition 2.2.12. *Let $f, g \in \mathcal{B}_n$. Then the function $h \in \mathcal{B}_{n+1}$ defined as $h(x, x_{n+1}) = (1 + x_{n+1})f(x) + x_{n+1}g(x)$ where $x \in \mathbb{F}_2^n$ and $x_{n+1} \in \mathbb{F}_2$ has nonlinearity $\mathsf{NL}(h) \geq \mathsf{NL}(f) + \mathsf{NL}(g)$.*

In general, if $f, g \in \mathcal{B}_n$, then $\mathsf{NL}(f+g) \leq \mathsf{NL}(f) + \mathsf{NL}(g)$.

Definition 2.2.9. Let $f \in \mathcal{B}_n$ and $g \in \mathcal{B}_m$, the direct sum $h \in \mathcal{B}_{m+n}$ of f and g is defined as $h(x,y) = f(x) + g(y)$, for $x \in \mathbb{F}_2^n$ and $y \in \mathbb{F}_2^m$.

Proposition 2.2.13. *[92, 101] Let $h \in \mathcal{B}_{m+n}$ defined as $h(x,y) = f(x) + g(y)$ for $x \in \mathbb{F}_2^n$ and $y \in \mathbb{F}_2^m$. Then*

$$W_h(a,b) = W_f(a) \cdot W_g(b) \qquad \textit{and}$$

$$\mathsf{NL}(h) = 2^m\mathsf{NL}(f) + 2^n\mathsf{NL}(g) - 2\mathsf{NL}(f)\mathsf{NL}(g) > 2\mathsf{NL}(f)\mathsf{NL}(g).$$

Similarly, the following is the result on the nonlinearity of the product of two Boolean functions.

Proposition 2.2.14. *Let $h \in \mathcal{B}_{m+n}$ defined as $h(x,y) = f(x)g(y)$ for $x \in \mathbb{F}_2^n$ and $y \in \mathbb{F}_2^m$. Then $\mathsf{NL}(h) \geq \mathsf{NL}(f)\mathsf{NL}(g)$.*

Let n be odd. The equality of the upper bound in Proposition 2.2.9 is not achieved. Hence, $\mathsf{NL}(f) < 2^{n-1} - 2^{\frac{n}{2}-1}$ for n is odd. If $f \in \mathcal{B}_{n-1}$ is a quadratic bent function and let $g \in \mathcal{B}_n$ be defined as $g(x_1, x_2, \ldots, x_n) = f(x_1, x_2, \ldots, x_{n-1}) + x_n$, then $\mathsf{NL}(g) = 2\mathsf{NL}(f) = 2^{n-1} - 2^{\frac{n-1}{2}}$ (from Proposition 2.2.11). Patterson and Wiedemann [82, 102] provided a class of Boolean functions in $n \geq 15$ variables having nonlinearity $2^{n-1} - 2^{\frac{n-1}{2}} + 20 \cdot 2^{\frac{n-15}{2}}$, which is greater than the quadratic bound. Later, Kavut et al. [59] in 2007, proved that there exist Boolean functions in odd n variables having nonlinearity strictly greater than $2^{n-1} - 2^{\frac{n-1}{2}}$ if and only if $n > 7$. The following proposition provides a summary of the above discussion.

Proposition 2.2.15. *[54, 53, 80, 82] Let $f \in \mathcal{B}_n$. Then the following holds.*

i. If n is even, then $\max\{\mathsf{NL}(f) : f \in \mathcal{B}_n\} = 2^{n-1} - 2^{\frac{n}{2}-1}$.

ii. If $n = 3, 5, 7$, then $\max\{\mathsf{NL}(f) :; f \in \mathcal{B}_n\} = 2^{n-1} - 2^{\frac{n-1}{2}}$.

iii. If $n \geq 9$ and n is odd, then

$$2^{n-1} - 2^{\frac{n-1}{2}} < \max\{\mathsf{NL}(f) : f \in \mathcal{B}_n\} < 2\left\lfloor 2^{n-2} - 2^{\frac{n}{2}-2} \right\rfloor.$$

2.2.6 Algebraic Immunity

Algebraic immunity is a cryptographic property of a Boolean function that measures the resistance of a cryptographic system, particularly stream ciphers, using the Boolean function against some types of algebraic attacks. This concept was first introduced by Courtois in [22]. In this subsection, we define the algebraic immunity of the Boolean function and provide the bounds of the algebraic immunity.

Definition 2.2.10. Given $f \in \mathcal{B}_n$, a nonzero $g \in \mathcal{B}_n$ is called an annihilator of f if $fg = 0$, *i.e.*, $f(x)g(x) = 0$ for all $x \in \mathbb{F}_2^n$. The set of all annihilators of $f \in \mathcal{B}_n$ is denoted by $Ann(f)$.

The algebraic immunity of $f \in \mathcal{B}_n$ is defined as

$$\mathsf{AI}(f) = \min\{\deg(g) : g \in Ann(f) \cup Ann(1+f)\}.$$

Proposition 2.2.16. *[23] For any $f \in \mathcal{B}_n$, we have $\mathsf{AI}(f) \leq \lceil \frac{n}{2} \rceil$.*

Example 2.2.5. [27] Let $f \in \mathcal{B}_n$ be defined as

$$f(x) = \begin{cases} 1 & \text{if } \mathsf{w_H}(x) \leq \frac{n}{2} \\ 0 & \text{otherwise.} \end{cases}$$

Then $\mathsf{AI}(f) = \lceil \frac{n}{2} \rceil$ i.e., f has optimal algebraic immunity.

2.3 Symmetric Functions

Symmetric Boolean functions are a class of functions with the property that they are indistinguishable from different inputs with the same Hamming weight.

Definition 2.3.1. Let $f \in \mathcal{B}_n$. Then $f(x)$ is called a *symmetric Boolean function* if for every permutation $\pi \in \mathbb{S}_n$, we have

$$f(x_{\pi(1)}, x_{\pi(2)}, \dots, x_{\pi(n)}) = f(x_1, x_2, \dots, x_n).$$

The set of all symmetric Boolean functions in n variables is denoted by $\mathcal{S}_n$.

Let $\phi_t(x)$ for $t = 0, 1, \dots, n$, be the homogeneous symmetric functions where all monomials of degree t are present in the ANF of ϕ_t. These functions are also known as elementary symmetric functions in n variables. The ANF of ϕ_t are presented as

$$\begin{aligned}
\phi_0(x) =& 1,\\
\phi_1(x) =& x_1 + x_2 + \cdots + x_n,\\
\phi_2(x) =& x_1x_2 + x_1x_3 + \cdots + x_1x_n + x_2x_3 + x_2x_4 + \cdots + x_{n-1}x_n,\\
&\vdots\\
\phi_n(x) =& x_1x_2x_3 \cdots x_n.
\end{aligned}$$

Proposition 2.3.1. *The set of all n-variable symmetric Boolean functions, $\mathcal{S}_n$, forms a vector space of dimension $n+1$ over $\mathbb{F}_2$. Moreover, $\{\phi_t\}_{t=0}^{n}$ forms a basis on $\mathcal{S}_n$.*

The Walsh spectrum of a symmetric Boolean function $f \in \mathcal{S}_n$ can be computed as follows.

$$\begin{aligned}
W_f(a) &= \sum_{x \in \mathbb{F}_2^n} (-1)^{f(x)+a\cdot x}\\
&= \sum_{k=0}^{n} \sum_{\mathsf{w_H}(x)=k} (-1)^{f(x)+a\cdot x}\\
&= \sum_{k=0}^{n} (-1)^{\epsilon_i} \sum_{\mathsf{w_H}(x)=k} (-1)^{a\cdot x}\\
&= \sum_{k=0}^{n} (-1)^{\epsilon_k} \mathsf{K}_k(\mathsf{w_H}(a), n).
\end{aligned}$$

where

$$\mathsf{K}_k(x,n) = \sum_{j=0}^{k} (-1)^j \binom{x}{j} \binom{n-x}{k-j}$$

is the Krawchouk polynomial of degree k which is discussed in detail in Chapter 4. Savicky [90] proved that the only Boolean functions in n variables that are symmetric and bent have degree 2. The following result provides that there are only 4 distinct symmetric bent functions in n-variables of degree 2.

Proposition 2.3.2. *If $f \in \mathcal{B}_n$ is a symmetric bent function, then the ANF of f is*

$$f(x_1, x_2, \ldots, x_n) = \sum_{1 \le i < j \le n} x_i x_j + c \sum_{i=1}^{n} x_i + d$$

with $c, d \in \mathbb{F}_2$.

2.4 Symmetric Key Cryptography

A symmetric key encryption scheme [57] is defined by three functions: a probabilistic key generation function (`Gen`), encryption function (`Enc`), and decryption function (`Dec`) over a *key space* $\mathcal{K}$, a *message or plaintext space* $\mathcal{M}$ and a *ciphertext space* $\mathcal{C}$ along with

1. **Key generation** (`Gen`): The `Gen` is a probabilistic function that takes an integer n (security parameter) as input and outputs a key k from the key space $\mathcal{K}$ of length $l(n)$ using a distribution. Here, $l(n)$ is a polynomial on n.

2. **Encryption** (`Enc`): The encryption function $\texttt{Enc} : \mathcal{K} \times \mathcal{M} \to \mathcal{C}$, takes a key $k \in \mathcal{K}$ and a message $m \in \mathcal{M}$ as input, produces a ciphertext c, i.e., $\texttt{Enc}(k, m) = c$.

3. **Decryption** (`Dec`): The decryption function $\texttt{Dec} : \mathcal{K} \times \mathcal{C} \to \mathcal{M}$, takes a key $k \in \mathcal{K}$ and a ciphertext $c \in \mathcal{C}$ as input, outputs a plaintext m, i.e., $\texttt{Dec}(k, c) = m$.

A symmetric encryption scheme must satisfy the correctness criterion that for every key $k \in \mathcal{K}$ output by Gen and every message $m \in \mathcal{M}$, it holds that

$$\texttt{Dec}(k, \texttt{Enc}(k, m)) = m.$$

Stream Ciphers are a type of symmetric encryption that employs pseudorandom generators to expand a relatively short random key into a significantly longer pseudorandom sequence. The initial key is said as "seed" s of l-bit is randomly generated and secretly shared between the two communicating parties. The pseudorandom sequence is then XORed with the message to produce the ciphertext.

Mathematically, the string s is stretched using a probabilistic algorithm G that maps l-bit strings to L-bit strings for $l \leq L$, where L is a polynomial of l. For $s \in \{0, 1\}^l$ and $m, c \in \{0, 1\}^L$, encryption and decryption are defined as

$$Enc(s, m) = G(s) \oplus m \text{ and } Dec(s, c) = G(s) \oplus c.$$

The function G is called a pseudorandom generator [57, 12]. Figure 2.4 presents the general structure of the stream cipher.

2.4.1 Stream Cipher: FLIP

A family of stream ciphers named FLIP based on the filter generator construction introduced by Méaux et al. [75] in EUROCRYPT 2016 for the purpose of fully Homomorphic encryption applications. The following illustrates the general structure of filter permutators, as shown in Figure 2.4.1.

The FLIP cipher consists of three components.

- an n-bit key register to store the key,

- a bit permutation generator parametrized by a pseudorandom number generator

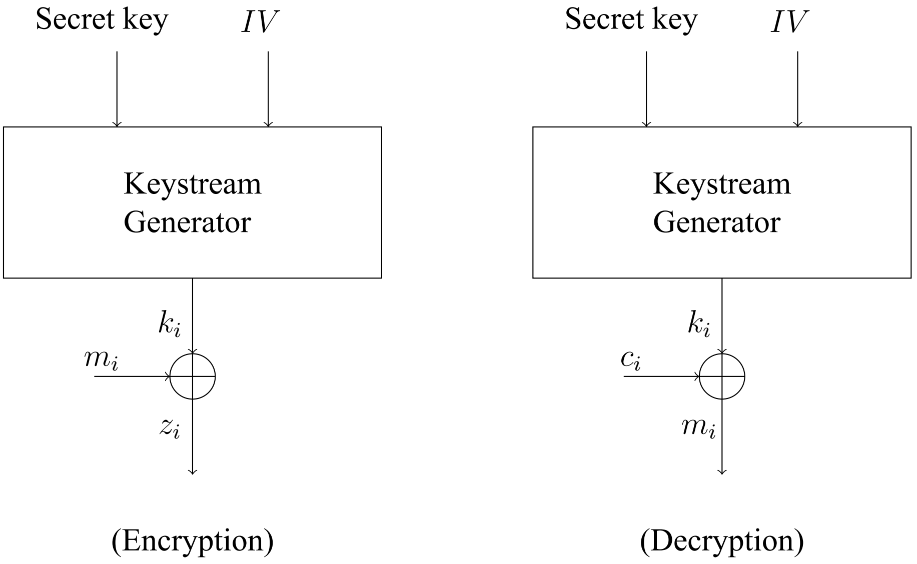


Figure 2.1: A general structure of stream cipher.

(PRNG) which is initialized with a public IV, which generates a permutation P_i on $\{1, 2, \ldots, n\}$ in each i-th cycle.

- a filtering function(n-variable Boolean function) f to generate keystream bit k_i in each i-th cycle. Then the keystream bit is XORed with the message bit m_i to generate the ciphertext c_i.

The filter function $f \in \mathcal{B}_n$, which is used in FLIP, is defined as follows:

$$\begin{aligned} f(x_1, x_2, \cdots, x_n) &= f_1(x_1, x_2, \cdots, x_{n_1}) + f_2(x_{n_1+1}, x_{n_1+2}, \cdots, x_{n_1+n_2}) \\ &\quad + f_3(x_{n_1+n_2+1}, x_{n_1+n_2+2}, \cdots, x_{n_1+n_2+n_3}) \end{aligned} \tag{2.9}$$

where $n = n_1 + n_2 + n_3$ and

- **Linear function:** $f_1 \in \mathcal{B}_{n_1}$ such that $f_1(x_1, x_2, \cdots, x_{n_1}) = \sum_{i=1}^{n_1} x_i$,

- **Quadratic function:** $f_2 \in \mathcal{B}_{n_2}$ (where n_2 is even) such that

$$f_2(x_1, x_2, \cdots, x_{n_2}) = \sum_{i=1}^{n_2-1} x_i x_{i+1},$$

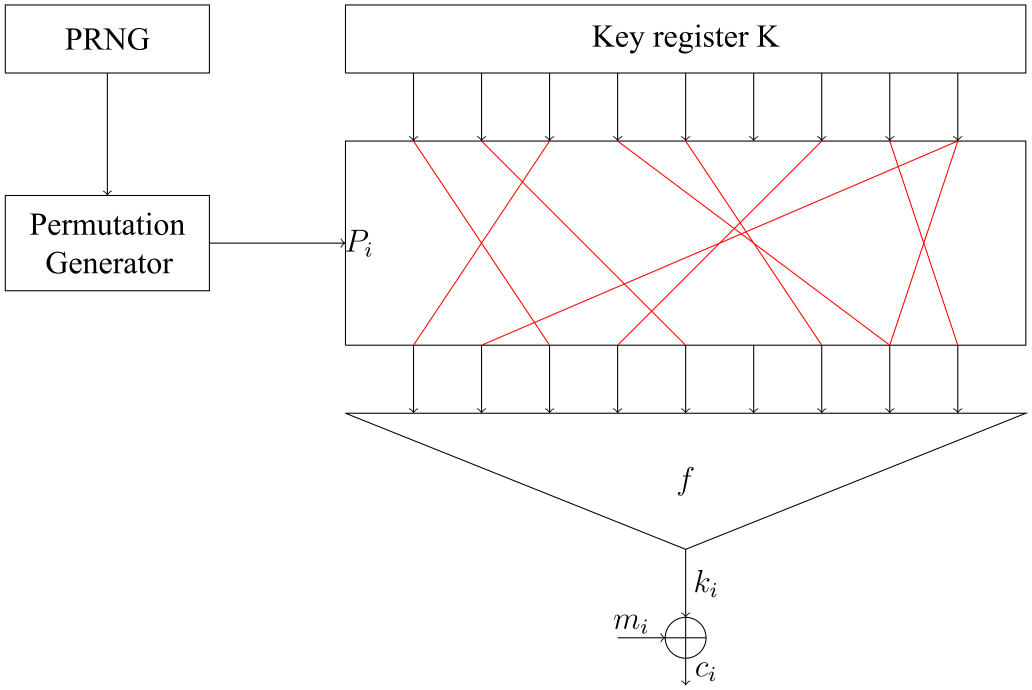


Figure 2.2: Structure of Filter permutator used in FLIP.

- **Triangular function:** $f_3 \in \mathcal{B}_{n_3}$ (where $n_3 = \frac{k(k+1)}{2}t$ for some positive integers k and t) such that

$$f_3(x_1, x_2, \cdots, x_{n_3}) = \sum_{j=1}^{t} T_k(x_{\frac{(j-1)k(k+1)}{2}+1}, x_{\frac{(j-1)k(k+1)}{2}+2}, \cdots, x_{\frac{(j-1)k(k+1)}{2}+\frac{k(k+1)}{2}}).$$

Here, T_k is the triangular function in $\frac{k(k+1)}{2}$ variables of degree k defined as

$$\begin{aligned} T_k(x_1, x_2, \cdots, x_{\frac{k(k+1)}{2}}) &= \sum_{i=1}^{k} \prod_{j=1}^{i} x_{j+\sum_{l=0}^{i-1} l} \\ &= x_1 + x_2x_3 + \ldots + x_{\frac{k(k-1)}{2}+1} x_{\frac{k(k-1)}{2}+2} \cdots x_{\frac{k(k-1)}{2}+k} \end{aligned}$$

That is, f_3 is the direct sum of t number of triangular functions of $\frac{k(k+1)}{2}$-variables.

The authors in [75] worked on the number of variables $n = n_1 + n_2 + n_3 \geq 500$. The table 2.4.1 presents the values of n, n_1, n_2, n_3 for different ciphers in the FLIP family as proposed in [75].

FLIP cipher	n	n_1	n_2	n_3	k	t
FLIP-530	530	42	128	360	9	8
FLIP-662	662	46	136	480	15	4
FLIP-1394	1394	82	224	1088	16	8
FLIP-1704	1704	86	238	1380	23	5

Table 2.3: Key parameters of ciphers of the FLIP Family

In this stream cipher, the keyspace $\mathcal{K}$ is restricted to the set of vectors of Hamming weight $\frac{n}{2}$ denote as $\mathsf{E}_{\frac{n}{2},n} \subset \mathbb{F}_2^n$ such that $\binom{n}{\frac{n}{2}} \geq 2^\lambda$, where λ (mostly ≥ 80) is the security parameter. As a result, the inputs to the filter function f are from the constant weight set $\mathsf{E}_{\frac{n}{2},n}$. To ensure the cryptographic security of any stream cipher, the generated keystream should look like a random sequence so that no polynomial time distinguisher $\mathcal{D}$ can distinguish the keystream from a random string. Hence, the output distribution of f must be uniform over $\mathsf{E}_{\frac{n}{2},n}$.

Let $\mathsf{E}_{k,n}$ be the set of all vectors in $\mathbb{F}_2^n$ of Hamming weight k, *i.e.*, $\mathsf{E}_{k,n} = \{x \in \mathbb{F}_2^n : \mathsf{w_H}(x) = k\}$. Extending the idea of FLIP's design, the researchers are interested in studying Boolean functions in the restricted domains $\mathsf{E}_{k,n}$ for $0 \leq k \leq n$ and trying to propose the constructions of Boolean functions that are cryptographically good in each restricted domain $\mathsf{E}_{k,n}$ for $0 \leq k \leq n$. For example, Boolean functions must be balanced, have good nonlinearity, and good algebraic immunity in each restricted domain $\mathsf{E}_{k,n}$ for $0 \leq k \leq n$. Additionally, a Boolean function having good cryptographic properties like balancedness and nonlinearity in each restricted domain $\mathsf{E}_{k,n}$ will also have good cryptographic properties in $\mathbb{F}_2^n$.

2.5 Conclusion

The restriction of the input to a set of constant Hamming weight vectors, as in the case of FLIP stream cipher, have introduced new challenges and opportunities in the design of Boolean functions with desirable cryptographic properties within this framework. The uniform distribution of the output over such $\mathsf{E}_{\frac{n}{2},n}$, is crucial to maintaining resistance against the polynomial time distinguisher and achieving pseudorandomness in the keystream. This has motivated a broader investigation into Boolean functions defined over $\mathsf{E}_{k,n}$ for all $k \in [0, n]$, which emphasizes in achieving key properties such as balancedness, high nonlinearity, and strong algebraic immunity. The development of such functions have enhanced the theoretical understanding of cryptographic primitives for practical advancements in lightweight and secure stream cipher designs.

Chapter 3

Cryptographic Analysis of Boolean Functions on Fixed Hamming weight

3.1 Introduction

Boolean functions serve as nonlinear components in stream ciphers. Classically, they appear in both combiner and filter-based pseudorandom generators. Grain family [51, 52, 50], Trivium [15] stream ciphers are some examples of such ciphers. In these cases, Boolean functions are typically evaluated with input taken from the entire vector space $\mathbb{F}_2^n$. The filter function f in FLIP stream cipher [75] is evaluated over a restricted input domain $\mathsf{E}_{\frac{n}{2},n} \subset \mathbb{F}_2^n$, which consists of all binary vectors of Hamming weight $\frac{n}{2}$. The FLIP cipher is a lightweight keystream generator which was introduced by Méaux et. al. in EUROCRYPT 2016.

Carlet et al. introduced the study of the cryptographic properties of Boolean functions restricted to subsets $\mathsf{E}_{k,n} = \{x \in \mathbb{F}_2^n : \mathsf{w_H}(x) = k\}$ for $k \in [0, n]$ in [20]. Several combinatorial studies on the restriction of Boolean functions to $\mathsf{E}_{k,n}$, which refer as slice of a Boolean cube $\{0, 1\}^n$, have been carried out by Filmus et al. in [36, 37, 41, 40, 39, 38]. Maitra et al. introduced another perspective on the study of Boolean functions in a restricted domain in [71] by considering a nonuniform probability distribution in the input domain of the Boolean function. In this chapter, we revisit the cryptographic properties of Boolean functions restricted to $\mathsf{E}_{k,n}$ for $k \in [0, n]$.

3.2 Boolean Functions over a restricted subset $\mathsf{E}_{k,n}$

We denote the set of vectors of weight k as $\mathsf{E}_{k,n} = \{x \in \mathbb{F}_2^n : \mathsf{w_H}(x) = k\}$ and therefore, $|\mathsf{E}_{k,n}| = \binom{n}{k}$. Let the support and Hamming weight of $f \in \mathcal{B}_n$ restricted to $\mathsf{E}_{k,n}$ defined as the set $\mathsf{supp}_k(f) = \{x \in \mathsf{E}_{k,n} : f(x) = 1\}$ and $\mathsf{w}_{k,n}(f) = |\mathsf{supp}_k(f)| = |\mathsf{supp}(f) \cap \mathsf{E}_{k,n}|$, respectively. The Hamming distance of two functions $f, g \in \mathcal{B}_n$ restricted to $\mathsf{E}_{k,n}$ is denoted as $d_{k,n}(f,g) = |\{x \in \mathsf{E}_{k,n} : f(x) \neq g(x)\}|$. The cryptographic criteria, such as balancedness, nonlinearity, and algebraic immunity of a function f restricted to $\mathsf{E}_{k,n}$, are defined as follows.

3.2.1 Balancedness

A key requirement for a Boolean function used in ciphers is that it should be balanced, or at least close to it. Therefore, we are interested in the Boolean functions that are balanced in the input set $\mathsf{E}_{k,n}$.

Definition 3.2.1. A Boolean function $f \in \mathcal{B}_n$ is said to be *weightwise almost perfectly balanced* (in short, *WAPB*) if for all $k \in [0, n]$,

$$\mathsf{w}_{k,n}(f) = \begin{cases} \frac{\binom{n}{k}}{2} & \textit{if } \binom{n}{k} \textit{ is even,} \\ \frac{\binom{n}{k} \pm 1}{2} & \textit{if } \binom{n}{k} \textit{ is odd.} \end{cases}$$

If $f \in \mathcal{B}_n$ is WAPB then we define $\delta_k^f \in \{-1, 0, 1\}$ for $k \in [0, n]$ as $\delta_k^f = 2\mathsf{w}_{k,n}(f) - \binom{n}{k}$. That is, $\mathsf{w}_{k,n}(f) = \frac{1}{2}\left[\binom{n}{k} + \delta_k^f\right]$.

By definition, a WAPB Boolean function is not necessarily balanced over $\mathbb{F}_2^n$. However, constructions typically aim to ensure that the function is also globally balanced over $\mathbb{F}_2^n$.

Definition 3.2.2. A Boolean function $f \in \mathcal{B}_n$ is said to be *weightwise perfectly balanced* (in short, *WPB*) if

$$\mathsf{w}_{k,n}(f) = \frac{\binom{n}{k}}{2}, \text{ for all } k \in [1, n-1],$$

and $f(0, 0, \ldots, 0) \neq f(1, 1, \ldots, 1)$.

If $f \in \mathcal{B}_n$ is WPB then $\delta_k^f = 0$ for $k \in [1, n-1]$ and $\delta_0^f = -\delta_n^f \neq 0$.

Let $x = (x_1, x_2, \ldots, x_n), y = (y_1, y_2, \ldots, y_n) \in \mathbb{F}_2^n$, then y *covers* x (*i.e.*, $x \preceq y$), if $x_i \leq y_i, \forall i \in [1, n]$.

Given a positive integer n, denote $e(n) = \{e_1, e_2, \ldots, e_w\} \subseteq \mathbb{N} \cup \{0\}$ if $n = 2^{e_1} + 2^{e_2} + \cdots + 2^{e_w}$. Therefore, for two positive integers m, n, we denote $m \preceq n$ if $e(m) \subseteq e(n)$. Further, for a set of non-negative integers T (i.e., $T \subseteq \mathbb{N} \cup \{0\}$), we define $2^T = \sum_{t \in T} 2^t$.

A remarkable theorem [67] was presented by Édouard Lucas in 1878 that provides a simple way to evaluate the binomial coefficient $\binom{n}{k}$ modulo a prime p. The same has been used to compute $\binom{n}{k}$ is odd or even.

Proposition 3.2.1 (Lucas' Theorem)**.** *Let the binary representation of n and k be given as $(n_1, n_2, \ldots, n_l)$ and $(k_1, k_2, \ldots, k_l)$ respectively, where $n_i, k_i \in \{0, 1\}$ for $i \in [1, l]$, then*

$$\binom{n}{k} = \begin{cases} 1 \mod 2 & \text{if } k \preceq n \\ 0 \mod 2 & \text{if } k \npreceq n. \end{cases}$$

Lucas' Theorem implies that a WPB function exists if and only if n is a power of 2. Hence, there are $\prod_{k=1}^{n-1} \binom{\binom{n}{k}}{\binom{n}{k}/2}$ WPB Boolean functions. Now we define the following class of balanced WAPB Boolean functions.

Definition 3.2.3. A WAPB Boolean function $f \in \mathcal{B}_n$ is called a *complementary WAPB* (shortly, *CWAPB*) if $\mathsf{w}_{k,n}(f) + \mathsf{w}_{n-k,n}(f) = \binom{n}{k}$ for $k \in [0, n]$.

Definition 3.2.4. [44] A WAPB Boolean function $f \in \mathcal{B}_n$ is called *special WAPB* (shortly, *SWAPB*) if $\mathsf{w}_{k,n}(f) = \frac{1}{2}\left[\binom{n}{k} + \delta_k^f\right]$, where

$$\delta_k^f = \begin{cases} 0 & \text{if } k \npreceq n, \\ -1 & \text{if } k \preceq n \text{ and } k < \frac{n}{2}, \\ 1 & \text{if } k \preceq n \text{ and } k > \frac{n}{2}. \end{cases}$$

Hence, a CWAPB Boolean function f is called *SWAPB* if $\mathsf{w}_{k,n}(f) = \frac{1}{2}\left[\binom{n}{k} - 1\right]$ for $0 \leq k \leq \frac{n}{2}$ and $k \preceq n$ (i.e., $\mathsf{w}_{n-k,n}(f) = \frac{1}{2}\left[\binom{n}{k} + 1\right]$ for $0 \leq k \leq \frac{n}{2}$ and $k \preceq n$). If $f \in \mathcal{B}_n$

is CWAPB, then $\delta^f_{n-k} = -\delta^f_k$ for $k \in [0, n]$. If $f \in \mathcal{B}_n$ is SWAPB, then $\delta^f_k \in \{-1, 0\}$ and $\delta^f_{n-k} = -\delta^f_k$ for $k \in [0, \lfloor \frac{n}{2} \rfloor]$. From the definition, every CWAPB Boolean function is balanced.

3.2.2 Weightwise Nonlinearity

Another key parameter for assessing the contribution of function to resistance against attacks by affine approximations is known as nonlinearity. Hence, it is important to define and construct such a Boolean function when defined over the domain restricted to the input set $\mathsf{E}_{k,n}$. Since the nonlinearity can also be related to the Walsh transform, it is necessary to define the Walsh transform over $\mathsf{E}_{k,n}$, followed by weightwise nonlinearity.

Definition 3.2.5. Let f be an n-variable Boolean function defined over $E \subseteq \mathbb{F}_2^n$, then its *restricted Walsh transform* $W_{f,E}(a)$ is defined as

$$W_{f,E}(a) = \sum_{x \in E} (-1)^{f(x)+a \cdot x}.$$

If $E = \mathsf{E}_{k,n}$ then we denote it as

$$W_{f,k}(a) = \sum_{x \in \mathsf{E}_{k,n}} (-1)^{f(x)+a \cdot x}.$$

It can be seen that the definition also holds for $E = \mathbb{F}_2^n$.

As stated by the authors in [20], Parseval's identity can also be expressed over $\mathsf{E}_{k,n}$.

$$\sum_{a \in \mathbb{F}_2^n} (W_{f,k}(a))^2 = 2^n |\mathsf{E}_{k,n}|.$$

Let $f, l_{a,b} \in \mathcal{B}_n$ where $l_{a,b}(x) = a \cdot x + b$ for $a \in \mathbb{F}_2^n$ and $b \in \mathbb{F}_2$ be an affine function. Then the distance between f and $l_{a,b}$ in the restricted domain $\mathsf{E}_{k,n}$ is

$$\begin{aligned} d_{k,n}(f(x), l_a(x)) =& \mathsf{w}_{k,n}(f(x) + l_a(x)) = \sum_{x \in \mathsf{E}_{k,n}} (f(x) + l_a(x)) \\ =& \frac{1}{2} \sum_{x \in \mathsf{E}_{k,n}} \left(1 - (-1)^{f(x)+l_a(x)}\right) = \frac{1}{2} \left[\binom{n}{k} - W_{f,k}(a) \right]. \end{aligned}$$

Similarly, the bias of $f \in \mathcal{B}_n$ over $\mathsf{E}_{k,n}$ can be defined as

$$Pr[f(x) = 1] \restriction_{\mathsf{E}_{k,n}} = \frac{\mathsf{w}_{k,n}(f)}{\binom{n}{k}}$$

Hence,

$$Pr[f(x) + l_a(x) = 1] \restriction_{\mathsf{E}_{k,n}} = \frac{1}{2}\left[1 - \frac{W_{f,k}(a)}{\binom{n}{k}}\right]$$

Definition 3.2.6. The *weightwise nonlinearity* of $f \in \mathcal{B}_n$ over $\mathsf{E}_{k,n}$, denoted as $\mathsf{NL}_k(f)$, is the Hamming distance of f to the set of all affine functions $\mathcal{A}_n$ in the restricted domain $\mathsf{E}_{k,n}$. That is, $\mathsf{NL}_k(f) = \min_{l \in \mathcal{A}_n} d_{k,n}(f, l) = \min_{l \in \mathcal{A}_n} \mathsf{w}_{k,n}(f + l)$. Hence,

$$\mathsf{NL}_k(f) = \frac{1}{2}\left[\binom{n}{k} - \max_{a \in \mathbb{F}_2^n} |W_{f,k}(a)|\right].$$

A connection between the k-weightwise nonlinearity and the Krawtchouk polynomial is presented in [66] as follows.

Proposition 3.2.2. *[66] Let $f \in \mathcal{B}_n$. Then, for $k \in [0, n]$, the weightwise nonlinearities of f are*

$$\mathsf{NL}_k(f) = \frac{1}{2}\binom{n}{k} - \frac{1}{2} \max_{\substack{a \in \mathbb{F}_2^n \\ 1 \le \mathsf{w_H}(a) \le \frac{n}{2}}} |\mathsf{K}_k(\mathsf{w_H}(a), n) - 2 \sum_{x \in \mathsf{E}_{k,n} \cap \mathsf{supp}(f)} (-1)^{a \cdot x}|$$

Any function $f \in \mathcal{B}_n$ is equal to an affine function in restricted domains $\mathsf{E}_{1,n}$ and $\mathsf{E}_{n-1,n}$ as stated in the following lemma.

Lemma 3.2.3. *For any Boolean function $f \in \mathcal{B}_n$, there always exists*

1. *a linear function $l(x) \in \mathcal{A}_n$ such that $l(x) = f(x)$ in $\mathsf{E}_{1,n}$.*

2. *an affine function $a(x) \in \mathcal{A}_n$ such that $a(x) = f(x)$ in $\mathsf{E}_{n-1,n}$.*

Proof. 1. Let $\mathsf{supp}_1(f) = \{1_{j_1}, 1_{j_2}, \ldots, 1_{j_k}\}$, where 1_{j_i} for $i \in [1, k]$ denotes the vectors with the j_i-th position in it is 1, and the other positions are 0. Considering the linear function $l(x) = x_{j_1} + x_{j_2} + \cdots + x_{j_k}$, we have $\mathsf{supp}_1(l) = \{1_{j_1}, 1_{j_2}, \ldots, 1_{j_k}\} = \mathsf{supp}_1(f)$.

2. Similarly, let $\mathsf{supp}_{n-1}(f) = \{0_{l_1}, 0_{l_2}, \ldots, 0_{l_s}\}$, where 0_{l_i} for $i \in [1, s]$ denotes the vectors with the l_i-th position in it is 0, and the other positions are 1. Considering the affine function $a(x) = (1+x_{l_1})+(1+x_{l_2})+\cdots+(1+x_{l_s}) = x_{l_1}+x_{l_2}+\cdots+x_{l_s}+s$ mod 2, we have $\mathsf{supp}_{n-1}(a) = \{0_{l_1}, 0_{l_2}, \ldots, 0_{l_s}\} = \mathsf{supp}_{n-1}(f)$.

□

Theorem 3.2.4. *For any $f \in \mathcal{B}_n$, we have*

1. $\mathsf{NL}_0(f) = \mathsf{NL}_n(f) = 0$*,*

2. $\mathsf{NL}_1(f) = \mathsf{NL}_{n-1}(f) = 0$*.*

3. *Let $f, g, c \in \mathcal{B}_n$ such that $f(x) = g(x) + c(x)$ for $x \in \mathbb{F}_2^n$, and $c(x)$ be an affine function in the domain $\mathsf{E}_{k,n}$ (although $c(x)$ is not necessarily an affine function over $\mathbb{F}_2^n$). Then $\mathsf{NL}_k(f) = \mathsf{NL}_k(g)$.*

4. *For an $f \in \mathcal{B}_n$, let $h \in \mathcal{B}_{n+1}$ defined as $h(x) = h(x', x_{n+1}) = f(x') + x_{n+1}$ for $x' \in \mathbb{F}_2^n, x_{n+1} \in \mathbb{F}_2$. Then $\mathsf{NL}_k(h) \geq \mathsf{NL}_k(f) + \mathsf{NL}_{k-1}(f)$ for $k \in [1, n]$.*

Proof. 1. From definition 3.2.6, we have

$$\mathsf{NL}_0(f) = \frac{1}{2}\left[\binom{n}{0} - \max_{a\in\mathbb{F}_2^n} |\sum_{x\in\mathsf{E}_{0,n}} (-1)^{f(x)+a\cdot x}|\right] = 0.$$

2. The proof is a consequence of Lemma 3.2.3.

3. Here, $f(x) = g(x) + c(x)$ where $g(x), c(x) \in \mathcal{B}_n$, and for $x \in \mathsf{E}_{k,n}$, $c(x) = l_{a,b}(x) = a \cdot x + b$ for some $a \in \mathbb{F}_2^n$ and $b \in \mathbb{F}_2$. Then

$$\mathsf{NL}_k(f) = \frac{1}{2}\left[\binom{n}{k} - \max_{\alpha\in\mathbb{F}_2^n} |\sum_{x\in\mathsf{E}_{k,n}} (-1)^{f(x)+\alpha\cdot x}|\right]$$

$$= \frac{1}{2}\left[\binom{n}{k} - \max_{\alpha\in\mathbb{F}_2^n} |\sum_{x\in\mathsf{E}_{k,n}} (-1)^{g(x)+c(x)+\alpha\cdot x}|\right]$$

$$= \frac{1}{2}\left[\binom{n}{k} - \max_{\alpha\in\mathbb{F}_2^n} |\sum_{x\in\mathsf{E}_{k,n}} (-1)^{g(x)+a\cdot x+b+\alpha\cdot x}|\right]$$

$$= \frac{1}{2}\left[\binom{n}{k} - \max_{\alpha'\in\mathbb{F}_2^n} |(-1)^{c}\sum_{x\in\mathsf{E}_{k,n}} (-1)^{g(x)+\alpha'\cdot x}|\right] = \mathsf{NL}_k(g).$$

4. Here, $h \in \mathcal{B}_{n+1}$ such that $h(x) = h(x', x_{n+1}) = f(x') + x_{n+1}$ for $f \in \mathcal{B}_n$ and $x' \in \mathbb{F}_2^n, x_{n+1} \in \mathbb{F}_2$. Then for $a = (a', a_{n+1}) \in \mathbb{F}_2^{n+1}$ and $k \geq 1$,

$$\begin{aligned} W_{h,k}(a) &= \sum_{x\in\mathsf{E}_{k,n+1}} (-1)^{h(x)+a\cdot x} = \sum_{x\in\mathsf{E}_{k,n+1}} (-1)^{f(x')+x_{n+1}+a'\cdot x'+a_{n+1}x_{n+1}} \\ &= \sum_{\substack{(x',0)\in\mathsf{E}_{k,n+1}\\ x'\in\mathsf{E}_{k,n}}} (-1)^{f(x')+a'\cdot x'} - \sum_{\substack{(x',1)\in\mathsf{E}_{k,n+1}\\ x'\in\mathsf{E}_{k-1,n}}} (-1)^{f(x')+a'\cdot x'+a_{n+1}} \\ &= W_{f,k}(a') - (-1)^{a_{n+1}} W_{f,k-1}(a'). \end{aligned}$$

Hence, for $k \in [1, n]$,

$$\begin{aligned} \mathsf{NL}_k(h) &= \frac{1}{2}\left[\binom{n+1}{k} - \max_{a\in\mathbb{F}_2^{n+1}} |W_{h,k}(a)|\right] \\ &= \frac{1}{2}\left[\binom{n+1}{k} - \max_{\substack{(a',a_{n+1})\in\mathbb{F}_2^{n+1}\\ a'\in\mathbb{F}_2^n}} |W_{f,k}(a') - (-1)^{a_{n+1}} W_{f,k-1}(a')|\right] \\ &= \frac{1}{2}\left[\binom{n+1}{k} - \max_{a'\in\mathbb{F}_2^n} \max\{|W_{f,k}(a') \pm W_{f,k-1}(a')|\}\right] \\ &\geq \frac{1}{2}\left[\binom{n}{k} + \binom{n}{k-1} - \max_{a'\in\mathbb{F}_2^n}\{|W_{f,k}(a')| + |W_{f,k-1}(a')|\}\right] \\ &\qquad (\because \max\{|a+b|, |a-b|\} \leq |a| + |b|) \\ &\geq \frac{1}{2}\left[\binom{n}{k} - \max_{a'\in\mathbb{F}_2^n} |W_{f,k}(a')| + \binom{n}{k-1} - \max_{a'\in\mathbb{F}_2^n} |W_{f,k-1}(a')|\right] \\ &= \mathsf{NL}_k(f) + \mathsf{NL}_{k-1}(f). \end{aligned}$$

□

An upper bound on the weightwise nonlinearities is presented by Carlet et. al. as follows.

Lemma 3.2.5 ([20], Propositions 5). *If $f \in \mathcal{B}_n$ then for $k \in [0, n]$,*

$$\mathsf{NL}_k(f) \leq \frac{1}{2}[|\mathsf{E}_{k,n}| - \sqrt{|\mathsf{E}_{k,n}|}] = \frac{1}{2}[\binom{n}{k} - \sqrt{\binom{n}{k}}].$$

From Item 1 of Theorem 3.2.4, the upper bound for $\mathsf{NL}_0(f)$ and $\mathsf{NL}_n(f)$ is tight. From Item 2 of Theorem 3.2.4, $\mathsf{NL}_1(f) = \mathsf{NL}_{n-1}(f) = 0$ for any $f \in \mathcal{B}_n$ where the upper bound is $\mathsf{NL}_1(f) = \mathsf{NL}_{n-1}(f) \leq \frac{n-\sqrt{n}}{2}$. The tightness is not achieved for $k = 1$ or $n-1$ even when n is a perfect square. The tightness of the bound of $\mathsf{NL}_k(f)$ for any $k \in [2, n-2]$ can only be checked when $\binom{n}{k}$ is a perfect square. For $k = 2$, $\binom{n}{2} = m^2$ has many positive solutions (see [3]), e.g., $\binom{2}{2} = 1$, $\binom{9}{2} = 6^2$, and $\binom{50}{2} = 35^2$. However, for all such cases of n and k, the tightness of $\mathsf{NL}_2(f)$ and $\mathsf{NL}_{n-2}(f)$ never occurs [20]. For $k = 3$, Győry [49] proved that the only case when $\binom{n}{k}$ is a perfect square is $n = 50, k = 3$ i.e., $\binom{50}{3} = 140^2$. Further, Carlet et al. mentioned in [20] (a work of Erdös' result mentioned in [3]) that $\binom{n}{k} = m^l$ has no integer solution for $l \geq 2$ and $k \in [4, n-4]$. Hence, the only case where the tightness of the upper bound can be checked is $\mathsf{NL}_3(f)$ and $\mathsf{NL}_{47}(f)$ for $f \in \mathcal{B}_{50}$. This is an open problem until now. Therefore, for $k \in [2, n-2]$, we have

$$\mathsf{NL}_k(f) < \frac{1}{2}[\binom{n}{k} - \sqrt{\binom{n}{k}}] \quad \text{(except the case } n = 50, k = 3)$$
$$\implies \mathsf{NL}_k(f) \leq \frac{1}{2}\lfloor\lfloor\binom{n}{k} - \sqrt{\binom{n}{k}}\rfloor\rfloor$$

The bound is further improved by Mesnager et al. in [79] for some cases.

3.2.3 Weightwise Algebraic Immunity

The concept of k-weightwise algebraic immunity is first introduced in [20]. The following definition of algebraic immunity of f over $\mathsf{E}_{k,n}$ plays a crucial role in understanding the security properties of cryptographic functions restricted to $\mathsf{E}_{k,n}$.

Definition 3.2.7. Let $f \in \mathcal{B}_n$ and $E \subseteq \mathbb{F}_2^n$. A function $g \in \mathcal{B}_n$ is called an annihilator of f over E if $g(x) \neq 0$ for some $x \in E$ and $f(x)g(x) = 0$ for all $x \in E$. The set of all annihilators of f over E is denoted by $Ann_E(f)$. The algebraic immunity of f over E is defined by

$$\mathsf{AI}_E(f) = \min\{\deg(g) : g \in Ann_E(f) \cup Ann_E(1+f)\}.$$

For $E = E_{n,k}$, we denote $Ann_E(f)$ and $\mathsf{AI}_E(f)$ as $Ann_k(f)$ and $\mathsf{AI}_k(f)$, respectively.

For $f \in \mathcal{B}_n$ and $E \subseteq \mathbb{F}_2^n$, if $g \in Ann_E(f)$ then there exists $x \in E$ such that $g(x) \neq 0$. This implies that an annihilator of f is not necessarily an annihilator of f on E. That is, $Ann(f) \not\subseteq Ann_E(f)$ and hence $\mathsf{AI}_E(f) \not\leq \mathsf{AI}(f)$ for $f \in \mathcal{B}_n$ and $E \subseteq \mathbb{F}_2^n$. The following example, also provided in [19], is presented in support of the claim.

Example 3.2.1. Let $0 \neq f \in \mathcal{B}_n$. Let us define a function $f' \in \mathcal{B}_{n+1}$ such that

$$f'(x, x_{n+1}) = \begin{cases} f(x), & \text{if } x_{n+1} = 0, \\ 0, & \text{if } x_{n+1} = 1 \end{cases}$$

for $x \in \mathbb{F}_2^n$. Then $g \in \mathcal{B}_{n+1}$ defined as $g(x) = x_{n+1}$ is an annihilator of f' *i.e.* $g \in \mathsf{AN}(f')$. Hence, $\mathsf{AI}(f') = 1$.

For $E = \mathbb{F}_2^n \times \{0\}$, $\mathsf{AI}_E(f') = \mathsf{AI}(f)$. If $f \in \mathcal{B}_n$ is a function whose algebraic immunity is greater than 1, then $\mathsf{AI}_E(f') > 1 = \mathsf{AI}(f')$.

3.3 Impact of Boolean Functions with restricted subset

In this section, we illustrate some examples of how the restriction to inputs of fixed Hamming weight can impact the cryptographic properties of Boolean functions. We discuss that certain Boolean functions which are known to possess optimal cryptographic properties when defined over the entire space $\mathbb{F}_2^n$, may lose these properties when their input domain is constrained to vectors of fixed Hamming weight. This shift in the function's behavior signifies the influence of the input space on the cryptographic strength of Boolean functions, particularly in the context of modern cryptographic constructions.

We begin by examining the degree of a Boolean function when restricted to the set $\mathsf{E}_{k,n}$ for $k \in [0, n]$. Let $f \in \mathcal{B}_n$ with degree d. For any $k \in [1, n-1]$, one can always find $g \in \mathcal{B}_n$ with $g(x) = f(x)$ for all $x \in \mathsf{E}_{k,n}$ such that deg$(g) \leq$ deg(f). Further, let express f as $f(x) = f_{\leq k}(x) + f_{>k}(x)$ where $f_{\leq k}(x)$ and $f_{>k}(x)$ are expressions that contain monomomials of degree less than or equal to k and greater than k, respectively. As $f_{>k}(x) = 0$ for $x \in \mathsf{E}_{k,n}$, $f(x) = f_{\leq k}(x)$ for $x \in \mathsf{E}_{k,n}$. Therefore, one can always find $g \in \mathcal{B}_n$ with $g(x) = f(x)$ for all $x \in \mathsf{E}_{k,n}$ such that deg$(g) \leq$ min$\{$deg$(f), k\}$.

A function f that is balanced over $\mathbb{F}_2^n$, is not necessarily balanced over $\mathsf{E}_{k,n}$ for all $k \in [1, n]$. In contrast, we can construct balanced Boolean functions over $\mathbb{F}_2^n$ that are (almost) balanced in all $\mathsf{E}_{k,n}$ for all $k \in [1, n]$.

Restricting the inputs with constant Hamming weight significantly deteriorates the nonlinearity of some highly nonlinear functions. A straightforward example of this case is the symmetric Boolean function. The n-variable symmetric bent function $f(x_1, x_2, \ldots, x_n) = x_1x_2 + x_1x_3 + \cdots + x_1x_n + \ldots + x_{n-1}x_n$ behaves as a constant function for all $k \in [0, n]$ as presented in the following proposition.

Proposition 3.3.1. *[20] For every even integer n, there exist n-variable bent functions f such that, for every $k = 0, 1, \ldots, n$, $\mathsf{NL}_k(f) = 0$*

Another efficient and straightforward to compute function is the Hidden Weighted Bit (HWB) function, as mentioned by D. Knuth in “The Art of Computer Programming, Volume 4”. It is defined as

$$f(x) = \begin{cases} 0, & \mathsf{w_H}(x) = 0 \\ x_{\mathsf{w_H}(x)}, & 1 \leq \mathsf{w_H}(x) \leq n. \end{cases}$$

Its cryptographic properties were studied by Wang et al. in [100]. It can be observed that the HWB function f restricted to the subset $\mathsf{E}_{k,n}$ for $1 \leq k \leq n$ is equivalent to the function $g(x) = x_k$ which is an affine function. The HWB function has nonlinearity $\mathsf{NL}(f) = 2^{n-1} - 2\binom{n-2}{\lceil \frac{n-2}{2} \rceil}$ and algebraic immunity $AI(f) \geq \lfloor \frac{n}{3} \rfloor + 1$ over $\mathbb{F}_2^n$. However, restricting the function, $\mathsf{NL}_k(f) = 0$ and $\mathsf{AI}_k(f) = 1$ for $k \in [1, n-1]$.

3.4 Conclusion

As highlighted, certain Boolean functions such as symmetric bent functions and the Hidden weighted bit (HWB) function, which demonstrates degradation despite being highly nonlinear and secure in the domain $\mathbb{F}_2^n$, become affine when restricted to $\mathsf{E}_{k,n}$ for all $k \in [0, n]$. These examples illustrate a critical challenge in cryptographic design *i.e.*functions with excellent global cryptographic properties, such as high nonlinearity or strong algebraic immunity over the space $\mathbb{F}_2^n$, may exhibit significantly weaker behavior when restricted to constant Hamming weight subsets $\mathsf{E}_{k,n}$.

Chapter 4

Binary Krawchouk Polynomial

4.1 Introduction

The Krawchouk polynomials, introduced by the Soviet Ukrainian mathematician Mykhailo Kravchuk in 1929 as part of his research in the theory of orthogonal polynomials [63]. Over the years, these polynomials have subsequently gained considerable importance in various areas of mathematics, including coding theory, information theory, cryptography, graph theory, and number theory. Several properties of the Krawchouk polynomials are well-established and have been extensively studied. These properties include their orthogonality, recurrence relations, and connections to binomial coefficients, among others.

For $m, n, q \in \mathbb{N}$, such that $m \leq n$ and $q \geq 2$, the Krawchouk polynomials $\{\mathsf{K}_m(x, n)_q\}$ are defined by

$$\sum_{m=0}^{n} \mathsf{K}_m(x, n)_q z^m = (1 - z)^x (1 + (q - 1)z)^{n-x} \tag{4.1}$$

Hence, for $x \in \{0, 1, \ldots, n\}$, the coefficient of z^m in the product of $(1 - z)^x$ and $(1 + (q - 1)z)^{n-x}$ is

$$\mathsf{K}_m(x, n)_q = \sum_{j=0}^{m} (-1)^j \binom{x}{j} \binom{n - x}{m - j} (q - 1)^{m-j}. \tag{4.2}$$

So, for $q = 2$, the binary Krawchouk polynomials $\{\mathsf{K}_m(x, n)_2\}$ or simply $\{\mathsf{K}_m(x, n)\}$ is defined as follows.

Definition 4.1.1 (Krawtchouk polynomial)**.** For a positive integer n, the Krawtchouk polynomial [68, Page 151] of degree k is given by

$$\mathsf{K}_k(x, n) = \sum_{j=0}^{k} (-1)^j \binom{x}{j} \binom{n - x}{k - j} \text{ for } k = 0, 1, \ldots n. \tag{4.3}$$

In this chapter, we present several fundamental results concerning binary Krawchouk polynomials in the case where $q = 2$. Section 4.2 is dedicated to analyzing the minimum value attained by the binary Krawchouk polynomial of a fixed degree k, as x varies. These findings offer insights into the lower bounds for both the nonlinearity and weightwise nonlinearity of the Boolean functions constructed in Chapter 7. Moreover, the properties of Krawchouk polynomials established here have significant implications from a graph-theoretic perspective. In Section 4.3, we use these results to calculate the smallest eigenvalue of the Hamming graph, highlighting the connection between the Krawchouk polynomials and the spectral properties of the Hamming graph. The following chapter is based on our result presented in [31].

4.2 Basic Properties of Binary Krawchouk Polynomial

We present some results on Krawtchouk polynomials and sequences that are useful for later results and can be of independent interest. Some nice properties of the Krawtchouk polynomials from [27, Proposition 4, Corollary 1] are presented in the following proposition.

Proposition 4.2.1. *1.* $\mathsf{K}_0(l,n) = 1, \mathsf{K}_1(l,n) = n - 2l$.

2. $\mathsf{K}_k(l,n) = (-1)^l \mathsf{K}_{n-k}(l,n)$ *(that implies,* $\mathsf{K}_{\frac{n}{2}}(l,n) = 0$ *for* n *even and* l *odd).*

3. $\mathsf{K}_k(l,n) = (-1)^k \mathsf{K}_k(n-l,n)$*, (that implies,* $\mathsf{K}_k(\frac{n}{2},n) = 0$ *for* n *even and* k *odd).*

4. For n *odd,* $|\mathsf{K}_k(1,n)| \geq |\mathsf{K}_k(l,n)|$ *where* $0 \leq k \leq n$ *and* $1 \leq l \leq n-1$*,*

5. For n *even,* $|\mathsf{K}_k(1,n)| \geq |\mathsf{K}_k(l,n)|$ *where* $0 \leq k \leq n$ *and* $1 \leq l \leq n-1$ *except* $k = \frac{n}{2}$ *or* $l = \frac{n}{2}$*.*

6. $(n-l)\mathsf{K}_k(l+1,n) = (n-2k)\mathsf{K}_k(l,n) - l\mathsf{K}_k(l-1,n)$*.*

The following relations connecting the Krawtchouk values with the slice $\mathsf{E}_{k,n}$ are derived from some results in [27, 43].

Proposition 4.2.2 (Krawtchouk polynomials relations). *For integers $n > 0$, $k \in [0, n]$ and fixed $a \in \mathbb{F}_2^n$ such that $\mathsf{w_H}(a) = \ell$, the following relations hold.*

1. $\sum\limits_{x \in \mathsf{E}_{k,n}} (-1)^{a \cdot x} = \mathsf{K}_k(\ell, n)$.
2. *If $l_{a,b}(x) = a \cdot x + b$, where $a \in \mathbb{F}_2^n, b \in \mathbb{F}_2$, is an affine Boolean function, then*

$$\mathsf{w}_{k,n}(l_{a,b}) = \frac{1}{2}(|\mathsf{E}_{k,n}| - (-1)^b \mathsf{K}_k(\ell, n)).$$

Now we present some results on the sequences from the Krawtchouk polynomial.

Lemma 4.2.3. *Let n be a positive integer and $k \in [0, n]$. Then*

1. $\mathsf{K}_k(1, n) \geq 0$ *if* $k \leq \lfloor \frac{n}{2} \rfloor$ *and* $\mathsf{K}_k(1, n) < 0$ *if* $k > \lfloor \frac{n}{2} \rfloor$.

   In particular, $\mathsf{K}_k(1, n) = 0$ *if n is even and* $k = \frac{n}{2}$.

2. $\mathsf{K}_k(2, n) \geq 0$ *if and only if* $k \leq \frac{n}{2} - \frac{\sqrt{n}}{2}$ *or,* $k \geq \frac{n}{2} + \frac{\sqrt{n}}{2}$.

   In particular, $\mathsf{K}_k(2, n) = 0$ *if n is an even square integer and* $k = \frac{n}{2} \pm \frac{\sqrt{n}}{2}$.

3. $\mathsf{K}_k(l+1, n) + \mathsf{K}_k(l, n) = 2\mathsf{K}_k(l, n-1)$ *and* $\mathsf{K}_k(l+1, n) - \mathsf{K}_k(l, n) = -2\mathsf{K}_{k-1}(l, n-1)$.

4. $\mathsf{K}_k(l, n) + \mathsf{K}_{k-1}(l, n) = \mathsf{K}_k(l, n+1)$ *and* $\mathsf{K}_k(l, n) - \mathsf{K}_{k-1}(l, n) = \mathsf{K}_k(l+1, n+1)$.

Proof. 1. The proof can be derived from the expression $\mathsf{K}_k(1, n) = \binom{n-1}{k} - \binom{n-1}{k-1}$.

2. $\mathsf{K}_k(2, n) = \binom{n-2}{k} - 2\binom{n-2}{k-1} + \binom{n-2}{k-2} = \frac{(n-2)!((2k-n)^2-n)}{k!(n-k)!}$.

   Hence, $\mathsf{K}_k(2, n) \geq 0 \iff (2k-n)^2 - n \geq 0 \iff (2k-n)^2 \geq n \iff 2k - n \geq \sqrt{n}$ or, $2k - n \leq -\sqrt{n} \iff k \geq \frac{n}{2} + \frac{\sqrt{n}}{2}$ or, $k \leq \frac{n}{2} - \frac{\sqrt{n}}{2}$. The particular case follows in a similar way.

3. We have, $\mathsf{K}_k(l+1,n)+\mathsf{K}_k(l,n)$

$$
\begin{aligned}
&= \sum_{j=0}^{k}(-1)^j\binom{l+1}{j}\binom{n-l-1}{k-j}+\sum_{j=0}^{k}(-1)^j\binom{l}{j}\binom{n-l}{k-j}\\
&= \sum_{j=0}^{k}(-1)^j\left[\binom{l+1}{j}\binom{n-l-1}{k-j}+\binom{l}{j}\binom{n-l}{k-j}\right]\\
&= \sum_{j=0}^{k}(-1)^j\left[\binom{l+1}{j}\binom{n-l-1}{k-j}+\binom{l}{j}\left(\binom{n-l-1}{k-j}+\binom{n-l-1}{k-j-1}\right)\right]\\
&= \sum_{j=0}^{k}(-1)^j\left[\left(\binom{l+1}{j}+\binom{l}{j}\right)\binom{n-l-1}{k-j}+\binom{l}{j}\binom{n-l-1}{k-j-1}\right]\\
&= \sum_{j=0}^{k}(-1)^j\left[\left(\binom{l+1}{j}+\binom{l}{j}\right)\binom{n-l-1}{k-j}\right]+\sum_{j=1}^{k+1}(-1)^{j-1}\binom{l}{j-1}\binom{n-l-1}{k-j}\\
&= \sum_{j=0}^{k}(-1)^j\left[\left(\binom{l+1}{j}+\binom{l}{j}\right)\binom{n-l-1}{k-j}\right]+\sum_{j=0}^{k}(-1)^{j-1}\binom{l}{j-1}\binom{n-l-1}{k-j}\\
&= \sum_{j=0}^{k}(-1)^j\left(\binom{l+1}{j}+\binom{l}{j}-\binom{l}{j-1}\right)\binom{n-l-1}{k-j}\\
&= 2\sum_{j=0}^{k}(-1)^j\binom{l}{j}\binom{n-l-1}{k-j}\\
&= 2\mathsf{K}_k(l,n-1).
\end{aligned}
$$

The second part of this item can be proven using a technique similar to that used that used above.

4. These equalities can be proved by adding and subtracting the equalities in Item 3 respectively.

□

We present the minimum of the sequence $\mathsf{K}_k(l,n), 0\le l\le n$ for a fixed k and n in the following theorem.

Theorem 4.2.4. *1. Let $n=2m+1$ be an odd integer for some $m\in\mathbb{Z}^+$. Then for*

$k \in [0, n]$

$$\min_{0 \le l \le n-1} \mathsf{K}_k(l, n) = \begin{cases} \mathsf{K}_k(n-1, n) = -\mathsf{K}_k(1, n) & \text{for } k \in [0, \frac{n-1}{2}] \text{ and odd,} \\ \mathsf{K}_k(1, n) & \text{for } k \in [\frac{n+1}{2}, n]. \end{cases}$$

In particular (when $k = m$), $\min_{0 \le l \le n-1} \mathsf{K}_m(l, n) = \mathsf{K}_m(2, n) = -\mathsf{K}_m(1, n)$.

2. *Let $n = 2m$ be an even integer for some $m \in \mathbb{Z}^+$.*

   Then for $k \in [m+1, n]$ and even, $\min_{0 \le l \le n} \mathsf{K}_k(l, n) = \mathsf{K}_k(1, n)$.

   Moreover,

   (a) $\min_{0 \le l \le n} \mathsf{K}_m(l, n) = \mathsf{K}_m(2, n)$ if m is even,

   (b) for $n \ge 10$, $\min_{0 \le l \le n} \mathsf{K}_{m-1}(l, n) = \mathsf{K}_{m-1}(2, n)$ if m is odd.

3. $\min_{0 \le l \le n} \mathsf{K}_2(l, n) = -\lfloor \frac{n}{2} \rfloor$.

4. $\max_{0 \le l \le n} \mathsf{K}_k(l, n) = \mathsf{K}_k(0, n) = \binom{n}{k}$.

5. $$\mathsf{K}_k(n, n) = \begin{cases} \max_{0 \le l \le n} \mathsf{K}_k(l, n) & \text{if } k \text{ is even,} \\ \min_{0 \le l \le n} \mathsf{K}_k(l, n) & \text{if } k \text{ is odd.} \end{cases}$$

Proof. 1. From Proposition 4.2.1[Item 4], for $k \in [0, n]$, we have $|\mathsf{K}_k(1, n)| \ge |\mathsf{K}_k(l, n)|$ for all $l \in [1, n-1]$.

For $k \in [\frac{n+1}{2}, n]$, $\mathsf{K}_k(1, n) < 0$ (from Lemma 4.2.3[Item 1]), we have $\min_{1 \le l \le n-1} \mathsf{K}_k(l, n)$ $= \mathsf{K}_k(1, n)$.

Furthermore, k being an odd integer in $[0, \frac{n-1}{2}]$, using Proposition 4.2.1[Item 3], we have $\mathsf{K}_k(1, n) = -\mathsf{K}_k(n-1, n)$. Hence, for $k \in [0, \frac{n-1}{2}]$, since $\mathsf{K}_k(1, n) \ge 0$ (from Lemma 4.2.3[Item 1]), we get $\max_{1 \le l \le n-1} \mathsf{K}_k(l, n) = \mathsf{K}_k(1, n)$ i.e., $\min_{1 \le l \le n-1} \mathsf{K}_k(l, n) =$ $\mathsf{K}_k(n-1, n)$.

Since $\mathsf{K}_k(0, n) = \binom{n}{k} > 0$, we have $\min_{0 \le l \le n-1} \mathsf{K}_k(l, n) = \min_{1 \le l \le n-1} \mathsf{K}_k(l, n)$ and it proves the result of the first part of the item.

Since $n = 2m+1$, we have from Lemma 4.2.3[Item 3 and Item 1] that $\mathsf{K}_m(2,n) + \mathsf{K}_m(1,n) = 2\mathsf{K}_m(1,n-1) = 0$. It implies $\mathsf{K}_m(2,n) = -\mathsf{K}_m(1,n) \leq 0$.

Since $\mathsf{K}_m(0,n) = \binom{n}{m} > 0$, we obtain $\min\limits_{0 \leq l \leq n} \mathsf{K}_m(l,n) = \min\limits_{1 \leq l \leq n-1} \mathsf{K}_m(l,n) = \mathsf{K}_m(2,n)$ (from Proposition 4.2.1[Item 4]). Accordingly, the second part of this item is proven.

2. Since $k \geq m+1$, from Proposition 4.2.1[Item 5], we have $|\mathsf{K}_k(1,n)| \geq |\mathsf{K}_k(l,n)|$ for $l \in [1, n-1]$. As $\mathsf{K}_k(1,n) \leq 0$ (from Lemma 4.2.3[Item 1]) and $\mathsf{K}_k(0,n) = \mathsf{K}_k(n,n) = \binom{n}{k} > 0$, $\min\limits_{0 \leq l \leq n} \mathsf{K}_k(l,n) = \min\limits_{1 \leq l \leq n-1} \mathsf{K}_k(l,n) = \mathsf{K}_k(1,n)$. Hence, the first part is proven.

   Additionally, from Lemma 4.2.3[Item 2]), we have $\mathsf{K}_m(2,n) \leq 0$ since $m = \frac{n}{2}$.

   (a) Here we consider the case $m = \frac{n}{2}$ even. We show $|\mathsf{K}_m(2,n)| \geq |\mathsf{K}_m(l,n)|$ for $1 \leq l \leq n-1$. At first, we use induction on l to show it for l even and $2 \leq l \leq \frac{n}{2}$.

   For $l = 2$, it is direct since $|\mathsf{K}_m(2,n)| = |\mathsf{K}_m(l,n)|$.

   Then, assume that $|\mathsf{K}_m(2,n)| \geq |\mathsf{K}_m(l,n)|$ for some even l and $2 \leq l \leq \frac{n}{2} - 2$. Then, we need to prove that $|\mathsf{K}_m(2,n)| \geq |\mathsf{K}_m(l+2,n)|$.

   From Proposition 4.2.1[Item 6], for $2 \leq l \leq \frac{n-4}{2} = \frac{n}{2} - 2$, we have:

$$\begin{aligned} n-(l+1))\mathsf{K}_m(l+2,n) &= (n-2m)\mathsf{K}_m(l+1,n) - (l+1)\mathsf{K}_m(l,n) \\ &= -(l+1)\mathsf{K}_m(l,n), \text{ since } n = 2m \\ \Longrightarrow \qquad |(n-l-1)\mathsf{K}_m(l+2,n)| &= |-(l+1)\mathsf{K}_m(l,n)| \\ \Longrightarrow \qquad |\mathsf{K}_m(l+2,n)| &= \tfrac{l+1}{n-l-1}|\mathsf{K}_m(l,n)| \leq |\mathsf{K}_m(l,n)| \leq |\mathsf{K}_m(2,n)|, \\ &\text{since } \tfrac{l+1}{n-l-1} \leq 1 \qquad . \end{aligned}$$

   Therefore, $|\mathsf{K}_m(2,n)| \geq |\mathsf{K}_m(l,n)|$ for all $2 \leq l \leq \frac{n}{2}$ and even. From Proposition 4.2.1[Item 2], $\mathsf{K}_m(l,n) = 0$ for l odd. Hence, $|\mathsf{K}_m(2,n)| \geq |\mathsf{K}_m(l,n)|$ for all $1 \leq l \leq \frac{n}{2}$. Further, from Proposition 4.2.1[Item 3], $\mathsf{K}_m(l,n) = \mathsf{K}_m(n-l,n)$ as m even. Hence, $|\mathsf{K}_m(2,n)| \geq |\mathsf{K}_m(l,n)|$ for all $1 \leq l \leq n-1$. Therefore, as $\mathsf{K}_m(2,n) < 0$ and $\mathsf{K}_m(0,n) = \mathsf{K}_m(n,n) = \binom{n}{m} > 0$, $\min\limits_{0 \leq l \leq n} \mathsf{K}_m(l,n) = \min\limits_{1 \leq l \leq n-1} \mathsf{K}_m(l,n) = \mathsf{K}_m(2,n)$ for m even.

(b) Now we focus on the case $m = \frac{n}{2}$ odd. At first, we show that $|\mathsf{K}_{m-1}(2,n)| \geq |\mathsf{K}_{m-1}(l,n)|$ for $2 \leq l \leq \frac{n}{2}$ using induction on l.

For $l = 2$, we have $|\mathsf{K}_{m-1}(2,n)| = |\mathsf{K}_{m-1}(l,n)|$.

Then, we prove the property for $l = 3$ i.e. $|\mathsf{K}_{m-1}(2,n)| \geq |\mathsf{K}_{m-1}(3,n)|$. This is needed since (as we will see later) the induction has a recursion with depth two. It can be checked from Lemma 4.2.3[Item 2] that $\mathsf{K}_{m-1}(2,n) \leq 0$ for $n \geq 4$. To prove $|\mathsf{K}_{m-1}(2,n)| \geq |\mathsf{K}_{m-1}(3,n)|$, we need to show that $\mathsf{K}_{m-1}(2,n) \leq \mathsf{K}_{m-1}(3,n) \leq -\mathsf{K}_{m-1}(2,n)$ i.e., $\mathsf{K}_{m-1}(3,n) - \mathsf{K}_{m-1}(2,n) \geq 0$ and $\mathsf{K}_{m-1}(3,n) + \mathsf{K}_{m-1}(2,n) \leq 0$.

From Lemma 4.2.3[Item 3 and Item 1], we have $\mathsf{K}_{m-1}(3,n) + \mathsf{K}_{m-1}(2,n) = 2\mathsf{K}_{m-1}(2,n-1) = 2\left(2\mathsf{K}_{m-1}(1,n-2) - \mathsf{K}_{m-1}(1,n-1)\right) = -2\mathsf{K}_{m-1}(1,n-1) \leq 0$.

Similarly, $\mathsf{K}_{m-1}(3,n) - \mathsf{K}_{m-1}(2,n) = -2\mathsf{K}_{m-2}(2,n-1) \geq 0$ if $m-2 \geq \frac{n-1-\sqrt{n-1}}{2}$ *i.e.*, if $n \geq 10$ (Lemma 4.2.3[Item 2]). Hence, $|\mathsf{K}_{m-1}(2,n)| \geq |\mathsf{K}_{m-1}(3,n)|$ if $n \geq 10$.

Assume that $|\mathsf{K}_{m-1}(2,n)| \geq |\mathsf{K}_{m-1}(l-1,n)|$ and $|\mathsf{K}_{m-1}(2,n)| \geq |\mathsf{K}_{m-1}(l,n)|$ for some $3 \leq l \leq \frac{n}{2}-1$. Then, we need to prove that $|\mathsf{K}_{m-1}(2,n)| \geq |\mathsf{K}_{m-1}(l+1,n)|$.

Using Proposition 4.2.1[Item 6], we have

$$\begin{aligned}
(n-l)\mathsf{K}_{m-1}(l+1,n) &= (n-2(m-1))\mathsf{K}_{m-1}(l,n) - l\mathsf{K}_{m-1}(l-1,n)\\
&= 2\mathsf{K}_{m-1}(l,n) - l\mathsf{K}_{m-1}(l-1,n)\\
\implies (n-l)|\mathsf{K}_{m-1}(l+1,n)| &\le 2|\mathsf{K}_{m-1}(l,n)| + l|\mathsf{K}_{m-1}(l-1,n)|\\
&\le (2+l)|\mathsf{K}_{m-1}(2,n)|\\
\implies |\mathsf{K}_{m-1}(l+1,n)| &\le \frac{l+2}{n-l}|\mathsf{K}_{m-1}(2,n)|\\
\implies |\mathsf{K}_{m-1}(l+1,n)| &\le |\mathsf{K}_{m-1}(2,n)| \qquad \text{as } \frac{l+2}{n-l} \le 1 \text{ for } 2 \le l \le \frac{n}{2}-1.
\end{aligned}$$

Hence, $|\mathsf{K}_{m-1}(2,n)| \ge |\mathsf{K}_{m-1}(l,n)|$ for $2 \le l \le \frac{n}{2}$.

Then, $|\mathsf{K}_{m+1}(2,n)| \ge |\mathsf{K}_{m+1}(l,n)|$ for $2 \le l \le \frac{n}{2}$ since, $|\mathsf{K}_{m+1}(l,n)| = |\mathsf{K}_{m-1}(l,n)|$ (from Proposition 4.2.1[Item 2]).

Now, we show that $\min_{0\le l\le n} \mathsf{K}_{m+1}(l,n) = \mathsf{K}_{m+1}(1,n)$. From Lemma 4.2.3[Item 3 and Item 1], we have $\mathsf{K}_{m+1}(2,n) - \mathsf{K}_{m+1}(1,n) = -2\mathsf{K}_m(1,n-1) > 0$. Similarly, $\mathsf{K}_{m+1}(2,n) + \mathsf{K}_{m+1}(1,n) = 2\mathsf{K}_{m+1}(1,n-1) \le 0$. That implies $-\mathsf{K}_{m+1}(1,n) \le \mathsf{K}_{m+1}(2,n) < \mathsf{K}_{m+1}(1,n)$ *i.e.*, $|\mathsf{K}_{m+1}(2,n)| \le |\mathsf{K}_{m+1}(1,n)|$.

From Lemma 4.2.3[Item 1], we have $\mathsf{K}_{m+1}(1,n) \le 0$.

Accordingly, $\min_{0\le l\le n} \mathsf{K}_{m+1}(l,n) = \mathsf{K}_{m+1}(1,n)$.

3. We have:

$$\begin{aligned}
\mathsf{K}_k(l,n) &= \sum_{j=0}^{k}(-1)^j\binom{l}{j}\binom{n-l}{k-j}\\
&= \sum_{j=0}^{k}\binom{l}{j}\binom{n-l}{k-j} - 2\sum_{\substack{j=0\\ j\text{ odd}}}^{k}\binom{l}{j}\binom{n-l}{k-j}\\
&= \binom{n}{k} - 2\sum_{\substack{j=0\\ j\text{ odd}}}^{k}\binom{l}{j}\binom{n-l}{k-j}. \qquad (4.4)
\end{aligned}$$

Hence, $\mathsf{K}_2(l,n) = \binom{n}{2} - 2\binom{l}{1}\binom{n-l}{1} = \binom{n}{2} - 2l(n-l)$. For real value of l, the function $\mathsf{K}_2(l,n)$ has minima at $l = \frac{n}{2}$ as $\frac{d(\mathsf{K}_2(l,n))}{dl} = 4l - 2n = 0$ at $l = \frac{n}{2}$ and $\frac{d^2(\mathsf{K}_2(l,n))}{dl^2} = 4 > 0$. Since ℓ in a positive integer, we have $\min\limits_{0 \leq l \leq n} \mathsf{K}_2(l,n)$ is $\binom{n}{2} - 2(\frac{n}{2})^2 = -\frac{n}{2}$ when n is even. For n is odd, it can be checked that $\mathsf{K}_2(l,n)$ has minimum at $l = \frac{n-1}{2}$ and $l = \frac{n+1}{2}$ with value $\min\limits_{0 \leq l \leq n} \mathsf{K}_2(l,n) = \binom{n}{2} - 2\frac{n-1}{2}\frac{n+1}{2} = -\frac{n-1}{2}$. Hence, combining both the cases, we have $\min\limits_{0 \leq l \leq n} \mathsf{K}_2(l,n) = -\lfloor \frac{n}{2} \rfloor$.

4. From Equation 4.4, we have $\mathsf{K}_k(l,n) = \binom{n}{k} - 2\sum\limits_{\substack{j=0 \\ j:\text{odd}}}^{k} \binom{l}{j}\binom{n-l}{k-j} \leq \binom{n}{k} = \mathsf{K}_k(0,n)$. It finishes the demonstration of the item.

5. From Proposition 4.2.1[Item 3], we have $\mathsf{K}_k(n,n) = (-1)^k \mathsf{K}_k(0,n)$. Hence, for k is even, $\mathsf{K}_k(n,n) = \mathsf{K}_k(0,n)$ is maximum in $\mathsf{K}_k(l,n), l \in [0,n]$. Like Equation 4.4, we have $\mathsf{K}_k(l,n) = 2\sum\limits_{\substack{j=0 \\ j:\text{even}}}^{k} \binom{l}{j}\binom{n-l}{k-j} - \binom{n}{k} \geq -\binom{n}{k} = -\mathsf{K}_k(0,n)$. Hence, for odd k, we have $\mathsf{K}_k(n,n) = -\mathsf{K}_k(0,n) \leq \mathsf{K}_k(l,n)$ for $l \in [0,n]$.

□

4.3 Krawchouk Polynomials and Hamming Graph

This section highlights several key findings from our investigation of Krawtchouk polynomials, particularly on the spectral analysis of Hamming graphs.

Let $q \geq 2$, $n \geq 1$ be integers and let $\sum$ be a set of alphabets with cardinality q. The Hamming graphs $H(n,q,k)$, where $0 \leq k \leq n$, have vertex set $V = \sum^n$, the set of all strings of length n over the alphabet $\sum$. Two vertices $v_i, v_j \in \sum^n$ for $i \neq j$ are adjacent if and only if $\mathsf{d_H}(v_i, v_j) = k$. Then $|E| = \frac{1}{2}\binom{n}{k}q^n(q-1)^k$ where E is the set of edges in $H(n,q,k)$.

Hence from the work in [14], the eigenvalues of the graph $H(n,q,k)$ are $\mathsf{K}_k(i,n)_q$ for a fix k and $0 \leq i \leq n$. The graph $H(n,q,k)$ is regular of degree $\mathsf{K}_k(0,n) = (q-1)^k\binom{n}{k}$.

For $q = 2$ and $k = 1$, $H(n, 2, 1)$ is the hypercube of dimension n and is denoted as Q_n. In 2016, Van Dam and Sotirov in [99] conjectured the following.

Conjecture 4.3.1. Let $q \geq 2$, and $k \geq n - \frac{n-1}{q}$, where k is taken even when $q = 2$. Then the smallest eigenvalue of $H(n, q, k)$ is $\mathsf{K}_k(1, n)$.

Alon and Sudakov [4], proved this for $q = 2$, and n large and $\frac{k}{n}$ fixed. The conjecture for $q = 2$ and for all n was fully resolved by Dummer and Kapralova in [34], which is presented as follows.

Proposition 4.3.2. *Let* $q = 2$*.*

i. *If* $k \neq \frac{n}{2}$*, then* $|\mathsf{K}_k(i, n)| \leq |\mathsf{K}_k(1, n)|$ *for all* $i, 1 \leq i \leq n - 1$*.*

ii. *If* $k = \frac{n}{2}$*, then* $\mathsf{K}_k(1, n) = 0$ *and* $|\mathsf{K}_k(i, n)| \leq |\mathsf{K}_k(2, n)|$ *for all* i*,* $1 \leq i \leq n - 1$*.*

Corollary 4.3.3. *Let* $q = 2$*, and* $k \geq \frac{(n+1)}{2}$*.*

i. $\mathsf{K}_k(1, n) \leq \mathsf{K}_k(i, n)$ *for all* i*,* $0 \leq i \leq n - 1$*.*

ii. $\mathsf{K}_k(1, n) \leq \mathsf{K}_k(n, n)$ *if and only if* k *is even or* $k = n$*.*

Example 4.3.1. Let $n = 8$. We represent the Krawchouk matrix $K = [a_{ki}]$, where $a_{ki} = \mathsf{K}_k(i, n)$, for $0 \leq k \leq n$ and $0 \leq i \leq n$. Hence, the row represents the eigenvalues of $H(n, 2, k)$.

$$K = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 8 & 6 & 4 & 2 & 0 & -2 & -4 & -6 & -8 \\ 28 & 14 & 4 & -2 & -4 & -2 & 4 & 14 & 28 \\ 56 & 14 & -4 & -6 & 0 & 6 & 4 & -14 & -56 \\ 70 & 0 & -10 & 0 & 6 & 0 & -10 & 0 & 70 \\ 56 & -14 & -4 & 6 & 0 & -6 & 4 & 14 & -56 \\ 28 & -14 & 4 & 2 & -4 & 2 & 4 & -14 & 28 \\ 8 & -6 & 4 & -2 & 0 & 2 & -4 & 6 & -8 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \end{pmatrix}$$

Therefore, some of our results provide additional insight into the smallest eigenvalue of the Hamming graph for $q = 2$ with $0 \leq k \leq n$. These findings deepen our understanding of the spectral properties of the graph.

Theorem 4.3.4. *Let $q = 2$.*

i. *The smallest eigenvalue of $H(n, 2, m)$, for $n = 2m + 1$ when m even is $\mathsf{K}_m(2, n) = -\mathsf{K}_m(1, n)$.*

ii. *The smallest eigenvalue of $H(n, 2, m)$, for $n = 2m$ when m even is $\mathsf{K}_m(2, n)$.*

iii. *The smallest eigenvalue of $H(n, 2, m - 1)$, for $n \geq 10$ when m odd is $\mathsf{K}_{m-1}(2, n)$.*

iv. *The smallest eigenvalue of $H(n, 2, 2)$ is $-\lfloor \frac{n}{2} \rfloor$.*

v. *The second largest eigenvalue of $H(n, 2, k)$ for $k \neq \frac{n}{2}$ odd, is $\mathsf{K}_k(1, n)$ for $k \leq \lfloor \frac{n}{2} \rfloor$ and $\mathsf{K}_k(n - 1, n)$ for $k > \lfloor \frac{n}{2} \rfloor$.*

Proof. i. The proof of the theorem is concluded by the Theorem 4.2.4[Item- 1].

ii. The proof of the theorem is concluded using the Theorem 4.2.4[Item- 2] and Proposition 4.2.1[Item- 3].

iii. The proof of the theorem is concluded by the Theorem 4.2.4[Item- 2].

iv. It is straightforward from Theorem 4.2.4[Item- 3].

v. Since k is odd, it follows from the Proposition 4.2.1[Item- 3] that $\mathsf{K}_k(0, n) = -\mathsf{K}_k(n, n)$. Therefore, the largest eigenvalue is $\mathsf{K}_k(0, n)$ and the smallest is $\mathsf{K}_k(n, n)$. Furthermore, applying Proposition 4.2.1[Item- 3], we have $\mathsf{K}_k(1, n) = -\mathsf{K}_k(n - 1, n)$. Now, using Proposition 4.3.2, Lemma 4.2.3[Item- 1] and Corollary 4.3.3, we conclude that if $k > \lfloor \frac{n}{2} \rfloor$, then $\mathsf{K}_k(1, n) < 0$, which implies $\mathsf{K}_k(n-1, n)$ is the second largest eigenvalue.

On the otherhand, when $k \leq \lfloor \frac{n}{2} \rfloor$, Lemma 4.2.3[Item- 1] ensures that $\mathsf{K}_k(1, n) \geq 0$, and Proposition 4.3.2 implies that $\mathsf{K}_k(1, n)$ is the second largest eigenvalue.

□

4.4 Conclusion

In this chapter, we have discussed the binary Krawchouk polynnomials where we have particularly emphasized on their minimum values for fixed degrees. These minimum values are proved to play significant role in deriving the lower bound for nonlinearity and weightwise nonlinearity of the Boolean function defined using permutation group in Chapter 7. Furthermore, we have highlighted the relevance of these results in the spectral analysis of Hamming graphs, where the minimum value of the Krawchouk polynomial directly determines the smallest eigenvalue of the graph.

Chapter 5

On the Secondary Constructions of WAPB Functions

5.1 Introduction

In the context of Boolean function construction, two main approaches are commonly used: primary constructions, which involve designing functions without relying on existing ones, and secondary constructions, which generate new functions by combining or modifying the known ones. Several well-behaved cryptographic Boolean functions in higher variables can be constructed following secondary construction by combining functions in a smaller number of variables from the same class. For example, a bent function with $n_1 + n_2$ variables can be constructed by combining bent functions with n_1 and n_2 variables, respectively. Secondary constructions include the direct sum of functions [95], Siegenthaler's construction [95], the indirect sum of functions [17], and constructions without extension of the number of variables [18], etc. However, identifying suitable combinations for constructing new WAPB/WPB functions that balance cryptographic properties remains relatively unexplored, although most of the existing constructions are based on modification of the support of known Boolean functions. The first construction of Weightwise Perfectly Balanced (WPB) Boolean functions, presented in [20] is based on the indirect sum of four Boolean functions. Subsequently, Zhu and Su in [106] proposed an elegant construction of WAPB functions by the direct sum of several known WPB Boolean functions. Guo and Su [48] introduced another class of WAPB Boolean functions on n-variables by modifying the support of quadratic Boolean functions. Several constructions of WPB and WAPB

Boolean functions are presented in [77] by modifying the support of linear and quadratic functions. In [44], a secondary construction of WAPB Boolean functions was introduced using Siegenthaler's construction.

In this chapter, we present several secondary constructions of Weightwise Almost Perfectly Balanced (WAPB) and Weightwise Perfectly Balanced (WPB) Boolean functions. Section 5.2 focuses on the construction of WAPB functions using Siegenthaler's classical secondary construction. In Section 5.3, we propose a recursive method to generate a class of WAPB Boolean functions from a given WAPB function. The construction integrates ideas from Siegenthaler's method and the WPB function construction proposed by Mesnager and Su [77]. The cryptographic properties of the resulting functions including their algebraic normal forms, nonlinearity and weightwise nonlinearities, as well as algebraic immunity and weightwise algebraic immunity, are analyzed in Subsection 5.3.1, Subsection 5.3.2, and Subsection 5.3.3, respectively. Additionally, Section 5.4 introduces a modified class of WAPB Boolean functions derived from the generalized construction. These modified functions demonstrate notable improvements in both nonlinearity and weightwise nonlinearity. Furthermore, we present a construction in Section 5.5 that perturbs the support vectors of a highly nonlinear Boolean function, yielding WAPB functions with substantially enhanced cryptographic parameters. We present experimental results and comparative tables for various values of n at the end of the chapter. The content of this chapter is based on our works in [28, 29].

The following combinatorial identities are useful for our study and construction. The book [47] contains a list of useful combinatorial identities and inequalities.

Lemma 5.1.1. *[47, Item3.10] For the positive integers m, k, we have*

$$\sum_{i=0}^{\lfloor \frac{k-1}{2} \rfloor} \binom{m}{2i+1}\binom{m}{k-(2i+1)} = \frac{1}{2}\binom{2m}{k} + \frac{(-1)^{\frac{k}{2}}}{2}\binom{m}{\frac{k}{2}}\frac{1+(-1)^k}{2}.$$

1. *If k is odd, that implies* $\displaystyle\sum_{j=1,\ j \text{ is odd}}^{k} \binom{m}{j}\binom{m}{k-j} = \frac{1}{2}\binom{2m}{k}$.
2. *If k is even, that implies* $\displaystyle\sum_{j=1,\ j \text{ is odd}}^{k} \binom{m}{j}\binom{m}{k-j} = \frac{1}{2}\binom{2m}{k} - (-1)^{\frac{k}{2}}\binom{m}{\frac{k}{2}}$.
   (a) *If $\frac{k}{2}$ is odd, that implies* $\displaystyle\sum_{j=1,\ j \text{ is odd}}^{k} \binom{m}{j}\binom{m}{k-j} = \frac{1}{2}\binom{2m}{k} + \binom{m}{\frac{k}{2}}$.
   (b) *If $\frac{k}{2}$ is even, that implies* $\displaystyle\sum_{j=1,\ j \text{ is odd}}^{k} \binom{m}{j}\binom{m}{k-j} = \frac{1}{2}\binom{2m}{k} - \binom{m}{\frac{k}{2}}$.

5.2 Siegenthaler's Construction

Let $g, h \in \mathcal{B}_n$ be two Boolean functions. Consider the function $f \in \mathcal{B}_{n+1}$ defined as

$$f(x_1, x_2, \cdots, x_n, x_{n+1}) = (x_{n+1} + 1)g(x_1, x_2, \cdots, x_n) + x_{n+1}h(x_1, x_2, \cdots, x_n). \quad (5.1)$$

Here, the truth table of f is the concatenation of the truth tables of g and h. The following are some results based on the construction of the WAPB Boolean function based on Siegenthaler's construction (as in Equation 5.1).

Lemma 5.2.1. *Let n be an odd integer. For $k \in [0, n]$,*

1. *if $k \not\preceq n$, then $k \not\preceq n-1$ and $k-1 \not\preceq n-1$.*
2. *if $k \preceq n$ and*
   (a) *k is odd, then $k \not\preceq n-1$ and $k-1 \preceq n-1$;*
   (b) *k is even, then $k \preceq n-1$ and $k-1 \not\preceq n-1$.*

Proof. Let the binary expansion of n and k be $n = \sum_{i=0}^{l} n_i 2^i$ and $k = \sum_{i=0}^{l} k_i 2^i$ where $n_i, k_i \in \{0, 1\}$ for $i \in [0, l]$. Further consider the binary expansion of $n-1$ and $k-1$ be $n-1 = \sum_{i=0}^{l} n_i' 2^i$ and $k = \sum_{i=0}^{l} k_i' 2^i$ where $n_i', k_i' \in \{0, 1\}$ for $i \in [0, l]$. As n is odd, $n_0 = 1$, $n_0' = 0$ and $n_i = n_i'$ for $i \in [1, l]$.

1. As $k \npreceq n$, $k_s > n_s$ for some $s \in [1, l]$. Here $s \neq 0$ as $n_0 = 1$. As $n'_s = n_s$, $k_s > n'_s$. So, $k \npreceq n - 1$.

   If k is odd, then in the binary expansion of $k - 1$, $k'_0 = 0$, $k_0 = 1$, and $k'_i = k_i$ for $i \in [1, l]$. Here, we have $k'_s > n'_s$ and that implies $k - 1 \npreceq n - 1$. If k is even, then $k - 1$ is odd. In this case, $k'_0 = 1$ and $n'_0 = 0$. So, $k - 1 \npreceq n - 1$.

2. As $k \preceq n$, $k_i \leq n_i, \forall i \in [0, l]$. If k is odd, $k - 1$ is even. Then $k'_0 = 0$ and $n'_0 = 0$ and other bits satisfy $k'_i \leq n'_i, \forall i \in [1, l]$. Hence, $k - 1 \preceq n - 1$. As k is odd, $n'_0 < k_0$. So, $k \npreceq n - 1$.

   Similarly, if k is even, $k_0 = n'_0 = 0$ and $k_i \leq n'_i = n_i, \forall i \in [1, l]$. That implies $k \preceq n - 1$. Furthermore, since $n'_0 = 0$ and $k'_0 = 1$, $k - 1 \npreceq n - 1$.

□

Now we propose to construct WAPB Boolean functions using Siegenthaler's secondary construction when n is odd. Gini and Méaux [44] have also used Siegenthaler's secondary construction to obtain n-variable SWAPB Boolean functions (definition 3.2.3) for any positive integer n.

Lemma 5.2.2. *Let $n > 1$ be an odd integer and $g, h \in \mathcal{B}_{n-1}$ be two WAPB Boolean functions. Then $f \in \mathcal{B}_n$ defined as*

$$f(x_1, x_2, \ldots, x_n) = (1 + x_n)g(x_1, x_2, \ldots, x_{n-1}) + x_n h(x_1, x_2, \ldots, x_{n-1}),$$

$$\textit{i.e.}, \mathsf{supp}(f) = \{(x, 0) \in \mathbb{F}_2^n : x \in \mathsf{supp}(g)\} \cup \{(y, 1) \in \mathbb{F}_2^n : y \in \mathsf{supp}(h)\}$$

is a WAPB Boolean function.

If $\mathsf{w}_{k,n-1}(g) = \frac{1}{2}\left[\binom{n-1}{k} + b_k^{n-1}\right]$, $\mathsf{w}_{k,n-1}(h) = \frac{1}{2}\left[\binom{n-1}{k} + c_k^{n-1}\right]$ *and* $\mathsf{w}_{k,n}(f)$

$= \frac{1}{2}\left[\binom{n}{k} + a_k^n\right]$ *then*

$$a_k^n = \begin{cases} 0 & \textit{if } k \npreceq n, \\ b_k^{n-1} & \textit{if } k \preceq n, k < n \textit{ and } k \textit{ is even}, \\ c_{k-1}^{n-1} & \textit{if } k \preceq n, k < n \textit{ and } k \textit{ is odd}. \end{cases}$$

Proof. Here $g, h \in \mathcal{B}_{n-1}$ are WAPB Boolean functions with $\mathsf{w}_{k,n}(g) = \frac{1}{2}\left[\binom{n-1}{k} + b_k^{n-1}\right]$ and $\mathsf{w}_{k,n}(h) = \frac{1}{2}\left[\binom{n-1}{k} + c_k^{n-1}\right]$ for $k \in [0, n-1]$ where

$$b_k^{n-1} = \begin{cases} 0 & if\ k \npreceq n-1, \\ \pm 1 & if\ k \preceq n-1 \end{cases} \quad \textit{and} \quad c_k^{n-1} = \begin{cases} 0, & if\ k \npreceq n-1, \\ \pm 1, & if\ k \preceq n-1. \end{cases}$$

As f is defined, we have $\mathsf{w}_{0,n}(f) = \mathsf{w}_{0,n-1}(g) = \frac{\binom{n-1}{0}+b_0^{n-1}}{2} = \frac{\binom{n}{0}+b_0^{n-1}}{2}$ and $\mathsf{w}_{n,n}(f) = \mathsf{w}_{n-1,n-1}(h) = \frac{\binom{n-1}{n-1}+c_{n-1}^{n-1}}{2} = \frac{\binom{n}{n}+c_{n-1}^{n-1}}{2}$.

Further, for $k \in [1, n-1]$, we have

$\mathsf{supp}_k(f) = \{(x, 0) : x \in \mathsf{supp}_k(g)\} \cup \{(y, 1) : y \in \mathsf{supp}_{k-1}(h)\}$. That implies,

$$\begin{aligned} \mathsf{w}_{k,n}(f) &= \mathsf{w}_{k,n-1}(g) + \mathsf{w}_{k-1,n-1}(h) \\ &= \frac{1}{2}\left[\binom{n-1}{k} + b_k^{n-1}\right] + \frac{1}{2}\left[\binom{n-1}{k-1} + c_{k-1}^{n-1}\right] \\ &= \frac{1}{2}\left[\binom{n}{k} + a_k^n\right]; \qquad \textit{where } a_k^n = b_k^{n-1} + c_{k-1}^{n-1}. \end{aligned} \tag{5.2}$$

- If $k \npreceq n$, then using Lemma 5.2.1, $k \npreceq n-1$ and $k-1 \npreceq n-1$. Then $b_k^{n-1} = c_{k-1}^{n-1} = 0$. Hence, $a_k^n = 0$, *i.e.*, $\mathsf{w}_{k,n}(f) = \frac{1}{2}\binom{n}{k}$.
- If $k \preceq n$, then from Lemma 5.2.1,
  - if k is odd, then $k \npreceq n-1$ and $k-1 \preceq n-1$. Then $b_k^{n-1} = 0$ and $c_{k-1}^{n-1} \neq 0$. Hence $a_k^n = c_{k-1}^{n-1}$, *i.e.*, $\mathsf{w}_{k,n}(f) = \frac{1}{2}\left(\binom{n}{k} + c_{k-1}^{n-1}\right)$.
  - if k is even, then $k \preceq n-1$ and $k-1 \npreceq n-1$. Then $b_k^{n-1} \neq 0$ and $c_{k-1}^{n-1} = 0$. Then, $a_k^n = b_k^{n-1}$, *i.e.*, $\mathsf{w}_{k,n}(f) = \frac{1}{2}\left(\binom{n}{k} + b_k^{n-1}\right)$.

Hence, $f \in \mathcal{B}_n$ is a WAPB Boolean function with $\mathsf{w}_{k,n}(f) = \dfrac{\binom{n}{k} + a_k^n}{2}$ where

$$a_k^n = \begin{cases} 0 & \textit{if } k \npreceq n, \\ b_k^{n-1} & \textit{if } k \preceq n, k < n \textit{ and } k \textit{ is even}, \\ c_{k-1}^{n-1} & \textit{if } k \preceq n, k \leq n \textit{ and } k \textit{ is odd}. \end{cases}$$

□

In general, Lemma 5.2.2 does not hold when n is an even integer. If n is even (*i.e.*, $n-1$ is odd) then there may exist some $k \in [0, n-1]$ such that $k \preceq n-1$ and $k-1 \preceq n-1$. For example, $5 \preceq 7$ and $4 \preceq 7$ for $n = 8$. As a result, there may exist WAPB functions $g, h \in \mathcal{B}_{n-1}$ such that $\mathsf{w}_{k,n}(f) = \frac{1}{2}(\binom{n}{k} \pm 2)$ as shown in Equation 5.2 in the proof of Lemma 5.2.2. This deviation of $\mathsf{w}_{k,n}(f)$ implies that f is not a WAPB function.

Corollary 5.2.3. *Let $n > 1$ be an odd integer. Let $g, h \in \mathcal{B}_{n-1}$ be two CWAPB Boolean functions defined in Definition 3.2.3. Then the function constructed by Lemma 5.2.2 is a CWAPB if $b_k^{n-1} = c_k^{n-1}$ for $0 \leq k \leq n-1$, where $\mathsf{w}_{k,n-1}(g) = \frac{\binom{n-1}{k}+b_k^{n-1}}{2}$ and $\mathsf{w}_{k,n-1}(h) = \frac{\binom{n-1}{k}+c_k^{n-1}}{2}$.*

Proof. From Lemma 5.2.2, we have

$$a_k^n = \begin{cases} 0 & \text{if } k \npreceq n \\ b_k^{n-1} & \text{if } k \preceq n; k < n \text{ and } k \text{ is even} \\ c_{k-1}^{n-1} & \text{if } k \preceq n; k \leq n \text{ and } k \text{ is odd.} \end{cases}$$

Since both f and g are CWAPB Boolean functions, $b_k^{n-1} = -b_{n-k-1}^{n-1}$ and $c_k^{n-1} = -c_{n-k-1}^{n-1}$. For $k \in [0, n]$, consider the following cases.

- $k \preceq n$ and k is even (*i.e.*, $n-k$ is odd). Then $a_k^n = b_k^{n-1} = c_k^{n-1} = -c_{n-1-k}^{n-1} = -a_{n-k}^n$.
- $k \preceq n$ and k is odd (*i.e.*, $n-k$ is even). Then $a_k^n = c_{k-1}^{n-1} = b_{k-1}^{n-1} = -b_{n-k}^{n-1} = -a_{n-k}^n$.
- $k \npreceq n$. Then $n-k \npreceq n$ and hence, $a_k^n = 0 = -a_{n-k}^n$.

Hence, f is a CWAPB. □

Further, the following is a construction of a (2^m+1)-variable WAPB Boolean function from two 2^m-variable WPB Boolean functions.

Corollary 5.2.4. *Let $n = 2^m \geq 2$ and $g, h \in \mathcal{B}_n$ be two WPB Boolean functions. Then $f \in \mathcal{B}_{n+1}$ such that $f(x_1, \ldots, x_{n+1}) = (1 + x_{n+1})g(x_1, \ldots, x_n) + x_{n+1}h(x_1, \ldots, x_n)$ is a WAPB Boolean function.*

If $g = h$, then Lemma 5.2.2 is a case of the construction proposed in Proposition 6.1.2 (refer to Chapter 6) for $n = 2^m + 1$. Further, if we take $h = 1 + g$ in Lemma 5.2.2 then the following corollary is useful for our construction.

Corollary 5.2.5. *Let $n > 1$ be an odd integer and $f_{n-1} \in \mathcal{B}_{n-1}$ be a WAPB Boolean function. Then $f_n \in \mathcal{B}_n$ defined as*

$$\begin{aligned} f_n(x_1, x_2, \ldots, x_n) &= (1 + x_n)f_{n-1}(x_1, \ldots, x_{n-1}) + x_n(1 + f_{n-1}(x_1, \ldots, x_{n-1})) \\ &= x_n + f_{n-1}(x_1, x_2, \ldots, x_{n-1}), \end{aligned}$$

i.e., $\mathsf{supp}(f_n) = \{(x, 0) \in \mathbb{F}_2^n : x \in \mathsf{supp}(f_{n-1})\} \cup \{(x, 1) \in \mathbb{F}_2^n : x \notin \mathsf{supp}(f_{n-1})\}$

is a WAPB Boolean function.

In the above corollary, if $\mathsf{w}_{k,n}(f_n) = \frac{\binom{n}{k}+a_k^n}{2}$ and $\mathsf{w}_{k,n-1}(f_{n-1}) = \frac{\binom{n-1}{k}+a_k^{n-1}}{2}$, then

$$a_k^n = \begin{cases} 0 & \textit{if } k \npreceq n, \\ a_k^{n-1} & \textit{if } k \preceq n, k < n \textit{ and } k \textit{ is even}, \\ -a_{k-1}^{n-1} & \textit{if } k \preceq n, k \leq n \textit{ and } k \textit{ is odd}. \end{cases}$$

The nonlinearity and weightwise nonlinearity of the Siegenthaler's construction are already discussed in Proposition 2.2.11 and Theorem 3.2.4, respectively. Therefore, the weightwise nonlinearity of the function f_n (defined in Corollary 5.2.5) satisfies $\mathsf{NL}_k(f_n) \geq \mathsf{NL}_k(f_{n-1}) + \mathsf{NL}_{k-1}(f_{n-1})$ for $k \in [1, n-1]$. Hence, for $k \in \{0, 1, 2, n-2, n-1, n\}$, we have $\mathsf{NL}_0(f_n) = \mathsf{NL}_n(f_n) = 0$, $\mathsf{NL}_1(f_n) = \mathsf{NL}_{n-1}(f_n) = 0$, $\mathsf{NL}_2(f_n) \geq \mathsf{NL}_2(f_{n-1})$, $\mathsf{NL}_{n-2}(f_n) \geq \mathsf{NL}_{n-3}(f_{n-1})$.

5.3 An extension of Mesnager and Su's WPB Construction in [77]

The WPB Boolean function construction presented in Proposition 5.3.1 is vital for our study, as we develop a generalized version of this approach. In 2021, Sihem Mesnager and Sihong Su introduced this construction method for generating WPB functions in [77]. Let the symmetric difference between two sets be denoted by $\triangle$, then for two Boolean functions $f, g \in \mathcal{B}_n$, we have $\mathsf{supp}(f+g) = \mathsf{supp}(f)\triangle\mathsf{supp}(g)$.

Proposition 5.3.1. *[77] For a positive integer $n = 2^m$, the support of $f_n \in \mathcal{B}_n$ is defined by*

$$\begin{aligned}\mathsf{supp}(f_n) =& \triangle_{i=1}^{m}\{(x,y,x,y,\ldots,x,y) \in \mathbb{F}_2^n : x,y \in \mathbb{F}_2^{2^{m-i}}, \mathsf{w}_{\mathsf{H}}(x) \textit{ is odd}\}.\\ =& \begin{cases}\{(1,y) : y \in \mathbb{F}_2\} & \textit{if } n = 2,\\ \{(x,y) : x,y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w}_{\mathsf{H}}(x) \textit{ is odd}\}\triangle\{(x,x) : x \in \mathsf{supp}(f_{\frac{n}{2}})\} & \textit{if } n > 2.\end{cases}\end{aligned}$$

Then the Boolean function f_n is WPB.

Hence, the following result generalizes the Proposition 5.3.1, providing a method to construct a WAPB Boolean function by lifting the support of a known WAPB function defined in dimention n_0.

Lemma 5.3.2. *Let $n = n_0 2^m$ where n_0 is an odd positive integer and $m \geq 0$ is an integer. Let $f_{n_0} \in \mathcal{B}_{n_0}$ be a WAPB Boolean function. Then $f_n \in \mathcal{B}_n$, recursively defined as*

$$\mathsf{supp}(f_n) = \begin{cases}\mathsf{supp}(f_{n_0}) & \textit{if } n = n_0,\\ \{(x,y) : x,y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w}_{\mathsf{H}}(x) \textit{ is odd}\}\triangle\{(z,z) : z \in \mathsf{supp}(f_{\frac{n}{2}})\}, & \textit{if } n > n_0,\end{cases}$$

is a WAPB Boolean function.

Proof. The proof follows the idea of the proof of [77, Theorem 3]. As f_{n_0} is a WAPB Boolean function, $\mathsf{w}_{k,n_0}(f_{n_0}) = \dfrac{\binom{n_0}{k} + a_k^{n_0}}{2}$ with $a_k^{n_0} \in \{0, \pm 1\}$ for $k \in [0, n_0]$. Now, we shall prove that $\mathsf{w}_{k,n}(f_n) = \dfrac{\binom{n}{k} + a_k^n}{2}$ where $a_k^n = \begin{cases} 0 & \textit{if}\, k \npreceq n, \\ \pm 1 & \textit{if}\, k \preceq n. \end{cases}$ for $k \in [0, n]$. If $k = 0$, $a_k^n = a_k^{\frac{n}{2}} = a_k^{n_0}$. So it is satisfied for $k = 0$. Now consider, $k \in [1, n]$, which can be factored as $k = k_0 2^b$ where k_0 is odd and $b \geq 0$ is an integer. It will be proved in two different cases *i.e.*, for $m > b$ and for $m \leq b$.

If $m > b$ (in this case $k \npreceq n$), then

$$
\begin{aligned}
\mathsf{supp}_k(f_n) &= \{(x,y) : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = k\} \\
&\quad \triangle \{(z,z) : z \in \mathsf{supp}_{\frac{k}{2}}(f_{\frac{n}{2}})\} \\
&= \{(x,y) : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = k\} \\
&\quad \triangle \{(x,y,x,y) : x, y \in \mathbb{F}_2^{\frac{n}{2^2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = \frac{k}{2}\} \\
&\quad \triangle \{(z,z,z,z) : z \in \mathsf{supp}_{\frac{k}{2^2}}(f_{\frac{n}{2^2}})\} \quad \text{(as $\triangle$ operation is associative)} \\
&= \{(x,y) : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = k\} \\
&\quad \triangle \{(x,y,x,y) : x, y \in \mathbb{F}_2^{\frac{n}{2^2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = \frac{k}{2}\} \\
&\quad \vdots \\
&\quad \triangle \{(x,y,x,y,\ldots,x,y) : x, y \in \mathbb{F}_2^{\frac{n}{2^b}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = \frac{k}{2^{b-1}}\} \\
&\quad \triangle \{(z,z,\ldots,z,z) : z \in \mathsf{supp}_{\frac{k}{2^b}}(f_{\frac{n}{2^b}})\} \\
&= \{(x,y) : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = k\} \qquad (S_1) \\
&\quad \triangle \{(x,y,x,y) : x, y \in \mathbb{F}_2^{\frac{n}{2^2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = \frac{k}{2}\} \qquad (S_2) \\
&\quad \vdots \\
&\quad \triangle \{(x,y,\ldots,x,y) : x, y \in \mathbb{F}_2^{\frac{n}{2^b}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = \frac{k}{2^{b-1}}\} \qquad (S_b) \\
&\quad \triangle \{(z,z\ldots,z) : z \in \mathbb{F}_2^{\frac{n}{2^b}}, \mathsf{w_H}(z) = \frac{k}{2^b}\} \qquad (S_{b+1})
\end{aligned}
$$

Here, we stop at S_{b+1} as $\frac{k}{2^{b+1}}$ is not an integer. As $\mathsf{w_H}(x)$ is odd in the set S_1 and $\mathsf{w_H}(x,y) = \frac{k}{2}$ is even in the set S_2 in the above identity, the sets S_1 and S_2 are disjoint. Similarly, we can check that the set S_3 is pairwise disjoint with the sets S_1 and S_2. Continuing the argument, we have that the set $S_j, j \in [2, b]$ is pairwise disjoint with $S_1, S_2, \ldots, S_{j-1}$, *i.e.*, the sets $S_j, j \in [1, b]$ are mutually disjoint. Further, since $\mathsf{w_H}(z) = \frac{k}{2^b} = k_0$ is odd in S_{b+1}, then $S_{b+1} \subset S_b$. Therefore,

$$\mathsf{w}_{k,n}(f_n) = |S_1| + |S_2| + \cdots + |S_b| - |S_{b+1}|.$$

Here, we can check that

$$\begin{aligned}
|S_j| &= |\{(x, y, x, y, \ldots, x, y) : x, y \in \mathbb{F}_2^{\frac{n}{2^j}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x, y) = \frac{k}{2^{j-1}}\}| \\
&= \sum_{\substack{i=1 \\ i \text{ is odd}}}^{\frac{k}{2^{j-1}}} \binom{\frac{n}{2^j}}{i} \binom{\frac{n}{2^j}}{\frac{k}{2^{j-1}} - i} \\
&= \begin{cases}
\frac{1}{2}\binom{\frac{n}{2^{j-1}}}{\frac{k}{2^{j-1}}} - \frac{1}{2}\binom{\frac{n}{2^j}}{\frac{k}{2^j}} & \textit{if } j \in [1, b-1] \quad \textit{(using Lemma 5.1.1[Item 2b])}, \\[2ex]
\frac{1}{2}\binom{\frac{n}{2^{b-1}}}{\frac{k}{2^{b-1}}} + \frac{1}{2}\binom{\frac{n}{2^b}}{\frac{k}{2^b}} & \textit{if } j = b \quad \textit{(using Lemma 5.1.1[Item 2a])}, \\[2ex]
\frac{1}{2}\binom{\frac{n}{2^b}}{\frac{k}{2^b}} & \textit{if } j = b+1 \quad \textit{(using Lemma 5.1.1[Item 1])}.
\end{cases}
\end{aligned}$$

Therefore,

$$\mathsf{w}_{k,n}(f_n) = \frac{1}{2}\left[\sum_{j=1}^{b-1}\left(\binom{\frac{n}{2^{j-1}}}{\frac{k}{2^{j-1}}} - \binom{\frac{n}{2^j}}{\frac{k}{2^j}}\right) + \binom{\frac{n}{2^{b-1}}}{\frac{k}{2^{b-1}}} + \binom{\frac{n}{2^b}}{\frac{k}{2^b}} - \binom{\frac{n}{2^b}}{\frac{k}{2^b}}\right] = \frac{1}{2}\binom{n}{k}.$$

If $m \leq b$, then using similar process as the above case we have

$$\begin{aligned}
\mathsf{supp}_k(f_n) &= \{(x, y) : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x, y) = k\} \\
&\quad \triangle\{(x, y, x, y) : x, y \in \mathbb{F}_2^{\frac{n}{2^2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x, y) = \frac{k}{2}\} \\
&\quad \vdots \\
&\quad \triangle\{(x, y, \ldots, x, y) : x, y \in \mathbb{F}_2^{\frac{n}{2^m}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x, y) = \frac{k}{2^{m-1}}\} \\
&\quad \triangle\{(z, z, \ldots, z, z) : z \in \mathsf{supp}_{\frac{k}{2^m}}(f_{\frac{n}{2^m}})\}
\end{aligned}$$

$$\begin{aligned}
&= \{(x,y) : x,y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = k\} && (T_1)\\
&\quad \triangle\{(x,y,x,y) : x,y \in \mathbb{F}_2^{\frac{n}{2^2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = \frac{k}{2}\} && (T_2)\\
&\quad \vdots\\
&\quad \triangle\{(x,y,\dots,x,y) : x,y \in \mathbb{F}_2^{\frac{n}{2^m}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = \frac{k}{2^{m-1}}\} && (T_m)\\
&\quad \triangle\{(z,z,\dots,z,z) : z \in \mathsf{supp}_{\frac{k}{2^m}}(f_{n_0})\} && (T_{m+1})
\end{aligned}$$

It can be checked (as the earlier way) that the sets $T_1, T_2, \dots, T_m$ are pairwise disjoint.

If $\frac{k}{2^m}$ is even (*i.e.*, $b > m$), then $\mathsf{w_H}(z)$ is even in T_{m+1}. So, T_{m+1} is too disjoint with $T_i, i \in [1,m]$. Hence,

$$\mathsf{w}_{k,n}(f_n) = |T_1| + |T_2| + \cdots + |T_m| + |T_{m+1}|.$$

Here,

$$|T_j| = \begin{cases} \frac{1}{2}\left(\binom{\frac{n}{2^{j-1}}}{\frac{k}{2^{j-1}}} - \binom{\frac{n}{2^j}}{\frac{k}{2^j}}\right) & \textit{if } j \in [1,m] \quad \textit{(using Lemma 5.1.1[Item 2b])},\\ \frac{1}{2}\left(\binom{\frac{n}{2^m}}{\frac{k}{2^m}} + a^{n_0}_{\frac{k}{2^m}}\right) & \textit{if } j = m+1 \quad \textit{(as } f_{n_0} \textit{ is WAPB with}\\ & \qquad \mathsf{w}_{l,n_0}(f_{n_0}) = \frac{1}{2}\left(\binom{n_0}{l} + a^{n_0}_l\right)). \end{cases}$$

Hence,

$$\mathsf{w}_{k,n}(f_n) = \frac{1}{2}\sum_{j=1}^{m}\left(\binom{\frac{n}{2^{j-1}}}{\frac{k}{2^{j-1}}} - \frac{1}{2}\binom{\frac{n}{2^j}}{\frac{k}{2^j}}\right) + \frac{1}{2}\left(\binom{\frac{n}{2^m}}{\frac{k}{2^m}} + a^{n_0}_{\frac{k}{2^m}}\right) = \frac{1}{2}\left(\binom{n}{k} + a^{n_0}_{\frac{k}{2^m}}\right).$$

Further, if $\frac{k}{2^m}$ is odd (*i.e.*, $b = m$), then $\mathsf{w_H}(z)$ is odd in T_{m+1}. As $\frac{k}{2^{m-1}}$ is even, $\mathsf{w_H}(x)$ and $\mathsf{w_H}(y)$ are also odd in the set T_m. Hence, $T_{m+1} \subset T_m$. That implies,

$$\mathsf{w}_{k,n}(f_n) = |T_1| + |T_2| + \cdots + |T_m| - |T_{m+1}|.$$

Here,

$$|T_j| = \begin{cases} \frac{1}{2}\binom{\frac{n}{2^{j-1}}}{\frac{k}{2^{j-1}}} - \frac{1}{2}\binom{\frac{n}{2^j}}{\frac{k}{2^j}} & \textit{if } j \in [1, m-1] \quad \textit{(using Lemma 5.1.1[Item 2b]),} \\ \frac{1}{2}\binom{\frac{n}{2^{m-1}}}{\frac{k}{2^{m-1}}} + \frac{1}{2}\binom{\frac{n}{2^m}}{\frac{k}{2^m}} & \textit{if } j = m \quad \textit{(using Lemma 5.1.1[Item 2a]),} \\ \frac{1}{2}\left(\binom{\frac{n}{2^m}}{\frac{k}{2^m}} + a^{n_0}_{\frac{k}{2^m}}\right) & \textit{if } j = m+1 \quad \textit{(as } f_{n_0} \textit{ is WAPB with} \\ & \qquad \mathsf{w}_{l,n_0}(f_{n_0}) = \frac{1}{2}\left(\binom{n_0}{k} + a^{n_0}_l\right)\textit{).} \end{cases}$$

So, $\mathsf{w}_{k,n}(f_n) = \frac{1}{2}\sum_{j=1}^{m-1}\left(\binom{\frac{n}{2^{j-1}}}{\frac{k}{2^{j-1}}} - \frac{1}{2}\binom{\frac{n}{2^j}}{\frac{k}{2^j}}\right) + \frac{1}{2}\left(\binom{\frac{n}{2^{m-1}}}{\frac{k}{2^{m-1}}} + \binom{\frac{n}{2^m}}{\frac{k}{2^m}}\right) - \frac{1}{2}\left(\binom{\frac{n}{2^m}}{\frac{k}{2^m}} + a^{n_0}_{\frac{k}{2^m}}\right)$
$= \frac{1}{2}\left(\binom{n}{k} - a^{n_0}_{\frac{k}{2^m}}\right)$.

Therefore, for $n = n_0 2^m$ and $k = k_0 2^b$, we got that $\mathsf{w}_{k,n}(f_n) = \frac{1}{2}\left(\binom{n}{k} + a^n_k\right)$, where

$$a^n_k = \begin{cases} 0 & \textit{if } m > b, \\ a^{n_0}_{\frac{k}{2^m}} & \textit{if } m < b, \\ -a^{n_0}_{\frac{k}{2^m}} & \textit{if } m = b, \end{cases} \qquad \text{and} \qquad a^n_0 = a^{n_0}_0.$$

Hence, f_n is a WAPB Boolean function if f_{n_0} is a WAPB Boolean function. □

As a result, Corollary 5.2.5 and Lemma 5.3.2 present a general construction for WAPB Boolean functions in any number of variables in the following theorem.

Theorem 5.3.3. *For $n \geq 2$, the support of a Boolean function $f_n \in \mathcal{B}_n$ defined as*

$$\mathsf{supp}(f_n) = \begin{cases} \{(x,1) \in \mathbb{F}_2^2 : x \in \mathbb{F}_2\} = \{(0,1),(1,1)\} & \textit{if } n = 2, \\ \{(x,0) \in \mathbb{F}_2^n : x \in \mathsf{supp}(f_{n-1})\} & \\ \qquad \cup\{(x,1) \in \mathbb{F}_2^n : x \notin \mathsf{supp}(f_{n-1})\} & \textit{if } n > 2 \textit{ and odd,} \\ \{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}\} & \\ \qquad \triangle\{(z,z) \in \mathbb{F}_2^n : z \in \mathsf{supp}(f_{\frac{n}{2}})\} & \textit{if } n > 2 \textit{ and even,} \end{cases} \tag{5.3}$$

is a WAPB Boolean function.

When $n = 2^m$, is a power of 2, we get the WPB Boolean function presented in [77]. The base Boolean function used in the above recursive construction (*i.e.*, f_2) is linear. As a result, the nonlinearity of the destined Boolean function remains weak. The construction in Theorem 5.3.3 can be generalized by beginning with a base WAPB Boolean function on a higher number of variables. As a result, the final function can have better cryptographic properties than the function that resulted in Theorem 5.3.3.

Theorem 5.3.4. *For $p \geq 2$, let f_p be a WAPB Boolean function. Let n be a positive integer such that*

$$p = \lfloor \frac{n}{2^m} \rfloor \text{ i.e, } n = a_0 2^0 + a_1 2^1 + \cdots + a_{m-1} 2^{m-1} + p 2^m \tag{5.4}$$

$$\text{or, } \quad p+1 = \lfloor \frac{n}{2^m} \rfloor \text{ i.e, } n = a_0 2^0 + a_1 2^1 + \cdots + a_{m-1} 2^{m-1} + (p+1) 2^m \text{ if } p \text{ is even}, \tag{5.5}$$

where $m \geq 0$ and $a_0, a_1, \ldots, a_m \in \{0,1\}$. Then $f_n \in \mathcal{B}_n$ whose support is defined as

$$\mathsf{supp}(f_n) = \begin{cases} \mathsf{supp}(f_p) & \text{if } n = p, \\ \{(x,0) \in \mathbb{F}_2^n : x \in \mathsf{supp}(f_{n-1})\} & \\ \qquad \cup \{(x,1) \in \mathbb{F}_2^n : x \notin \mathsf{supp}(f_{n-1})\} & \text{if } n > p \text{ and odd}, \\ \{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \text{ is odd}\} & \\ \qquad \triangle \{(z,z) \in \mathbb{F}_2^n : z \in \mathsf{supp}(f_{\frac{n}{2}})\} & \text{if } n > p \text{ and even}, \end{cases} \tag{5.6}$$

is a WAPB Boolean function.

For illustration, if we have a WAPB Boolean function on 5 variables, then we can construct a WAPB Boolean function on $10, 11, 20, 21, 22, 23, 40, 41, \ldots$ variables. Similarly, if we have a WAPB Boolean function on 6 variables, then we can construct a WAPB Boolean function on $7, 12, 13, 14, 15, 24, 25, 26, 27, \ldots$ variables.

Note 5.3.5. The construction in Theorem 5.3.4 can further be generalized. When n is odd, instead of using the WAPB Boolean functions f_{n-1} and $1 + f_{n-1}$, one can use any two WAPB Boolean functions $g, h \in \mathcal{B}_{n-1}$ (following Lemma 5.2.2) to have f_n in the recursive construction presented in Equation 5.6. The nonlinearity of f_n can be improved in this way.

5.3.1 Algebraic Normal Form

Since the recursive construction involves two different liftings depending on whether n is odd or even, the algebraic normal form (ANF) of the Boolean function $f_n \in \mathcal{B}_n$ is influenced by the binary representation of n. Consequently, the ANF of f_n can be determined as described in the following theorem.

Theorem 5.3.6. *For $p \geq 2$, let f_p be a WAPB Boolean function. Let n be a positive integer such that, for an $m \geq 0$,*

- $p = \lfloor \frac{n}{2^m} \rfloor$ *i.e.,* $n = a_0 2^0 + a_1 2^1 + \cdots + a_{m-1} 2^{m-1} + p 2^m$, *or,*
- $p + 1 = \lfloor \frac{n}{2^m} \rfloor$ *i.e.,* $n = a_0 2^0 + a_1 2^1 + \cdots + a_{m-1} 2^{m-1} + (p+1) 2^m$ *if p is even.*

Then the ANF of f_n, defined in Theorem 5.3.4 is

$$f_n(x_1, \ldots, x_n) = \begin{cases} f_p & \text{if } n = p, \\ x_n + f_{n-1}(x_1, x_2, \ldots, x_{n-1}) & \text{if } n > p \text{ and odd}, \\ \sum_{i=1}^{\frac{n}{2}} x_i + f_{\frac{n}{2}}(x_1, \ldots, x_{\frac{n}{2}}) \prod_{i=1}^{\frac{n}{2}} (x_i + x_{\frac{n}{2}+i} + 1) & \text{if } n > p \text{ and even}. \end{cases} \tag{5.7}$$

Proof. If $n = p$, then by our assumption $f_n = f_p$. Now consider that $n > p$. If n is odd then the ANF of f_n is derived from the Corollary 5.2.5. Now if n is even, we prove it following the technique of the proof from [77, Theorem 4]. Here

$$\mathsf{supp}(f_n) = \{(x, y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}\} \triangle \{(z, z) \in \mathbb{F}_2^n : z \in \mathsf{supp}(f_{\frac{n}{2}})\}$$

Let denote $g_n, h_n \in \mathcal{B}_n$ such that $\mathsf{supp}(g_n) = \{(x, y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}\}$; and $\mathsf{supp}(h_n) = \{(z, z) \in \mathbb{F}_2^n : z \in \mathsf{supp}(f_{\frac{n}{2}})\}$.

Then the ANFs of g_n and h_n are

$$g_n(x_1, \ldots, x_n) = \sum_{i=1}^{\frac{n}{2}} x_i \text{ and } h_n(x_1, \ldots, x_n) = f_{\frac{n}{2}}(x_1, \ldots, x_{\frac{n}{2}}) \prod_{i=1}^{\frac{n}{2}} (x_i + x_{\frac{n}{2}+i} + 1).$$

As the XOR operation imitates the symmetric difference of the supports of two Boolean functions, we have

$$f_n(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{\frac{n}{2}} x_i + f_{\frac{n}{2}}(x_1, x_2, \ldots, x_{\frac{n}{2}}) \prod_{i=1}^{\frac{n}{2}} (x_i + x_{\frac{n}{2}+i} + 1).$$

□

For example, we shall compute the ANF of $f_{11} \in \mathcal{B}_{11}$ for a given WAPB $f_5 \in \mathcal{B}_5$. Then we shall have $f_{10}(x_1, \ldots, x_{10}) = \sum_{i=1}^{5} x_i + f_5(x_1, x_2, \ldots, x_5) \prod_{i=1}^{5}(x_i + x_{5+i} + 1)$. Then we have $f_{11}(x_1, \ldots, x_{11}) = \sum_{i=1}^{5} x_i + x_{11} + f_5(x_1, \ldots, x_5) \prod_{i=1}^{5}(x_i + x_{5+i} + 1)$.

Therefore, the WAPB Boolean functions can be computed recursively very fast although the ANF of the WAPB Boolean functions contain a lot of monomials.

The algebraic degree of f_n can be computed as follows.

Theorem 5.3.7. *Let $f_p \in \mathcal{B}_p$ be a Boolean function with $\deg(f_p) \geq 1$ and $f_n \in \mathcal{B}_n$ be the Boolean function constructed as in Theorem 5.3.4. Then the algebraic degree of f_n is $\deg(f_n) = \lfloor \frac{n}{2} \rfloor + \lfloor \frac{n}{2^2} \rfloor + \cdots + \lfloor \frac{n}{2^m} \rfloor + \deg(f_p)$. That is,*

1. $\deg(f_n) = n - (p + a_{m-1} + a_{m-2} + \cdots + a_0) + \deg(f_p)$ *if $p = \lfloor \frac{n}{2^m} \rfloor$ as in Equation 5.4;*

2. $\deg(f_n) = n - (p + 1 + a_{m-1} + a_{m-2} + \cdots + a_0) + \deg(f_p)$ *if $p + 1 = \lfloor \frac{n}{2^m} \rfloor$ as in Equation 5.5.*

Proof. If $a_0 = 1$ (*i.e.*, n is odd), then $f_n(x_1, x_2, \ldots, x_n) = x_n + f_{n-1}(x_1, x_2, \ldots, x_{n-1})$. Hence, $\deg(f_n) = \deg(f_{n-a_0})$ irrespective of $a_0 = 0$ or 1.

Further, if n is even (*i.e.*, $a_0 = 0$), then as per the construction

$$f_n(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{\frac{n}{2}} x_i + f_{\frac{n}{2}}(x_1, x_2, \ldots, x_{\frac{n}{2}}) \prod_{i=1}^{\frac{n}{2}} (x_i + x_{\frac{n}{2}+i} + 1).$$

That implies $\deg(f_n) = \frac{n}{2} + \deg(f_{\frac{n}{2}})$. Performing this process recursively, we have

$$\begin{aligned}
\deg(f_n) &= \deg(f_{n-a_0}) = \frac{n-a_0}{2} + \deg(f_{\frac{n-a_0}{2}}) = \lfloor \frac{n}{2} \rfloor + \deg(f_{\lfloor \frac{n}{2} \rfloor}) \\
&= \lfloor \frac{n}{2} \rfloor + \deg(f_{\lfloor \frac{n}{2} \rfloor - a_1}) = \lfloor \frac{n}{2} \rfloor + \frac{\lfloor \frac{n}{2} \rfloor - a_1}{2} + \deg(f_{\frac{\lfloor \frac{n}{2} \rfloor - a_1}{2}}) \\
&= \lfloor \frac{n}{2} \rfloor + \lfloor \frac{n}{2^2} \rfloor + \deg(f_{\lfloor \frac{n}{2^2} \rfloor}) \\
&\vdots \\
&= \lfloor \frac{n}{2} \rfloor + \lfloor \frac{n}{2^2} \rfloor + \cdots + \lfloor \frac{n}{2^m} \rfloor + \deg(f_{\lfloor \frac{n}{2^m} \rfloor}) \\
&= \lfloor \frac{n}{2} \rfloor + \lfloor \frac{n}{2^2} \rfloor + \cdots + \lfloor \frac{n}{2^m} \rfloor + \deg(f_p) \\
&\qquad\qquad\qquad (\text{if } p \text{ is even, still } \deg(f_{p+1}) = \deg(f_p)).
\end{aligned}$$

If $p = \lfloor \frac{n}{2^m} \rfloor$ (as Equation 5.4), $\lfloor \frac{n}{2^k} \rfloor = p2^{m-k} + a_{m-1}2^{m-1-k} + \cdots + a_k$ for $0 \leq k \leq m$. Then, substituting $\lfloor \frac{n}{2^k} \rfloor$ for $1 \leq k \leq m$ in the above equation we shall have $\deg(f_n) = n - (p + a_{m-1} + a_{m-2} + \cdots + a_0) + \deg(f_p)$. Similarly, for the case as in Equation 5.5, we shall have $\deg(f_n) = n - (p + 1 + a_{m-1} + a_{m-2} + \cdots + a_0) + \deg(f_p)$. □

The following corollary presents the degree of f_n (constructed as in Theorem 5.3.3) for some special values of n.

Corollary 5.3.8. *For $n \geq 2$, consider the WAPB Boolean function $f_n \in \mathcal{B}_n$ as in Theorem 5.3.3. The degree of f_n is*

1. *$n - 1$ if $n = 2^k$ for some $k \geq 1$ (i.e., n is a power of 2) [77, Corollary 3];*
2. *$n - k$ if $n = 2^k - 1$ for some $k \geq 2$.*

Proof. 1. Here $p = 2$ as per Theorem 5.3.3 and $n = 2^k$. That is, $m = k - 1$ and $a_{m-1} = \cdots = a_0 = 0$. So $\deg(f_n) = n - p + \deg(f_2) = n - 2 + 1 = n - 1$.

2. Here $p = 2$ as per Theorem 5.3.3 and $n = 2^k - 1 = (2+1)2^{k-2} + 2^{k-3} + \cdots + 2^1 + 1$. That implies, $m = k - 2$ and $a_{m-1} = \cdots = a_0 = 1$. So $\deg(f_n) = n - (p + 1 + (a_{m-1} + \cdots + a_0)) + \deg(f_2) = n - (2 + 1 + m) + 1 = n - 2 - (k - 2) = n - k$.

□

5.3.2 Nonlinearity

Nonlinearity is a crucial cryptographic property of Boolean functions. In this subsection, we analyze the nonlinearity of the proposed WAPB Boolean functions. We begin by examining the nonlinearity of the function constructed in Proposition 5.3.1, and then conclude with the nonlinearity analysis of the resulting function f_n.

Lemma 5.3.9. *Let $n \geq 4$ be an even integer and $f_n \in \mathcal{B}_n$ such that*

$$\mathsf{supp}(f_n) = \{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}\} \triangle \{(z,z) \in \mathbb{F}_2^n : z \in \mathsf{supp}(f_{\frac{n}{2}})\}$$

where $f_{\frac{n}{2}} \in \mathcal{B}_{\frac{n}{2}}$. Then $\mathsf{NL}(f_n) = \mathsf{w_H}(f_{\frac{n}{2}})$.

Proof. Consider $g, h \in \mathcal{B}_n$ with $\mathsf{supp}(g) = \{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}\}$ and $\mathsf{supp}(h) = \{(z,z) \in \mathbb{F}_2^n : z \in \mathsf{supp}(f_{\frac{n}{2}})\}$. Then $f_n = g + h$ and $g(x_1, x_2, \ldots, x_n) = x_1 + x_2 + \cdots + x_{\frac{n}{2}}$ is a linear function. Therefore, $\mathsf{d_H}(f_n, g) = \mathsf{w_H}(h) = |\{(z,z) \in \mathbb{F}_2^n : z \in \mathsf{supp}(f_{\frac{n}{2}})\}| = \mathsf{w_H}(f_{\frac{n}{2}})$. That implies, $\mathsf{NL}(f_n) \leq \mathsf{w_H}(f_{\frac{n}{2}})$.

Let $l_{a,b} \in \mathcal{B}_n$ be an affine function (other than g) such that $\mathsf{d_H}(f_n, l_{a,b}) < \mathsf{w_H}(f_{\frac{n}{2}})$. Then $\mathsf{d_H}(g, l_{a,b}) \leq \mathsf{d_H}(g, f_n) + \mathsf{d_H}(f_n, l_{a,b}) < 2\mathsf{w_H}(f_{\frac{n}{2}}) \leq 2^{\frac{n}{2}+1}$. Since g and $l_{a,b}$ are both affine functions, $\mathsf{d_H}(g, l_{a,b}) = 2^{n-1}$ (or, 2^n if $l_{a,b} = 1 + g$). Hence, $\mathsf{NL}(f_n) = \mathsf{w_H}(f_{\frac{n}{2}})$. □

If $f_{\frac{n}{2}}$ is a balanced function, then $\mathsf{NL}(f_n) = \mathsf{w_H}(f_{\frac{n}{2}}) = 2^{\frac{n}{2}-1}$. Therefore, the proposed construction has very poor nonlinearity. This happens due to the addition of the linear part $\{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}\}$. Moreover, we observe a limiting result regarding the weightwise nonlinearity.

Corollary 5.3.10. *If n is even and $n > p$ then $\mathsf{NL}_k(f_n) = 0$ for all odd integer $k \in [0, n]$.*

Proof. Consider the functions g and h as in the proof of Lemma 5.3.9. Then $\mathsf{supp}_k(h) = \emptyset$ for every odd $k \in [0, n]$. Therefore, $f_n = g$ is a linear function and $\mathsf{NL}_k(f_n) = 0$ for every odd $k \in [0, n]$. □

Table 5.7 summarizes the nonlinearity and weightwise nonlinearity of the function f_n constructed using Theorem 5.3.3, for $8 \leq n \leq 16$.

Since $\mathsf{NL}_0(f_n) = \mathsf{NL}_1(f_n) = \mathsf{NL}_{n-1}(f_n) = \mathsf{NL}_n(f_n) = 0$, these trivial values are omitted from the table. The global weightwise nonlinearity of f is defined as $\mathtt{GWNL}(f) = \sum_{k=0}^{n} \mathsf{NL}_k(f)$ (refer [43]).

n	NL	NL_2	NL_3	NL_4	NL_5	NL_6	NL_7	NL_8	NL_9	NL_{10}	NL_{11}	NL_{12}	NL_{13}	NL_{14}	GWNL
8	8	2	0	3	0	2	0	0	0	-	-	-	-	-	7
9	16	2	2	3	3	2	2	0	0	-	-	-	-	-	16
10	16	3	0	5	0	5	0	3	0	0	-	-	-	-	16
11	32	3	3	5	5	5	5	3	3	0	0	-	-	-	32
12	32	3	0	7	0	10	0	8	0	3	0	0	-	-	31
13	64	3	3	7	7	10	10	8	8	3	3	0	0	-	62
14	64	4	0	10	0	18	0	18	0	10	0	4	0	-	64
15	128	4	4	10	10	18	18	18	18	10	10	4	4	0	128
16	128	4	0	14	0	28	0	35	0	14	0	4	0	28	127

Table 5.1: Listing of $\mathsf{NL}(f_n), \mathsf{NL}_k(f_n), \mathtt{GWNL}(f_n)$ for $8 \leq n \leq 16$.

5.3.3 Algebraic Immunity

In this subsection, we analyze the algebraic immunity of the Boolean function f_n, as defined in Equation 5.3 and Equation 5.6) along with its algebraic immunity over the restricted domain $\mathsf{E}_{k,n}$ for $k \in [0, n]$.

Theorem 5.3.11. *Consider $n \geq 2$ and $f_n \in \mathcal{B}_n$ defined in Equation 5.6 with $n > p$. Then,*

1. $\mathsf{AI}(f_n) = 2$ *if n is even;*

2. $\mathsf{AI}(f_n) \leq 1 + \mathsf{AI}(f_{n-1})$ *if n is odd.*

Proof. For n even, the ANF of f_n (see Theorem 5.3.6) is given by

$$f_n(x) = \sum_{i=1}^{\frac{n}{2}} x_i + f_{\frac{n}{2}}(x_1, x_2, \ldots, x_{\frac{n}{2}}) \prod_{i=1}^{\frac{n}{2}} (x_i + x_{\frac{n}{2}+i} + 1) \text{ where } x = (x_1, \ldots, x_n) \in \mathbb{F}_2^n.$$

Then $g(x) = (1 + \sum_{i=1}^{\frac{n}{2}} x_i)(x_1 + x_{\frac{n}{2}+1})$ is an annihilator f_n. Hence, $\mathsf{AI}(f_n) \leq \deg(g) = 2$. Furthermore, using a similar technique for proofing WAPB Boolean functions as in [45, Theorem 1] for WPB Boolean functions, we have $\mathsf{AI}(f_n) \geq 2$. Hence, $\mathsf{AI}(f_n) = 2$.

For n odd, the ANF of f_n (see Theorem 5.3.6) is given by
$f_n(x) = x_n + f_{n-1}(x_1, \ldots, x_{n-1})$. By [26, Corollary 3], $\mathsf{AI}(f_n) \leq 1 + \mathsf{AI}(f_{n-1})$. □

If n is odd and $n > p + 1$, then $\mathsf{AI}(f_n) \leq 1 + \mathsf{AI}(f_{n-1}) \leq 1 + 2 = 3$.

Corollary 5.3.12. *Consider $f_n \in \mathcal{B}_n, n \geq 2$ as in Equation 5.6 with $n > p + 1$. Then* $\mathsf{AI}(f_n) \leq 3$.

Theorem 5.3.13. *Consider $n \geq 2$ and $f_n \in \mathcal{B}_n$ defined in Equation 5.6 with $n > p$. Then,*

1. $\mathsf{AI}_k(f_n) = 1$ *if n is even and k is odd;*

2. $\mathsf{AI}_k(f_n) \leq 2$ *if n is even and k is even;*

3. $\mathsf{AI}_k(f_n) \leq 1 + \mathsf{AI}_k(f_{n-1})$ *if n is odd;*

Proof. For n even, $\mathsf{supp}(f_n) = \{(x, y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}\} \triangle \{(z, z) \in \mathbb{F}_2^n : z \in \mathsf{supp}(f_{\frac{n}{2}})\}$ (see Equation 5.6). If k is odd, the weight of vectors in $\mathsf{E}_{k,n}$ are odd and hence $\{(z, z) \in \mathbb{F}_2^n : z \in \mathsf{supp}(f_{\frac{n}{2}})\}$ is an empty set for k odd. As a result, $f_n = \sum_{i=1}^{\frac{n}{2}} x_i$ over $\mathsf{E}_{k,n}$. Then, $g(x) = 1 + \sum_{i=1}^{\frac{n}{2}} x_i$ is an annihilator of f_n over $\mathsf{E}_{k,n}$ as $g(x) \neq 0$ for some $x \in \mathsf{E}_{k,n}$. It can be easily checked that f_n is not a constant function over $\mathsf{E}_{k,n}$ for every $1 \leq k \leq n$ by taking an input vector $x \in \mathbb{F}_2^n$ with odd (respectively, even) number of 1s in $x_1, x_2, \ldots, x_{\frac{n}{2}}$. So, f and $1 + f$ have no constant annihilator. Hence, $\mathsf{AI}_k(f_n) = 1$ if n is even and k is odd.

If k is even, consider $g(x) = (1 + \sum_{i=1}^{\frac{n}{2}} x_i)(x_1 + x_{\frac{n}{2}+1})$. It can also be checked that there are some $x \in \mathsf{E}_{k,n}$ such that $g(x) \neq 0$. Since $g(x)$ is an annihilator of $f_n(x)$, $g(x)$ is an annihilator of $f_n(x)$ over $\mathsf{E}_{k,n}$. Hence, $\mathsf{AI}_k(f_n) = 2$ if n is even and k is even.

Now consider n is odd. Then $f_n(x) = x_n + f_{n-1}(x_1, \ldots, x_{n-1})$ and $g(x) = (1 + x_n)h(x_1, \ldots, x_{n-1})$ is an annihilator of f_n where $h(x_1, \ldots, x_{n-1}) \in Ann(f_{n-1})$. If we choose h as an annihilator of f_{n-1} in the first part of the proof, $h(x_1, \ldots, x_{n-1}) \neq 0$ for some $(x_1, \ldots, x_{n-1}) \in \mathsf{E}_{k,n-1}$. Hence, $g(x_1, \ldots, x_n) \neq 0$ for some $(x_1, \ldots, x_n) \in \mathsf{E}_{k,n}$ by adding $x_n = 0$ with those vectors. As a result, we have $\mathsf{AI}_k(f_n) \leq 1 + \mathsf{AI}_k(f_{n-1})$. □

5.4 A modification of WAPB function f_n for higher nonlinearity

The function $f_n \in \mathcal{B}_n$ proposed in Theorem 5.3.3 and Theorem 5.3.4 are WAPB Boolean functions and fast computable. Unfortunately, these are not having good nonlinearity and algebraic immunity. The main reasons for not having these essential cryptographic properties are the following.

1. When n is odd, the construction uses a concatenation of f_{n-1} and $1 + f_{n-1}$.

2. When n is even, the construction modifies only a few support vectors of a linear function $\sum_{i=1}^{\frac{n}{2}} x_i$. Consequently, both nonlinearity and algebraic immunity remain low over entire space $\mathbb{F}_2^n$ as well as restricted domains $\mathsf{E}_{k,n}$. Moreover, it holds that $f_n(x_1, \ldots, x_n) = \sum_{i=1}^{\frac{n}{2}} x_i$ on $\mathsf{E}_{k,n}$ when k is odd (see the proof of Corollary 5.3.10). As a result, $\mathsf{NL}_k(f_n) = 0$ forall odd k.

Our objective is to enhance the nonlinearities $\mathsf{NL}(f_n), \mathsf{NL}_k(f_n)$, and the algebraic immunities $\mathsf{AI}(f_n), \mathsf{AI}_k(f_n)$. Specifically, since the function f_n is linear on the restricted domain $\mathsf{E}_{k,n}$ when n is even and k is odd, we focus on modifying this behavior. Therefore, we first propose modifying $\mathsf{supp}(f_n)$ in $\mathsf{E}_{k,n}$ for n even and k odd. This leads us to define a new class of WAPB Boolean functions denoted as F_n, obtained by modifying the $\mathsf{supp}(f_n)$ as described in Theorem 5.3.4 and again formalized in the following lemma. The modification involves swapping of the support vectors (x, y) with (y, x) where $x, y \in \mathbb{F}_2^{\frac{n}{2}}$, $\mathsf{w_H}(x, y)$ is odd, and the $\mathsf{w_H}(y)$-th coordinate of y is 0 (*i.e.*, $y_{\mathsf{w_H}(y)} = 0$).

Lemma 5.4.1. *Let $n = n_0 2^m$ where $n_0 > 1$ is an odd positive integer and $m \geq 0$ is an integer. Let $F_{n_0} \in \mathcal{B}_{n_0}$ be a WAPB Boolean function. Let $F'_n \in \mathcal{B}_n$ be recursively defined as*

$$\mathsf{supp}(F'_n) = \begin{cases} \mathsf{supp}(F_{n_0}) & \text{if } n = n_0, \\ W_e \cup W_o^0 \cup W_o^1 & \text{if } n > n_0, \end{cases}$$

where

$$\begin{aligned} W_e &= \{(x, y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x, y) \textit{ is even}\} \\ &\qquad\qquad\qquad\qquad \triangle\{(z, z) \in \mathbb{F}_2^n : z \in \mathsf{supp}(F'_{\frac{n}{2}})\}, \\ W_o^0 &= \{(y, x) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x, y) \textit{ is odd and } y_{\mathsf{w_H}(y)} = 0\}, \\ W_o^1 &= \{(x, y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x, y) \textit{ is odd and } y_{\mathsf{w_H}(y)} = 1\}. \end{aligned}$$

Here, $y_{\mathsf{w_H}(y)}$ is the $\mathsf{w_H}(y)$-th coordinate bit in $y = (y_1, y_2, \ldots, y_{\frac{n}{2}})$. Then F'_n is a WAPB Boolean function.

Proof. The proof is similar to the proof of Lemma 5.3.2. If $n = n_0$, then F'_n is already a WAPB Boolean function. Consider $n = n_0 2^m$ where $m > 1$. Let $k \in [1, n]$ can be written as $k = k_0 2^b$ where k_0 is odd and $b \geq 0$ is an integer. We will prove this in three different cases by taking $b = 0$; $1 \leq b < m$ and $b \geq m$. Here, $k \not\leq n$ for first two cases *i.e.*, when $0 \leq b < m$.

Case I: If $b = 0$, then $k = k_0$, which is odd. Then $\mathsf{supp}_k(F'_n) = A \cup B$ where
$A = \{w \in W_o^0 : \mathsf{w_H}(w) = k\} = \{(y, x) : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x, y) = k, y_{\mathsf{w_H}(y)} = 0\}$ and

$B = \{w \in W_o^1 : \mathsf{w_H}(w) = k\} = \{(x,y) : x,y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = k, y_{\mathsf{w_H}(y)} = 1\}$.

Since $\mathsf{w_H}(x)$ is odd and $\mathsf{w_H}(y)$ is even in the vectors in both the set A and B, $A \cap B = \emptyset$. Now using the equality in Lemma 5.1.1[Item 1], we have

$$\begin{aligned}
\mathsf{w}_{k,n}(F'_n) &= |A| + |B| = \sum_{\substack{i=0 \\ i \text{ is odd}}}^{k} \binom{\frac{n}{2}}{i}\binom{\frac{n}{2}-1}{k-i} + \sum_{\substack{i=0 \\ i \text{ is odd}}}^{k} \binom{\frac{n}{2}}{i}\binom{\frac{n}{2}-1}{(k-1)-i} \\
&= \sum_{\substack{i=0 \\ i \text{ is odd}}}^{k} \binom{\frac{n}{2}}{i}\left(\binom{\frac{n}{2}-1}{k-i} + \binom{\frac{n}{2}-1}{(k-1)-i}\right) \\
&= \sum_{\substack{i=0 \\ i \text{ is odd}}}^{k} \binom{\frac{n}{2}}{i}\binom{\frac{n}{2}}{k-i} = \frac{1}{2}\binom{n}{k}.
\end{aligned}$$

Case II: If $1 \leq b < m$, then $k = k_0 2^b$ is even. As $\mathsf{supp}(F'_n)$ is defined recursively, we have

$$\begin{aligned}
\mathsf{supp}_k(F'_n) = &\{(x,y) : x,y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = k\} \\
&\triangle\{(x,y,x,y) : x,y \in \mathbb{F}_2^{\frac{n}{2^2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = \frac{k}{2}\} \\
&\vdots \\
&\triangle\{(x,y,x,y,\ldots,x,y) : x,y \in \mathbb{F}_2^{\frac{n}{2^b}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = \frac{k}{2^{b-1}}\} \\
&\triangle\{(z,z,\ldots,z,z) : z \in \mathsf{supp}_{\frac{k}{2^b}}(F'_{\frac{n}{2^b}})\} \\
&=\{(x,y) : x,y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = k\} && (S_1) \\
&\triangle\{(x,y,x,y) : x,y \in \mathbb{F}_2^{\frac{n}{2^2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = \frac{k}{2}\} && (S_2) \\
&\vdots \\
&\triangle\{(x,y,\ldots,x,y) : x,y \in \mathbb{F}_2^{\frac{n}{2^b}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = \frac{k}{2^{b-1}}\} && (S_b) \\
&\triangle\{(y,x,\ldots,y,x) : x,y \in \mathbb{F}_2^{\frac{n}{2^{b+1}}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = \frac{k}{2^b}, y_{\mathsf{w_H}(y)} = 0\} && (S^0_{b+1}) \\
&\triangle\{(x,y,\ldots,x,y) : x,y \in \mathbb{F}_2^{\frac{n}{2^{b+1}}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = \frac{k}{2^b}, y_{\mathsf{w_H}(y)} = 1\}. && (S^1_{b+1})
\end{aligned}$$

By following Lemma 5.3.2, we can see $S_i \cap S_j = \emptyset$ for $1 \leq i < j \leq b$. Further, since the $\mathsf{w_H}(x,y) = \frac{k}{2^b} = k_0$ is odd in the sets S^0_{b+1} and S^1_{b+1}, the set $S^0_{b+1} \cup S^1_{b+1} \subset S_b$ (where

$S_{b+1}^0 \cap S_{b+1}^1 = \emptyset$). Therefore,

$$\mathsf{w}_{k,n}(\bar{f}_n) = |S_1| + |S_2| + \cdots + |S_b| - |S_{b+1}^0 \cup S_{b+1}^1|.$$

The cardinality of S_j for $j \in [1, b]$ has been computed in Lemma 5.3.2. Now,

$$\begin{aligned} |S_{b+1}^0 \cup S_{b+1}^1| &= \sum_{\substack{i=0 \\ i \text{ is odd}}}^{\frac{k}{2^b}} \binom{\frac{n}{2^{b+1}}}{i} \binom{\frac{n}{2^{b+1}} - 1}{\frac{k}{2^b} - 1 - i} + \sum_{\substack{i=0 \\ i \text{ is odd}}}^{\frac{k}{2^b}} \binom{\frac{n}{2^{b+1}}}{i} \binom{\frac{n}{2^{b+1}} - 1}{\frac{k}{2^b} - i} \\ &= \sum_{\substack{i=0 \\ i \text{ is odd}}}^{\frac{k}{2^b}} \binom{\frac{n}{2^{b+1}}}{i} \binom{\frac{n}{2^{b+1}}}{\frac{k}{2^b} - i} = \frac{1}{2} \binom{\frac{n}{2^b}}{\frac{k}{2^b}}. \end{aligned}$$

Therefore,

$$\mathsf{w}_{k,n}(F_n') = \frac{1}{2} \left[\sum_{j=1}^{b-1} \left(\binom{\frac{n}{2^{j-1}}}{\frac{k}{2^{j-1}}} - \binom{\frac{n}{2^j}}{\frac{k}{2^j}} \right) + \binom{\frac{n}{2^{b-1}}}{\frac{k}{2^{b-1}}} + \binom{\frac{n}{2^b}}{\frac{k}{2^b}} - \binom{\frac{n}{2^b}}{\frac{k}{2^b}} \right] = \frac{1}{2} \binom{n}{k}.$$

Case III: If $b \geq m$, then as Lemma 5.3.2, we have

$$\begin{aligned} \mathsf{supp}_k(F_n') = \{(x, y) : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x, y) = k\} && (T_1) \\ \triangle\{(x, y, x, y) : x, y \in \mathbb{F}_2^{\frac{n}{2^2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x, y) = \frac{k}{2}\} && (T_2) \\ \vdots \\ \triangle\{(x, y, \ldots, x, y) : x, y \in \mathbb{F}_2^{\frac{n}{2^m}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x, y) = \frac{k}{2^{m-1}}\} && (T_m) \\ \triangle\{(z, z, \ldots, z, z) : z \in \mathsf{supp}_{\frac{k}{2^m}}(F_{n_0}')\}. && (T_{m+1}) \end{aligned}$$

By following Lemma 5.3.2, we can see $T_i \cap T_j = \emptyset$ for $1 \leq i < j \leq m$. For $b > m$, using the value of $|T_j|, j \in [1, m+1]$ as computed in Lemma 5.3.2, we have

$$\mathsf{w}_{k,n}(F_n') = \frac{1}{2} \left(\binom{n}{k} + a_{\frac{k}{2^m}}^{n_0} \right).$$

For $m = b$, $\frac{k}{2^m} = k_0$ is odd. Similarly, as in the proof of Lemma 5.3.2, we have

$$\mathsf{w}_{k,n}(F_n') = \frac{1}{2} \left(\binom{n}{k} - a_{\frac{k}{2^m}}^{n_0} \right).$$

Therefore, for $n = n_0 2^m$ and $k = k_0 2^b$, we got that $\mathsf{w}_{k,n}(F'_n) = \frac{1}{2}\left(\binom{n}{k} + a^n_k\right)$, where

$$a^n_k = \begin{cases} 0 & \textit{if } b < m, \\ a^{n_0}_{\frac{k}{2^m}} & \textit{if } b > m, \\ -a^{n_0}_{\frac{k}{2^m}} & \textit{if } m = b. \end{cases}$$

Therefore, it is established that F'_n is a WAPB Boolean function, provided that F'_{n_0} is also WAPB Boolean function. □

The modified function F'_n is no longer equivalent to a linear function in $\mathsf{E}_{k,n}$ for n even and k odd. Experimentally, the results show the nonlinearity $\mathsf{NL}_k(F'_n)$ improves significantly compared to Corollary 5.3.10 for n even and k odd. Nevertheless, we found that there is no improvement when k is even.

Therefore, our next attempt is to improve NL_k for n even and k even by replacing some vectors in W_e, as defined in Lemma 5.4.1. Since, $k = \mathsf{w_H}(x, y)$ is even and $\mathsf{w_H}(x)$ is odd (implying $\mathsf{w_H}(y)$ is also odd), replacing some (x, y) with (y, x) may result in repetition of vectors and count of support vectors gets reduced. Consequently, the approach used in Lemma 5.4.1, is not applicable in this case.

Instead of swapping x and y in some (x, y), we propose swapping two bits $x_{\frac{k}{2}}$ and $y_{\frac{k}{2}}$ in those vectors where $x_{\frac{k}{2}} = 1$ and $y_{\frac{k}{2}} = 0$, with $k = \mathsf{w_H}(x, y)$. This bit transformation avoids redundancy and allows for improvement in the weightwise nonlinearity for even k.

Lemma 5.4.2. *Let $n = n_0 2^m$ where $n_0 > 1$ is an odd positive integer and $m \geq 0$ is an integer. Let $F_{n_0} \in \mathcal{B}_{n_0}$ be a WAPB Boolean function. Let $F_n \in \mathcal{B}_n$ be recursively defined as*

$$\mathsf{supp}(F_n) = \begin{cases} \mathsf{supp}(F_{n_0}) & \textit{if } n = n_0, \\ ((W_e \setminus W_e^{10}) \cup W_e^{01}) \cup (W_o^0 \cup W_o^1) & \\ \qquad = (\mathsf{supp}(F'_n) \setminus W_e^{01}) \cup W_e^{10} & \textit{if } n > n_0, \end{cases}$$

where W_e, W_o^0, W_o^1 and F_n' are as defined in Lemma 5.4.1 and

$$\begin{aligned} W_e^{10} &= \{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) \textit{ is even and} \\ &\qquad\qquad\qquad\qquad x_{\frac{\mathsf{w_H}(x,y)}{2}} = 1, y_{\frac{\mathsf{w_H}(x,y)}{2}} = 0\}, \\ W_e^{01} &= \{(\overline{x}, \overline{y}) \in \mathbb{F}_2^n : (x,y) \in W_e^{10}\}. \end{aligned}$$

Here, $x_{\frac{\mathsf{w_H}(x,y)}{2}}$ and $y_{\frac{\mathsf{w_H}(x,y)}{2}}$ are the $l = \frac{\mathsf{w_H}(x,y)}{2}$-th cordinate bit in x, y respectively and $(\overline{x}, \overline{y}) = (x_1, \ldots, x_{l-1}, y_l, x_{l+1}, \ldots, x_{\frac{n}{2}}, y_1, \ldots, y_{l-1}, x_l, y_{l+1}, \ldots, y_{\frac{n}{2}})$ obtained by swapping l-th cordinate bits of x and y.

Then F_n is a WAPB Boolean function.

Proof. We have already proved in Lemma 5.4.1 that F_n' is WAPB.

Here, $\mathsf{w_H}(\overline{x})$ and $\mathsf{w_H}(\overline{y})$ are even and $\overline{x} \neq \overline{y}$ for the vectors $(\overline{x}, \overline{y}) \in W_e^{01}$. Then, W_e^{01} and $\mathsf{supp}(F_n')$ are disjoint sets as

- $\mathsf{w_H}(x,y)$ is odd for the vectors $(x,y) \in W_o^0 \cup W_o^1$;

- $\mathsf{w_H}(x)$ and $\mathsf{w_H}(y)$ are odd or, $x = y$ for the vectors $(x,y) \in W_e$.

Further, $W_e^{10} \subseteq \mathsf{supp}(F_n')$. As we replace (x,y) by a same weight vector $(\overline{x}, \overline{y})$, $\mathsf{w}_{k,n}(F_n) = \mathsf{w}_{k,n}(F_n')$. Hence, F_n is a WAPB Boolean function with $\mathsf{w}_{k,n}(F_n) = \frac{1}{2}\left(\binom{n}{k} + a_k^n\right)$, where

$$a_k^n = \begin{cases} 0 & \textit{if } b < m, \\ a^{n_0}_{\frac{k}{2^m}} & \textit{if } b > m, \\ -a^{n_0}_{\frac{k}{2^m}} & \textit{if } m = b. \end{cases}$$

□

Hence, in the following Theorem, we present a recursive construction for a WAPB Boolean function $F_n \in \mathcal{B}_n$ from a WAPB Boolean function $F \in \mathcal{B}_p$ in fewer variables.

Theorem 5.4.3. *For $p \geq 2$, let $F \in \mathcal{B}_p$ be a WAPB Boolean function. Let n be a positive integer such that, for an $m \geq 0$,*

$$p = \lfloor \frac{n}{2^m} \rfloor \text{ i.e, } n = a_0 2^0 + a_1 2^1 + \cdots + a_{m-1} 2^{m-1} + p 2^m \tag{5.8}$$

$$\text{or,} \qquad p+1 = \lfloor \frac{n}{2^m} \rfloor \text{ i.e, } n = a_0 2^0 + a_1 2^1 + \cdots + a_{m-1} 2^{m-1} + (p+1) 2^m \text{ if } p \text{ is even.} \tag{5.9}$$

Let $F_n \in \mathcal{B}_n$ whose support is defined as

$$\mathsf{supp}(F_n) = \begin{cases} \mathsf{supp}(F) & \text{if } n = p, \\ \{(x,0) \in \mathbb{F}_2^n : x \in \mathsf{supp}(F_{n-1})\} & \\ \qquad \cup \{(x,1) \in \mathbb{F}_2^n : x \notin \mathsf{supp}(F_{n-1})\} & \text{if } n > p \text{ and } n \text{ is odd}, \\ (W_e \setminus W_e^{10}) \cup W_e^{01} \cup W_o^0 \cup W_o^1 & \text{if } n > p \text{ and } n \text{ is even}, \end{cases} \tag{5.10}$$

where

$$\begin{aligned} W_e &= \{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \text{ is odd}, \mathsf{w_H}(x,y) \text{ is even}\} \\ &\qquad\qquad \triangle \{(z,z) \in \mathbb{F}_2^n : z \in \mathsf{supp}(F_{\frac{n}{2}})\}, \\ W_e^{10} &= \{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \text{ is odd}, \mathsf{w_H}(x,y) \text{ is even and} \\ &\qquad\qquad x_{\frac{\mathsf{w_H}(x,y)}{2}} = 1, y_{\frac{\mathsf{w_H}(x,y)}{2}} = 0\}, \\ W_e^{01} &= \{(\overline{x}, \overline{y}) \in \mathbb{F}_2^n : (x,y) \in W_e^{10}\}, \\ W_o^0 &= \{(y,x) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \text{ is odd}, \mathsf{w_H}(x,y) \text{ is odd}, y_{\mathsf{w_H}(y)} = 0\}, \\ W_o^1 &= \{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \text{ is odd}, \mathsf{w_H}(x,y) \text{ is odd}, y_{\mathsf{w_H}(y)} = 1\}. \end{aligned}$$

Then F_n is a WAPB Boolean function.

Note 5.4.4. In the set W_o^0 and W_o^1, it is possible to choose any fixed index in y (rather than $y_{\mathsf{w_H}(y)}$), which is fixed for the vectors of the same weight. For instance, one can choose y_1 instead of $y_{\mathsf{w_H}(y)}$. However, we choose $y_{\mathsf{w_H}(y)}$ because it varies with the weight for an expectation of better nonlinearity. Similarly, in the sets W_e^{10} and W_e^{01} one can replace the bit at index $\frac{\mathsf{w_H}(x+y)}{2}$ with any other fixed index, as this position is also constant among vectors with the same total weight.

Here, the function F_n is built up by modifying $\mathsf{supp}(f_n)$ for n even. As a result, F_n is no longer a linear function in $\mathsf{E}_{k,n}$ for n even, k odd. Therefore, the cryptographic properties of F_n improve compared to f_n, as we will see now.

5.4.1 Algebraic Normal Form of F_n

Theorem 5.4.5. *For $p \geq 2$, let F_p be a WAPB Boolean function. Let n be a positive integer such that, for an $m \geq 0$,*

- $p = \lfloor \frac{n}{2^m} \rfloor$ *i.e, $n = a_0 2^0 + a_1 2^1 + \cdots + a_{m-1} 2^{m-1} + p 2^m$, or,*
- $p + 1 = \lfloor \frac{n}{2^m} \rfloor$ *i.e, $n = a_0 2^0 + a_1 2^1 + \cdots + a_{m-1} 2^{m-1} + (p+1) 2^m$ if p is even.*

Then the ANF of F_n, defined in Theorem 5.4.3 is

$$F_n(x_1, \ldots, x_n) = \begin{cases} F_p & \text{if } n = p, \\ x_n + F_{n-1}(x_1, \ldots, x_{n-1}) & \text{if } n > p \text{ and odd}, \\ g_{n_0}(x_1, \ldots, x_n) + g_{n_1}(x_1, \ldots, x_n) & \\ +g_{n_2}(x_1, \ldots, x_n) + z_n(x_1, \ldots, x_n) & \\ +h_{n_0}(x_1, \ldots, x_n) + h_{n_1}(x_1, \ldots, x_n) & \text{if } n > p \text{ and even} \end{cases} \tag{5.11}$$

where the ANF of $g_{n_0}, g_{n_1}, g_{n_2}, z_n, h_{n_0}, h_{n_1}$ are presented in the Equations 5.12 - 5.17.

Proof. The ANF of F_n for $n > p$ and n odd, can easily be derived. Now consider the case n is even. We can rewrite the support of F_n as follows.

$$\begin{aligned} \mathsf{supp}(F_n) = \{&(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = k \textit{ is even}, x_{\frac{k}{2}} = y_{\frac{k}{2}}\} \\ &\cup\{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = k \textit{ is even}, x_{\frac{k}{2}} = 1, y_{\frac{k}{2}} = 0\} \\ &\cup\{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is even}, \mathsf{w_H}(x,y) = k \textit{ is even}, x_{\frac{k}{2}} = 1, y_{\frac{k}{2}} = 0\} \\ &\triangle\{(z,z) \in \mathbb{F}_2^n : z \in \mathsf{supp}(F_{\frac{n}{2}}) \\ &\cup\{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is even}, \mathsf{w_H}(x,y) \textit{ is odd and } x_{\mathsf{w_H}(x)} = 0\} \\ &\cup\{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) \textit{ is odd and } x_{\mathsf{w_H}(y)} = 1\}. \end{aligned}$$

Let $g_{n_0}, g_{n_1}, g_{n_2}, h_{n_1}, h_{n_0}, z_n \in \mathcal{B}_n$ such that

$$\begin{aligned}
\mathsf{supp}(g_{n_0}) &= \{(x,y) : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = k \textit{ even}, x_{\frac{k}{2}} = y_{\frac{k}{2}}\};\\
\mathsf{supp}(g_{n_1}) &= \{(x,y) : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) = k \textit{ even}, x_{\frac{k}{2}} = 1, y_{\frac{k}{2}} = 0\};\\
\mathsf{supp}(g_{n_2}) &= \{(x,y) : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is even}, \mathsf{w_H}(x,y) = k \textit{ even}, x_{\frac{k}{2}} = 1, y_{\frac{k}{2}} = 0\};\\
\mathsf{supp}(z_n) &= \{(z,z) : z \in \mathsf{supp}(F_{\frac{n}{2}})\};\\
\mathsf{supp}(h_{n_0}) &= \{(x,y) : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is even}, \mathsf{w_H}(x,y) \textit{ odd}, x_{\mathsf{w_H}(x)} = 0\} \text{ and}\\
\mathsf{supp}(h_{n_1}) &= \{(x,y) : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x,y) \textit{ odd}, x_{\mathsf{w_H}(y)} = 1\}.
\end{aligned}$$

Let $\phi_k^n(x_1, x_2, \ldots, x_n)$ be the symmetric function with support $\mathsf{supp}(\phi_k^n) = \mathsf{E}_{k,n}$. Then the ANFs of $g_{n_0}, g_{n_1}, g_{n_2}, h_{n_1}, h_{n_0}, z_n \in \mathcal{B}_n$ are as follows;

$$g_{n_0}(x_1, \ldots, x_n) = \sum_{i=1}^{\frac{n}{2}} x_i \sum_{i=\frac{n}{2}+1}^{n} x_i \left(\sum_{k=1,\textit{ even}}^{n} \left(1 + x_{\frac{k}{2}} + x_{\frac{n}{2}+\frac{k}{2}}\right) \phi_k^n(x_1, \ldots, x_n) \right), \tag{5.12}$$

$$g_{n_1}(x_1, \ldots, x_n) = \sum_{i=1}^{\frac{n}{2}} x_i \sum_{i=\frac{n}{2}+1}^{n} x_i \left(\sum_{k=1,\textit{ even}}^{n} x_{\frac{k}{2}} \left(1 + x_{\frac{n}{2}+\frac{k}{2}}\right) \phi_k^n(x_1, \ldots, x_n) \right), \tag{5.13}$$

$$g_{n_2}(x_1, \ldots, x_n) = \left(1 + \sum_{i=1}^{\frac{n}{2}} x_i\right) \left(1 + \sum_{i=\frac{n}{2}+1}^{n} x_i\right) \left(\sum_{\substack{k=1\\ \textit{even}}}^{n} x_{\frac{k}{2}} \left(1 + x_{\frac{n}{2}+\frac{k}{2}}\right) \phi_k^n(x_1, \ldots, x_n) \right), \tag{5.14}$$

$$z_n(x_1, \ldots, x_n) = \prod_{i=1}^{\frac{n}{2}} \left(x_i + x_{\frac{n}{2}+i} + 1\right) F_{\frac{n}{2}}(x_1, \ldots, x_{\frac{n}{2}}), \tag{5.15}$$

$$h_{n_0}(x_1, \ldots, x_n) = \left(1 + \sum_{i=1}^{\frac{n}{2}} x_i\right) \left(\sum_{i=\frac{n}{2}+1}^{n} x_i\right) \left(\sum_{k=1,\textit{ even}}^{\frac{n}{2}} (1 + x_k) \phi_k^{\frac{n}{2}}(x_1, \ldots, x_{\frac{n}{2}}) \right), \tag{5.16}$$

$$h_{n_1}(x_1, \ldots, x_n) = \left(\sum_{i=1}^{\frac{n}{2}} x_i\right) \left(1 + \sum_{i=\frac{n}{2}+1}^{n} x_i\right) \left(\sum_{k=1,\textit{ even}}^{\frac{n}{2}} x_{\frac{n}{2}+k}\, \phi_k^{\frac{n}{2}}(x_{\frac{n}{2}+1}, \ldots, x_n) \right). \tag{5.17}$$

Since $\mathsf{supp}(g_{n_0}) \triangle \mathsf{supp}(z_n), \mathsf{supp}(g_{n_1}), \mathsf{supp}(g_{n_2}), \mathsf{supp}(h_{n_0})$ and $\mathsf{supp}(h_{n_1})$ are pairwise disjoint sets, their union is same as their symmetric difference.

Hence, $\mathsf{supp}(F_n) = \mathsf{supp}(g_{n_0}) \triangle \mathsf{supp}(g_{n_1}) \triangle \mathsf{supp}(g_{n_2}) \triangle \mathsf{supp}(h_{n_0}) \triangle \mathsf{supp}(h_{n_1}) \triangle \mathsf{supp}(z_n)$.

Therefore,

$$
\begin{aligned}
F_n(x_1, x_2, \ldots, x_n) = & g_{n_0}(x_1, x_2, \ldots, x_n) + g_{n_1}(x_1, x_2, \ldots, x_n) + g_{n_2}(x_1, x_2, \ldots, x_n) \\
& + z_n(x_1, \ldots, x_n) + h_{n_0}(x_1, \ldots, x_n) + h_{n_1}(x_1, \ldots, x_n).
\end{aligned}
$$

□

5.4.2 Nonlinearity of F_n

Since the modified function F_n is no longer equivalent to a linear function in $\mathsf{E}_{k,n}$ for n even and k odd, the weightwise nonlinearity $\mathsf{NL}_k(F_n)$ improves significantly in contrast to Corollary 5.3.10. Table 5.2 presents the nonlinearity $\mathsf{NL}(F_n)$, weightwise nonlinearities $\mathsf{NL}_k(F_n)$, and the global weightwise nonlinearity $\mathtt{GWNL}$ of the functions F_n for $8 \leq n \leq 16$, which are generated using Theorem 5.4.3 with the base function $F_2(x_1, x_2) = x_2$. As $\mathsf{NL}_0(F_n) = \mathsf{NL}_1(F_n) = \mathsf{NL}_{n-1}(F_n) = \mathsf{NL}_n(F_n) = 0$, these values are omitted from the table. Comparing the nonlinearity results of f_n (in Table 5.1) and F_n (in Table 5.2) shows that both the nonlinearity and weightwise nonlinearities improve substantially.

n	NL	NL_2	NL_3	NL_4	NL_5	NL_6	NL_7	NL_8	NL_9	NL_{10}	NL_{11}	NL_{12}	NL_{13}	NL_{14}	$\mathtt{GWNL}$
8	82	7	13	14	14	7	0	0	0	-	-	-	-	-	55
9	164	8	26	27	33	26	8	0	0	-	-	-	-	-	128
10	356	10	28	47	66	49	32	10	0	0	-	-	-	-	242
11	712	11	44	79	113	171	93	43	11	0	0	-	-	-	565
12	1504	12	45	127	234	210	244	127	50	12	0	0	-	-	1061
13	3008	13	63	177	361	582	594	371	178	79	13	0	0	-	2431
14	6112	15	66	230	635	708	1016	709	575	230	72	15	0	-	4271
15	12224	16	89	302	925	1510	2424	1725	1295	960	302	88	16	0	9652
16	24338	17	91	378	1321	2001	3485	3002	3499	1995	1363	378	126	18	17464

Table 5.2: Listing of $\mathsf{NL}(F_n), \mathsf{NL}_k(F_n), \mathtt{GWNL}(F_n)$ for $8 \leq n \leq 16$.

An upper bound on weightwise nonlinearities of WAPB Boolean functions was established in [20], and it is $\mathsf{NL}_k(f) \leq \frac{1}{2}[|\mathsf{E}_{k,n}| - \sqrt{|\mathsf{E}_{k,n}|}]$ (see Lemma 3.2.5). However, this upper bound is not tight.

For instance, the NL_1 of any Boolean function is 0 from the Theorem 3.2.4 but the upper bound of NL_1 by [20] is $\frac{n-\sqrt{n}}{2}$. More experimental examples can be found in [43, Table

3] for $4 \leq n \leq 8$ and $2 \leq k \leq n-2$. In Table 5.3, we provide a comparison between the weightwise nonlinearities of the modified functions F_n and the theoretical upper bound. Further, no tight upper bound is available for the nonlinearity of WAPB Boolean functions. Therefore, we have presented a comparison of the nonlinearity of F_n with the upper bound of the nonlinearity of n-variable Boolean functions [55].

n	*function*	NL	NL_2	NL_3	NL_4	NL_5	NL_6	NL_7	NL_8	NL_9	NL_{10}	NL_{11}	$\mathtt{GWNL}$
9	*UB*	244	15	37	57	57	37	15	-	-	-	-	218
	F_9	164	8	26	27	33	26	8	-	-	-	-	128
10	*UB*	496	19	54	97	118	97	54	19	-	-	-	498
	F_{10}	356	10	28	47	66	49	32	10	-	-	-	128
11	*UB*	1000	23	76	155	220	220	155	76	23	-	-	948
	F_{11}	712	11	44	79	113	171	93	43	11	-	-	565
12	*UB*	2016	28	102	236	381	446	381	236	102	28	-	1940
	F_{12}	1056	12	45	127	234	210	244	127	50	12	-	1061
13	*UB*	4050	34	134	344	625	837	837	625	344	134	34	3948
	F_{13}	3008	13	63	177	361	582	594	371	178	79	13	2431

Table 5.3: Comparison of $\mathsf{NL}_k(F_n)$ with the upper bound(UB) presented in [20]

We present a comparison of the nonlinearities of our proposed functions with those from recent constructions in the following tables. The results clearly demonstrate that the class of WAPB Boolean functions F_n achieves better weightwise nonlinearities compared to existing constructions, particularly in the case where n is not a power of 2.

WPB/ WAPB Boolean functions	NL_2	NL_3	NL_4
Upper Bound [20]	5	7	5
[20]	2	4	2
[98]	2	2	2
[106, f_n Equation(8)]	1	4	1
[48]	1	4	1
F_n[Theorem 5.4.3]	3	4	3

Table 5.4: Comparison of NL_k of 6-variable WAPB constructions

WPB/WAPB Boolean functions	NL_2	NL_3	NL_4	NL_5
Upper Bound [20]	8	14	14	8
[106, f_n Equation(8)]	1	5	5	1
[48]	1	5	5	1
F_n[Theorem 5.4.3]	5	11	7	5

Table 5.5: Comparison of NL_k of 7-variable WAPB constructions

WPB/ WAPB functions	NL_2	NL_3	NL_4	NL_5	NL_6
Upper Bound [20]	11	24	30	24	11
[20]	2	12	19	12	6
[66]	6,9	0,8,14,16, 18,20, 21,22	19,22,23,24, 25,26,27	19,20,21,22	6,9
[65, g_{2^q+2} Equation(9)]	2	12	19	12	2
[77, f_m Equation(13)]	2	0	3	0	2
[77, g_m Equation(22)]	2	14	19	14	2
[78, f_m Equation(2)]	2	8	8	8	2
[78, f_m Equation(3)]	6	8	26	8	6
[44, Table 1]	5,3,2,2	10,7,12,12	16,15,18,19	12,11,12,12	5,3,2,6
[44, Table 3]	5	16	20	17	5
[104, g_m Equation(11)]	2	12	19	12	6
[45]	6,6,7	19,14,15	21,20,18	11,11,14	3,6,6
F_n[Theorem 5.4.3]	7	13	14	14	7

Table 5.6: Comparison of NL_k of 8-variable WPB constructions.

5.5 Perturbation of Vectors for Constructing Highly Nonlinear WAPB Boolean Functions

In this section, we will present a class of n variable WAPB Boolean functions by modifying $\mathsf{supp}(f_n)$ presented in Theorem 5.3.3 using the support of a highly nonlinear function $\phi \in \mathcal{B}_n$. The following result presents the nonlinearity of the function derived from Proposition 2.2.11 and Lemma 5.3.9.

1. Let $f_n \in \mathcal{B}_n$ ($n > 2$), defined as in Theorem 5.3.3. Then $\mathsf{NL}(f_n) = 2\mathsf{NL}(f_{n-1})$ if n is odd and $\mathsf{NL}(f_n) = \mathsf{w_H}(f_{\frac{n}{2}})$ if n is even.

2. For n even, the nonlinearity of f_n defined in Theorem 5.3.3 is very low as $X_1 =$

$\{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}\}$ is the support of a linear function $\sum_{i=1}^{\frac{n}{2}} x_i$ and the cardinality of $X_2 = \{(z,z) \in \mathbb{F}_2^n : z \in \mathsf{supp}(f_{\frac{n}{2}})\}$ is $\mathsf{w_H}(f_{\frac{n}{2}})$. Furthermore, for n even and k odd, $\mathsf{supp}_k(f_n) = \mathsf{supp}(f_n) \cap \mathsf{E}_{k,n} = \{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}\} \cap \mathsf{E}_{k,n} = \mathsf{supp}_k(\sum_{i=1}^{\frac{n}{2}} x_i)$ and hence $\mathsf{NL}_k(f_n) = 0$.

As we observed that the nonlinearity of f_n tends to be weak when n is even, primarily because its support $\mathsf{supp}(f_n)$ is a linear function. In Section 5.4, we introduced a method that involves the swapping of vector components by fixing a fixed position, which led to a significant improvement in both the nonlinearity and weightwise nonlinearity of the function f_n defined in Theorem 5.3.3. Building on this idea, we now propose an alternative modification technique that is taking advantage of the support of a highly nonlinear function. Specifically, we aim to enhance the nonlinearity by applying coordinate permutations to selected support vectors in $\mathsf{supp}(f_n)$ when n is even.

Therefore, it is assumed that $n > 2$ and is *even* in this section. Hence, when n is even, as in Theorem 5.3.3, $\mathsf{supp}(f_n) = \{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}\} \triangle \{(z,z) \in \mathbb{F}_2^n : z \in \mathsf{supp}(f_{\frac{n}{2}})\}$. Then

$$\mathsf{supp}_k(f_n) = \begin{cases} \{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x) + \mathsf{w_H}(y) = k\} & \\ \qquad \triangle \{(z,z) \in \mathbb{F}_2^n : z \in \mathsf{supp}_{\frac{k}{2}}(f_{\frac{n}{2}})\} & \textit{if } k \textit{ even} \\ \{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x) + \mathsf{w_H}(y) = k\} & \textit{if } k \textit{ odd} \end{cases}$$

Now we will consider both cases of k (i.e., odd and even) and will propose to permute the coordinates of some vectors in $\mathsf{supp}_k(f_n)$.

5.5.1 When k is odd

In this case, $\mathsf{supp}_k(f_n) = \{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x) + \mathsf{w_H}(y) = k\} = \mathsf{supp}_k(l(x,y))$ where $l(x,y) = \sum_{i=1}^{\frac{n}{2}} x_i$, as we discussed at the end of Section 5.3. The linear function $l(x,y) = \sum_{i=1}^{\frac{n}{2}} x_i$ depends solely on the coordinates of x and is independent of y. To break the independence and enhance the nonlinearity, we introduce a modification using the support of a nonlinear function $\phi \in \mathcal{B}_{\frac{n}{2}}$. Specifically, for each $x \in \mathbb{F}_2^{\frac{n}{2}}$ such that $\mathsf{w_H}(x)$ is odd (i.e., satisfy $l(x, y)$), we keep (x, y) if $y \in \mathsf{supp}(\phi)$, and otherwise we replace (x, y) with (y, x). By choosing ϕ to be a highly nonlinear function, we ensure that the component y is expected to be far from the linear functions, and as a result, we have a high nonlinearity in f_n.

Here, if $\mathsf{w_H}(x, y) = k$ then $\mathsf{w_H}(y, x) = k$. Further, if $(x, y) \in \mathsf{supp}_k(f_n)$ then $\mathsf{w_H}(y)$ is even as $\mathsf{w_H}(x)$ is odd. So, $(y, x) \notin \mathsf{supp}_k(f_n)$ if $(x, y) \in \mathsf{supp}_k(f_n)$. Therefore, the replacement of $(x, y) \in \mathsf{supp}_k(f_n)$ by (y, x) does not change the weight of the resultant function in the domain $\mathsf{E}_{k,n}$.

Lemma 5.5.1. *Let $\phi \in \mathcal{B}_{\frac{n}{2}}$. A function $f \in \mathcal{B}_n$ such that for every $k \in [0, n]$ and odd,* $\mathsf{supp}_k(f^{\phi}) =$

$$\{(x, y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(y) = k - \mathsf{w_H}(x), y \in \mathsf{supp}(\phi)\}$$
$$\cup\{(y, x) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(y) = k - \mathsf{w_H}(x), y \notin \mathsf{supp}(\phi)\}. \quad (5.18)$$

Then $\mathsf{w}_{k,n}(f^{\phi}) = \frac{1}{2}\binom{n}{k}$.

Proof. Let $A = \{(x, y) \in \mathbb{F}_2^n | x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x) + \mathsf{w_H}(y) = k\}$. Hence,

$$|A| = \sum_{\substack{i=1 \\ i \text{ is odd}}}^{k} \binom{\frac{n}{2}}{i}\binom{\frac{n}{2}}{k-i} = \frac{1}{2}\binom{n}{k}$$

For any $\phi \in \mathcal{B}_{\frac{n}{2}}$, we have $A = A_1 \cup A_2$ where,

$$A_1 = \{(x, y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x) + \mathsf{w_H}(y) = k, y \in \mathsf{supp}(\phi)\} \text{ and}$$
$$A_2 = \{(x, y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x) + \mathsf{w_H}(y) = k, y \notin \mathsf{supp}(\phi)\}.$$

So, $A_1 \cap A_2 = \emptyset$. Further denote,

$$A_2^s = \{(y, x) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(y) = k - \mathsf{w_H}(x), y \notin \mathsf{supp}(\phi)\}.$$

Here, $\mathsf{w_H}(y)$ is even in A as $\mathsf{w_H}(x)$ and k are odd. So, $|A_2^s| = |A_2|$ and $A \cap A_2^s = \emptyset$. As $\mathsf{sup}_k(f^\phi) = (A \backslash A_2) \cup A_2^s$, $\mathsf{w}_{k,n}(f^\phi) = |(A \backslash A_2) \cup A_2^s| = |A| - |A_2| + |A_2^s| = |A| = \frac{1}{2}\binom{n}{k}$. □

5.5.2 When k is even

In this case, $\mathsf{supp}_k(f_n) = \{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x) + \mathsf{w_H}(y) = k\} \triangle \{(z,z) \in \mathbb{F}_2^n : z \in \mathsf{supp}_{\frac{k}{2}}(f_{\frac{n}{2}})\}$. Let denote the set $L = \{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x) + \mathsf{w_H}(y) = k\}$ and $M = \{(z,z) \in \mathbb{F}_2^n : z \in \mathsf{supp}_{\frac{k}{2}}(f_{\frac{n}{2}})\}$. In this case, the replacement of $(x,y) \in \mathsf{supp}_k(f_n)$ by (y,x) is not straight forward as in Subsection 5.5.1. If $(x,y) \in L$ then $\mathsf{w_H}(y)$ is odd as $\mathsf{w_H}(x)$ is odd. As a result, (y,x) could be present in L. Therefore, replacement of $(x,y) \in \mathsf{supp}_k(f_n)$ by (y,x) can possibly duplicate an existing vector in L, which reduces the weight of the resultant function. Therefore, we attempt to swap two bits of x and y instead of swapping x and y as in the following lemma. For given $(x,y) \in \mathbb{F}_2^n$ where $x = (x_1, \ldots, x_{\frac{n}{2}}), y = (y_1, \ldots, y_{\frac{n}{2}}) \in \mathbb{F}_2^{\frac{n}{2}}$, we denote $(x^i, y^i) = (x_1, \ldots, x_{i-1}, y_i, x_{i+1}, \ldots, x_{\frac{n}{2}}, y_1, \ldots, y_{i-1}, x_i, y_{i+1}, \ldots, y_{\frac{n}{2}})$. That is, (x^i, y^i) is obtained by swapping the i-th bits of x and y.

Lemma 5.5.2. *Let $f_n \in \mathcal{B}_n$ be the function defined in Theorem 5.3.3. For every $k \in [0,n]$ and even, let $W_k = \{(x,y) \in \mathsf{supp}_k(f_n) : \ \mathsf{w_H}(x) \textit{ odd}, \exists i \in [1, \frac{n}{2}] \textit{ s.t. } x_j = y_j \textit{ for } j \in [1, i-1], \textit{ and } x_i = 0, y_i = 1\}$, and*
$W_k' = \{(x^i, y^i) : \ (x,y) \in W_k, i \in [1, \frac{n}{2}] \textit{ s.t. } x_j = y_j \textit{ for } j \in [1, i-1], \textit{ and } x_i = 0, y_i = 1\}$. A function $g_n \in \mathcal{B}_n$ defined as $\mathsf{supp}_k(g_n) = (\mathsf{supp}_k(f_n) \setminus W_k) \cup W_k'$ for every $k \in [0,n]$ and even. Then $\mathsf{w}_{k,n}(g_n) = \mathsf{w}_{k,n}(f_n)$, if k is even.

Proof. From Theorem 5.3.3, $\mathsf{supp}_k(f_n) = \{(x,y) \in \mathbb{F}_2^n : x, y \in \mathbb{F}_2^{\frac{n}{2}}, \mathsf{w_H}(x) \textit{ is odd}, \mathsf{w_H}(x) + \mathsf{w_H}(y) = k\} \triangle \{(z,z) \in \mathbb{F}_2^n : z \in \mathsf{supp}_{\frac{k}{2}}(f_{\frac{n}{2}})\}$ for $k \in [0,n]$ and even. Here, the weight of each vector in W_k' is k and $|W_k| = |W_k'|$. As k is even, $\mathsf{w_H}(x)$ and $\mathsf{w_H}(y)$ are odd for every $(x,y) \in W_k$. That implies, $\mathsf{w_H}(x^i)$ and $\mathsf{w_H}(y^i)$ are even for every $(x^i, y^i) \in W_k'$. Hence,

$W_k \cap W_k^{'} = \emptyset$. Further, $x^i \neq y^i$ for every $(x^i, y^i) \in W_k^{'}$ as i-th bit in x and y are different. Hence, $\mathsf{supp}_k(f_n) \cap W_k^{'} = \emptyset$. Hence, $\mathsf{w}_{k,n}(g_n) = \mathsf{w}_{k,n}(f_n) - |W_k| + |W_k^{'}| = \mathsf{w}_{k,n}(f_n)$. □

Similar to the approach in Lemma 5.5.1, we now use the support of another Boolean function (possibly, a highly nonlinear) to swap x^i and y^i in some of $(x^i, y^i) \in W_k'$ as defined in Lemma 5.5.2.

Lemma 5.5.3. *Let $\psi \in \mathcal{B}_{\frac{n}{2}}$. Let $g_n \in \mathcal{B}_n$ as defined in Lemma 5.5.2 with W_k and W_k'. A function $h_n^{\psi} \in \mathcal{B}_n$ such that for every $k \in [0, n]$ and even,*

$\mathsf{supp}_k(h_n^{\psi}) = \{(x, y) \in \mathsf{supp}_k(g_n) : (x, y) \notin W_k'\} \cup \{(x, y) : (x, y) \in W_k',$ *and* $y \in \mathsf{supp}(\psi)\} \cup \{(y, x) : (x, y) \in W_k'$ *and* $y \notin \mathsf{supp}(\psi)\}$.

Then $\mathsf{w}_{k,n}(h_n^{\psi}) = \mathsf{w}_{k,n}(g_n)$.

Proof. Denote $\mathsf{supp}_k(h_n^{\psi}) = H_k \cup W_k^{\psi} \cup W_k^{\overline{\psi}}$, where $H_k = \{(x, y) \in \mathsf{supp}_k(g_n) : (x, y) \notin W_k'\}, W_k^{v} = \{(x, y) : (x, y) \in W_k',$ and $y \in \mathsf{supp}(\psi)\}$ and $W_k^{\overline{\psi}} = \{(y, x) : (x, y) \in W_k'$ *and* $y \notin \mathsf{supp}(\psi)\}$.

From Lemma 5.5.2, we have $\mathsf{supp}_k(g_n) = (\mathsf{supp}_k(f_n) \setminus W_k) \cup W_k^{'}$. Here, $\mathsf{w_H}(x)$ and $\mathsf{w_H}(y)$ are even for every $(x, y) \in W_k^{'}$. Since, $x_i \neq y_i$ in $(x, y) \in W_k^{'}$, $(x, y) \neq (y, x)$ for each $(x, y) \in W_k^{'}$. For any $\psi \in \mathcal{B}_{\frac{n}{2}}$, $W_k^{'}$ can be partitioned as

$$W_k^{'} = \{(x, y) \in W_k^{'} : y \in \mathsf{supp}(\psi)\} \cup \{(x, y) \in W_k^{'} : y \notin \mathsf{supp}(\psi)\}.$$

Then $|\{(x, y) : (x, y) \in W_k^{'}$ and $y \notin \mathsf{supp}(\psi)\}| = |\{(y, x) : (x, y) \in W_k^{'}$ and $y \notin \mathsf{supp}(\psi)\}| = |W_k^{\overline{\psi}}|$. From the definition of $W_k^{'}$, for every $(x, y) \in W_k^{'}$ there is an $i \in [1, \frac{n}{2}]$ such that $x_j = y_j$ for $1 \leq j \leq i - 1$ and $y_i = 0, x_i = 1$. Hence, $W_k^{'} \cap \{(y, x) : (x, y) \in W_k^{'}$ and $y \notin \mathsf{supp}(\psi)\} = W_k^{'} \cap W_k^{\overline{\psi}} = \emptyset$.

Further, as $\mathsf{w_H}(x), \mathsf{w_H}(y)$ are odd for every $(x, y) \in H_k$ and $\mathsf{w_H}(x), \mathsf{w_H}(y)$ are even for every $(x, y) \in W_k^{'} \cup W_k^{\overline{\psi}}$, $H_k \cap W_k^{\overline{\psi}} = \emptyset$. Hence $\mathsf{w}_{k,n}(h_n^{\psi}) = |H_k| + |W_k^{\psi}| + |W_k^{\overline{v}}| = (\mathsf{w}_{k,n}(g_n) - |W_k^{'}|) + |W_k^{'}| = \mathsf{w}_{k,n}(g_n)$. □

5.5.3 A class of WAPB Boolean function

Now we will apply Lemma 5.5.1 and Lemma 5.5.3 to construct a WAPB Boolean function with improved nonlinearity.

Theorem 5.5.4. *Let $\phi, \phi \in \mathcal{B}_{\frac{n}{2}}$. Let $f_n \in \mathcal{B}_n$ be the function defined in Theorem 5.3.3. Let $F_n^{\phi\psi} \in \mathcal{B}_n$ with support $\mathsf{supp}_k(F_n^{\phi\psi}) = \begin{cases} \mathsf{supp}_k(h_n^{\psi}) & \text{if } k \text{ is even} \\ \mathsf{supp}_k(f_n^{\phi}) & \text{if } k \text{ is odd,} \end{cases}$ where f_n^{ϕ}, h_n^{ψ} are as defined in Lemma 5.5.1 and Lemma 5.5.3 respectively. Then $F_n^{\phi\psi}$ is a WAPB Boolean function.*

The following is a recursive construction of a WAPB Boolean function.

Construction 5.5.5. For $n \geq 2$, let $F_n^{\phi\psi} \in \mathcal{B}_n$ with support

$$\mathsf{supp}(F_n^{\phi\psi}) = \begin{cases} \{(x,1) \in \mathbb{F}_2^2 : x \in \mathbb{F}_2\} = \{(0,1),(1,1)\} & \textit{if } n = 2, \\ \{(x,0) \in \mathbb{F}_2^n : x \in \mathsf{supp}(F_{n-1}^{\phi\psi})\} \cup & \\ \qquad\qquad \{(x,1) \in \mathbb{F}_2^n : x \notin \mathsf{supp}(F_{n-1}^{\phi\psi})\} & \textit{if } n > 2 \textit{ and odd}, \\ S_n \triangle \{(z,z) \in \mathbb{F}_2^n : z \in \mathsf{supp}(F_{\frac{n}{2}}^{\phi\psi})\} & \textit{if } n > 2 \textit{ and even}. \end{cases}$$

Here $S_n = \cup_{k=0}^n \mathsf{supp}_k(F_n^{\phi\psi})$ and for even $n > 2$, $\mathsf{supp}_k(F_n^{\phi\psi}) = \begin{cases} \mathsf{supp}_k(h_n^{\psi}) & \textit{if } k \textit{ is even} \\ \mathsf{supp}_k(f_n^{\phi}) & \textit{if } k \textit{ is odd.} \end{cases}$

The algorithm for computing $F_n^{\phi\psi}(x), x \in \mathbb{F}_2^n$ is presented in Algorithm 5.5.3.

The time complexity of computing $F_n^{\phi\psi}(x)$ for $x \in \mathbb{F}_2^n$ is $O(n \max\{O(\phi(\frac{n}{2}), \psi(\frac{n}{2}))\})$. If the chosen functions ϕ and ψ are easily computable, then computation would be very fast. If ϕ and ψ are quadratic bent function as taken in Section 7.5.1, the time complexity would be $O(n^2)$. Such efficient functions with good cryptographic properties can be used for the implementation of ciphers for lightweight cryptography.

Algorithm 1 Output of $F_n^{\phi\psi}(x)$

```
Require: $n$;  $x = (x_1, x_2, \dots, x_n) \in \mathbb{F}_2^n$;  $\phi, \psi \in \mathcal{B}_{\frac{n}{2}}$
Ensure: $F_n(x)$
if $n$ is odd then
    $z := (x_1, x_2, \dots, x_{n-1})$
    if $x_n = 0$ then return $F_{n-1}^{\phi\psi}(z)$
    else  return $1 + F_{n-1}^{\phi\psi}(z)$
    end if
else
    $X_{[1,\frac{n}{2}]} := (x_1, x_2, \dots, x_{\frac{n}{2}})$;  $X_{[\frac{n}{2}+1,n]} := (x_{\frac{n}{2}+1}, x_{\frac{n}{2}+2}, \dots, x_n)$
    $k := \mathsf{w_H}(x)$;  $k_1 := \mathsf{w_H}(X_{[1,\frac{n}{2}]})$;  $k_2 := \mathsf{w_H}(X_{[\frac{n}{2}+1,n]})$
    if $k$ is odd then
        if $k_1$ is odd then return $\phi(X_{[\frac{n}{2}+1,n]})$
        else
            return$1 + \psi(X_{[1,\frac{n}{2}]})$
        end if
    else
        if $X_{[1,\frac{n}{2}]} = X_{[\frac{n}{2}+1,n]}$ then
            if $k_1$ is even then
                return $F_{\frac{n}{2}}^{\phi\psi}(X_{[1,\frac{n}{2}]})$
            else
                return $1 + F_{\frac{n}{2}}^{\phi\psi}(X_{[1,\frac{n}{2}]})$
            end if
        else
            $i := 1$
            while $x_i = x_{\frac{n}{2}+i}$ do $i++$ ;
            end while
            if $k_1$ is even then
                if $x_i > x_{\frac{n}{2}+i}$ then
                    return  $\psi(X_{[\frac{n}{2}+1,n]})$
                else
                    return  $\psi(X_{[1,\frac{n}{2}]})$
                end if
            else
                if $x_i > x_{\frac{n}{2}+i}$ then return  $1$
                else  return  $0$
                end if
            end if
        end if
    end if
end if
```

5.5.4 Experimental results on nonlinearity

In this section, we present experimental results on the nonlinearity and weightwise nonlinearity of $F_n^{\phi\psi}$. We have chosen $\phi, \psi \in \mathcal{B}_{\frac{n}{2}}$, a highly nonlinear function;

$$\phi(y) = \psi(y) = \begin{cases} y_1y_2 + \cdots + y_{\frac{n}{2}-1}y_{\frac{n}{2}} & \textit{if } \frac{n}{2} \textit{ is even,} \\ y_1y_2 + \cdots + y_{\frac{n}{2}-2}y_{\frac{n}{2}-1} + y_{\frac{n}{2}} & \textit{if } \frac{n}{2} \textit{ is odd.} \end{cases}$$

This function is a bent function when $\frac{n}{2}$ is even and a concatenation of two bent functions when $\frac{n}{2}$ is odd. Further, these two functions are easy to compute, which is helpful for implementation in lightweight cryptography.

Table 5.7 presents the nonlinearity and weightwise nonlinearity of the functions $F_n^{\phi\psi}$ for $n = 8, 9, \ldots, 16$, which are generated using Construction 5.5.5.

n	NL	NL_2	NL_3	NL_4	NL_5	NL_6	NL_7	NL_8	NL_9	NL_{10}	NL_{11}	NL_{12}	NL_{13}	NL_{14}	$\sum_{k=0}^{n} \mathsf{NL}_k$
8	96	4	16	20	16	4	0	0	-	-	-	-	-	-	60
9	192	6	22	45	45	22	6	0	0	-	-	-	-	-	146
10	416	9	36	69	94	73	12	9	0	0	-	-	-	-	302
11	832	11	50	113	163	173	117	34	11	0	0	-	-	-	672
12	1596	12	36	146	264	286	264	148	36	14	0	0	-	-	1206
13	3192	15	69	219	507	660	660	495	240	69	17	0	0	-	2951
14	6904	19	102	336	764	1083	1484	1079	654	299	30	18	0	0	5868
15	13808	22	147	474	1155	2013	2735	2670	1965	1154	465	75	22	0	12897
16	28152	24	64	564	1216	2547	5036	4610	5036	2919	1216	516	64	24	23836

Table 5.7: Listing of $\mathsf{NL}(F_n^{\phi\psi})$, $\mathsf{NL}_k(F_n^{\phi\psi})$ and $\sum_{k=0}^{n} \mathsf{NL}_k(F_n^{\phi\psi})$ for $8 \le n \le 16$.

We have presented a comparison of weightwise nonlinearities of $F_n^{\phi\psi}$ with the upper bound presented in [20] in Table 5.8. Further, no upper bound is available for the nonlinearity of WAPB Boolean functions. Therefore, we have presented a comparison of nonlinearity of $F_n^{\phi\psi}$ with the upper bound of the nonlinearity of n variable Boolean functions.

A comparison of the nonlinearities of our result with some recent constructions for $n = 8$ are presented in Table 5.9.

n	*function*	NL	NL_2	NL_3	NL_4	NL_5	NL_6	NL_7	NL_8	NL_9	NL_{10}	NL_{11}	$\sum_{k=0}^{n}\mathsf{NL}_k$
8	*UB*	120	11	24	30	24	11	-	-	-	-	-	100
	F_8	96	4	16	20	16	4	-	-	-	-	-	60
9	*UB*	244	15	37	57	57	37	15	-	-	-	-	218
	F_9	192	6	22	45	45	22	6	-	-	-	-	146
10	*UB*	496	19	54	97	118	97	54	19	-	-	-	498
	F_{10}	416	9	36	69	94	73	12	9	-	-	-	302
11	*UB*	1000	23	76	155	220	220	155	76	23	-	-	948
	F_{11}	832	11	50	113	163	173	117	34	11	-	-	672
12	*UB*	2016	28	102	236	381	446	381	236	102	28	-	1940
	F_{12}	1596	12	36	146	264	286	264	148	36	14	-	1206
13	*UB*	4050	34	134	344	625	837	837	625	344	134	34	3948
	F_{13}	3192	15	69	219	507	660	660	495	240	69	17	2951

Table 5.8: Comparison of $\mathsf{NL}_k(F_n^{\phi\psi})$ with the upper bound(UB) presented in [20]

WPB/ WAPB functions	NL	NL_2	NL_3	NL_4	NL_5	NL_6
Upper Bound [20]		11	24	30	24	11
[20]	88	2	12	19	12	6
[66]	96,108	6,9	0,8,14,16, 18,20, 21,22	19,22,23,24, 25,26,27	19,20,21,22	6,9
[98]	88, 90	6,8	8	20, 22, 24	8	6,7
[65, $g_{2^{q+2}}$ Equation(9)]		2	12	19	12	2
[77, f_m Equation(13)]		2	0	3	0	2
[77, g_m Equation(22)]		2	14	19	14	2
[78, f_m Equation(2)]		2	8	8	8	2
[78, f_m Equation(3)]		6	8	26	8	6
[44, Table 1]		5,3,2,2	10,7,12,12	16,15,18,19	12,11,12,12	5,3,2,6
[44, Table 3]		5	16	20	17	5
[104, g_m Equation(11)]		2	12	19	12	6
[46]		6,6,7	19,14,15	21,20,18	11,11,14	3,6,6
[105]		6	17	23	17	6
F_8[Theorem 5.4.3]	82	7	13	14	14	7
F_8 [Construction 5.5.5]	96	4	16	20	16	4

Table 5.9: Comparison of $\mathsf{NL}_k(F_n^{\phi\psi})$ of 8-variable WPB constructions.

5.6 Conclusion

This chapter have explored different cases of Seigenthaler's method to construct WAPB or WPB Boolean function. We have introduced two distinct constructions of WAPB Boolean functions in n variables. The first construction has extended the WPB framework proposed in [77] and have presented a recursive approach based on a smaller WAPB Boolean function defined over n_0-variables, where $n_0 < n$. The second construction is by modifications aimed at improving both the nonlinearity and the weightwise nonlinearities of the resulting functions. Notably, the both the modified construction has achieved a better improved weightwise nonlinearity NL_k for even n and odd k. The algebraic degree, nonlinearity, weightwise nonlinearity, and algebraic immunity of the proposed constructions have been thoroughly analyzed in this chapter.

Chapter 6

On the Direct Sum Approach of WAPB Boolean Functions

6.1 Introduction

For cryptographic use, especially in lightweight ciphers, Boolean functions need to have good cryptographic properties along with a simple way of expression for fast computation, inexpensive implementation, and energy efficiency. The direct sum construction is a fundamental method in the design of Boolean functions. Given two Boolean functions $f \in \mathcal{B}_m$ and $g \in \mathcal{B}_n$, their direct sum is a Boolean function $h \in \mathcal{B}_{m+n}$ defined as $h(x, y) = f(x) + g(y)$, where $x \in \mathbb{F}_2^m, y \in \mathbb{F}_2^n$. Several cryptographic properties, such as nonlinearity and algebraic immunity of the direct sum construction, are defined in Section 2.2.5. If f or g is balanced, then their direct sum $h \in \mathcal{B}_{m+n}$ is also balanced over $\mathbb{F}_2^{m+n}$. However, when f and g are WAPB Boolean functions, it does not necessarily happen that h is a WAPB Boolean function. In the FLIP cipher, the filter function is a direct sum of three functions to achieve the efficiency criteria (refer to the function defined in Equation 2.9). Motivated by this cipher, we have studied the construction of WAPB Boolean functions from the direct sum of two WAPB Boolean functions.

The first construction for the class of WAPB and WPB Boolean functions was introduced in [20] using the direct sum of Boolean functions. The following propositions present some classes of WPB and WAPB Boolean functions by modifying the direct sum of two Boolean functions.

Proposition 6.1.1. *[20] Let $f, f', g, g' \in \mathcal{B}_n$ where f, f', g are WPB Boolean functions.*

Then the function $h \in \mathcal{B}_{2n}$ defined by

$$h(x, y) = f(x) + \prod_{i=1}^{n} x_i + g(y) + (f(x) + f'(x))g'(y), \text{ where } x, y \in \mathbb{F}_2^n,$$

is a WPB Boolean function.

Similarly, another WAPB construction presented in [106] is based on the direct sum of WPB Boolean functions of n_is variables where n_is are distinct powers of 2.

Proposition 6.1.2. *[106] Let $n = n_1 + n_2 + \cdots + n_p$ for n_i being the power of 2 for $1 \leq i \leq p$ and $0 < n_1 < n_2 < \cdots < n_p$. Let $f_{n_i} \in \mathcal{B}_{n_i}$ be WPB with $f_{n_i}(0, 0, \ldots, 0) = 0, f_{n_i}(1, 1, \ldots, 1) = 1$ for $1 \leq i \leq p$. Then $h \in \mathcal{B}_n$ defined as*

$$\begin{aligned} h_n(x_1, x_2, \ldots, x_n) &= f_{n_1}(x_1, x_2, \ldots, x_{n_1}) + f_{n_2}(x_{n_1+1}, x_{n_1+2}, \ldots, x_{n_1+n_2}) + \cdots \\ &\quad + f_{n_p}(x_{n-n_p+1}, x_{n-n_p+2}, \ldots, x_n) \end{aligned}$$

is a WAPB Boolean function.

In this chapter, we present a general result when the direct sum of the WAPB Boolean functions forms a WAPB Boolean function. In Section 6.2, we analyze two cases for which the direct sum of two WAPB Boolean functions is a WAPB or WPB Boolean function. Our results provide more straightforward proof of existing direct sum results of Proposition 6.1.2 and [20, Corollary 1]. We also provide a recursive construction based on direct sum to construct a WPB Boolean function and also study its nonlinearity and algebraic immunity, which are shown to be significantly better. Towards the end of the section, we introduce an elegant technique for obtaining a WAPB Boolean function of any variable n using the direct sum approach. Furthermore, in Section 6.3, we discuss the cryptographic properties of direct sum and provide an improved bound on the weightwise algebraic immunity established in[20].

6.2 Direct sum of WAPB Boolean functions

In this section we have studied on the formation of WAPB Boolean functions from the direct sum of two WAPB Boolean functions.

Theorem 6.2.1. *Let $f \in \mathcal{B}_m,\ g \in \mathcal{B}_n$ be two WAPB Boolean functions. Let $h \in \mathcal{B}_{m+n}$ be defined as $h(x,y) = f(x) + g(y)$ for $x \in \mathbb{F}_2^m$ and $y \in \mathbb{F}_2^n$. Then*

$$\mathsf{w}_{k,n}(h) = \frac{\binom{m+n}{k} - \sum_{i=0}^{k} \delta_i^f \delta_{k-i}^g}{2} \textit{ for } k \in [0, m+n].$$

Proof. As $f \in \mathcal{B}_m, g \in \mathcal{B}_n$ are WAPB Boolean functions, we have from Definition 3.2.1 that

$$\begin{aligned} \mathsf{w}_{k,(}f) &= \frac{\binom{m}{k} + \delta_k^f}{2} \text{ for } k \in [0, m+n] \text{ where } \delta_i^f = 0 \text{ for } i \in [m+1, m+n] \text{ and} \\ \mathsf{w}_{k,(}g) &= \frac{\binom{n}{k} + \delta_k^g}{2} \text{ for } k \in [0, m+n] \text{ where } \delta_i^g = 0 \text{ for } i \in [n+1, m+n]. \end{aligned}$$

Here, we could extend k (in $\mathsf{w}_{k,n}(f)$ and δ_k^f) till $m+n$, as $\binom{m}{k} = 0$ for $k > m$ and $\binom{n}{k} = 0$ for $k > n$.

As $h(x,y) = f(x) + g(y)$, $h(x,y) = 1$ if and only if (i) $f(x) = 1$ and $g(y) = 0$ or, (ii) $f(x) = 0$ and $g(y) = 1$. Hence, the weight of $h(x,y) = f(x) + g(y)$ in the restricted domain of $\mathsf{E}_{n,k}$ is

$$\begin{aligned}
\mathsf{w}_{k,m+n}(h) &= \sum_{i=0}^{k} [\mathsf{w}_{i,m}(f)\mathsf{w}_{k-i,n}(1+g) + \mathsf{w}_{i,m}(1+f)\mathsf{w}_{k-i,n}(g)] \\
&= \sum_{i=0}^{k} \Bigg[\left(\frac{\binom{m}{i} + \delta_i^f}{2} \right) \left(\binom{n}{k-i} - \frac{\binom{n}{k-i} + \delta_{k-i}^g}{2} \right) \\
&\qquad + \left(\binom{m}{i} - \frac{\binom{m}{i} + \delta_i^f}{2} \right) \left(\frac{\binom{n}{k-i} + \delta_{k-i}^g}{2} \right) \Bigg] \\
&= \frac{1}{4} \sum_{i=0}^{k} \Bigg[\left(\binom{m}{i} + \delta_i^f \right) \left(\binom{n}{k-i} - \delta_{k-i}^g \right) \\
&\qquad + \left(\binom{m}{i} - \delta_i^f \right) \left(\binom{n}{k-i} + \delta_{k-i}^g \right) \Bigg] \\
&= \frac{1}{4} \sum_{i=0}^{k} \left[2\binom{m}{i}\binom{n}{k-i} - 2\delta_i^f \delta_{k-i}^g \right] \\
&= \frac{1}{2}\binom{m+n}{k} - \frac{1}{2}\sum_{i=0}^{k} \delta_i^f \delta_{k-i}^g = \frac{\binom{m+n}{k} - \sum_{i=0}^{k} \delta_i^f \delta_{k-i}^g}{2}.
\end{aligned}$$

□

The proof of Theorem 6.2.1 can also be done using Piling-up lemma [73]. Here, the function h (in Theorem 6.2.1) is a WAPB if and only if the chosen functions f, g satisfies $\delta_k^h = -\sum_{i=0}^{k} \delta_i^f \delta_{k-i}^g \in \{-1, 0, 1\}$.

Example 6.2.1. Let $f \in \mathcal{B}_5$ and $g \in \mathcal{B}_4$ be two WAPB Boolean functions and $h \in \mathcal{B}_9$ be defined as $h(x, y) = f(x) + g(y)$ for $x \in \mathbb{F}_2^5, y \in \mathbb{F}_2^4$. Then $\delta_2^f = \delta_3^f = 0$ and $\delta_0^f, \delta_1^f, \delta_4^f, \delta_5^f \in \{-1, 1\}$. Further, $\delta_1^g = \delta_2^g = \delta_3^g = 0$ and $\delta_0^g, \delta_4^g \in \{-1, 1\}$. Here h is a WAPB iff $\delta_k^h = -\sum_{i=0}^{k} \delta_i^f \delta_{k-i}^g \in \{-1, 0, 1\}$ for $k \in [0, 9]$ with $\delta_k^h = 0$ for $k \not\precsim 9$ i.e., for $k = 2, 3, 4, 5, 6, 7$. Then the above conditions form the equations

$\delta_0^h = -\delta_0^f \delta_0^g \implies \delta_0^f \delta_0^g = \pm 1;$

$\delta_1^h = -\delta_0^f \delta_1^g - \delta_1^f \delta_0^g \implies \delta_1^f \delta_0^g = \pm 1;$

$\delta_2^h = -\sum_{i=0}^{2} \delta_i^f \delta_{2-i}^g \implies 0 = 0;$

$\delta^h_3 = -\sum_{i=0}^{3} \delta^f_i \delta^g_{3-i} \implies 0 = 0;$

$\delta^h_4 = -\sum_{i=0}^{4} \delta^f_i \delta^g_{4-i} \implies \delta^f_0\delta^g_4 + \delta^f_4\delta^g_0 = 0;$

$\delta^h_5 = -\sum_{i=0}^{5} \delta^f_i \delta^g_{5-i} \implies \delta^f_1\delta^g_4 + \delta^f_5\delta^g_0 = 0;$

$\delta^h_6 = -\sum_{i=0}^{6} \delta^f_i \delta^g_{6-i} \implies 0 = 0;$

$\delta^h_7 = -\sum_{i=0}^{7} \delta^f_i \delta^g_{7-i} \implies 0 = 0;$

$\delta^h_8 = -\sum_{i=0}^{8} \delta^f_i \delta^g_{8-i} \implies \delta^f_4\delta^g_4 = \pm 1;$

$\delta^h_9 = -\sum_{i=0}^{9} \delta^f_i \delta^g_{9-i} \implies \delta^f_5\delta^g_4 = \pm 1.$

If we consider all f, g and h are balanced, then $\delta^f_0 + \delta^f_1 + \delta^f_4 + \delta^f_5 = \delta^g_0 + \delta^g_4 = \delta^h_0 + \delta^h_1 + \delta^h_8 + \delta^h_9 = 0$. Using all the above equations, we have simplified conditions $\delta^g_0 = -\delta^g_4 = \pm 1$ and $\delta^f_0 = -\delta^f_1 = \delta^f_4 = -\delta^f_5 = \pm 1$. If we have $f(0) = g(0) = 0$ i.e., $\delta^f_0 = \delta^g_0 = -1$. Then the balanced functions $f \in \mathcal{B}_5, g \in \mathcal{B}_4$ satisfying $\delta^f_0 = \delta^f_4 = -1, \delta^f_1 = \delta^f_5 = 1$; and $\delta^g_0 = -1, \delta^g_4 = 1$ result a balanced WAPB $h(x, y) = f(x) + g(y) \in \mathcal{B}_9$.

The terms $\delta^h_k, k \in [0, m+n]$ are the products of the convolutions of the sequences $\{\delta^f_i : i \in [0, m]\}$ and $\{\delta^g_j : j \in [0, n]\}$. Therefore, h is a WAPB if and only if the following matrix multiplication satisfies

$$\Delta^f \delta^g = \delta^h, \text{ that is, } \begin{bmatrix} \delta^f_0 & 0 & 0 & \cdots & 0 \\ \delta^f_1 & \delta^f_0 & 0 & \cdots & 0 \\ \delta^f_2 & \delta^f_1 & \delta^f_0 & \cdots & 0 \\ & & \vdots & & \\ 0 & 0 & \cdots & \delta^f_m & \delta^f_{m-1} \\ 0 & 0 & \cdots & 0 & \delta^f_m \end{bmatrix} \begin{bmatrix} \delta^g_0 \\ \delta^g_1 \\ \vdots \\ \delta^g_n \end{bmatrix} = \begin{bmatrix} \delta^h_0 \\ \delta^h_1 \\ \vdots \\ \delta^h_{m+n} \end{bmatrix}. \tag{6.1}$$

Here, the matrices Δ^f, δ^g and δ^h are of order $(m+n+1) \times (n+1), (n+1) \times 1$ and $(m+n+1) \times 1$ respectively. This multiplication can also be written as

$$\Delta^g \delta^f = \delta^h, \text{ that is, } \begin{bmatrix} \delta^g_0 & 0 & 0 & \cdots & 0 \\ \delta^g_1 & \delta^g_0 & 0 & \cdots & 0 \\ \delta^g_2 & \delta^g_1 & \delta^g_0 & \cdots & 0 \\ & & \vdots & & \\ 0 & 0 & \cdots & \delta^g_n & \delta^g_{n-1} \\ 0 & 0 & \cdots & 0 & \delta^g_n \end{bmatrix} \begin{bmatrix} \delta^f_0 \\ \delta^f_1 \\ \vdots \\ \delta^f_m \end{bmatrix} = \begin{bmatrix} \delta^h_0 \\ \delta^h_1 \\ \vdots \\ \delta^h_{m+n} \end{bmatrix}. \tag{6.2}$$

The matrices Δ^g and δ^f are of order $(m+n+1)\times(m+1)$ and $(m+1)\times 1$ respectively. Here, $\delta_i^f, \delta_j^g, \delta_k^h \in \{-1,0,1\}$ with $\delta_i^f = 0$ if $i \not\preceq m$, $\delta_j^g = 0$ if $j \not\preceq n$ and $\delta_k^h = 0$ if $k \not\preceq m+n$. Now we will study some different cases of m and n.

Theorem 6.2.2. *Let m and n be positive integers such that $e(m)\cap e(n) = \emptyset$. Let $f \in \mathcal{B}_m$ and $g \in \mathcal{B}_n$ be two WAPB Boolean functions. Then $h \in \mathcal{B}_{m+n}$ defined as $h(x,y) = f(x)+g(y)$ for $x \in \mathbb{F}_2^m, y \in \mathbb{F}_2^n$ is a WAPB Boolean function with*

$$\delta_k^h = \begin{cases} 0 & \text{if } e(k) \not\subseteq e(m)\cup e(n) = e(m+n) \text{ i.e., } k \not\preceq m+n \\ -\delta_s^f \delta_{k-s}^g & \text{if } e(k) \subseteq e(m)\cup e(n) = e(m+n) \text{ i.e., } k \preceq m+n \end{cases}$$

where $e(s) = e(k)\cap e(m)$ (i.e., $s = k\&m$ where $\&$ is the bitwise AND operation).

Proof. Since $f \in \mathcal{B}_m$ and $g \in \mathcal{B}_n$ are WAPB Boolean functions, from Lucas' Theorem we have

$$\delta_i^f = \begin{cases} 0 & \text{if } e(i) \not\subseteq e(m) \\ \pm 1 & \text{if } e(i) \subseteq e(m) \end{cases} \quad \text{and} \quad \delta_j^g = \begin{cases} 0 & \text{if } e(j) \not\subseteq e(n) \\ \pm 1 & \text{if } e(j) \subseteq e(n). \end{cases} \tag{6.3}$$

Given that $e(m)\cap e(n) = \emptyset$. That implies, $e(m+n) = e(m)\cup e(n)$. As $h(x,y) = f(x)+g(y)$, for $k \in [0, m+n]$, substituting δ_i^f, δ_j^g from Equation 6.3 in Theorem 6.2.1, we have

$$\mathsf{w}_{k,m+n}(h) = \frac{1}{2}\left[\binom{m+n}{k} - \sum_{i\in[0,k]} \delta_i^f \delta_{k-i}^g\right] = \frac{1}{2}\left[\binom{m+n}{k} - \sum_{\substack{i\in[0,k] \\ e(i)\subseteq e(m) \\ e(k-i)\subseteq e(n)}} \delta_i^f \delta_{k-i}^g\right].$$

Consider $i \in [0,k]$ and $e(i) = \{a_1, a_2, \ldots, a_p\} \subseteq e(m)$. Further consider $e(k-i) = \{b_1, b_2, \ldots, b_q\} \subseteq e(n)$. Then $e(i)\cap e(k-i) = \emptyset$ and $e(k) = e(i)\cup e(k-i) \subseteq e(m)\cup e(n)$. Hence, we have

$$\mathsf{w}_{k,m+n}(h) = \begin{cases} \frac{1}{2}\binom{m+n}{k} & \text{if } e(k) \not\subseteq e(m)\cup e(n) \\ \frac{1}{2}\left[\binom{m+n}{k} - \sum_{\substack{i\in[0,k] \\ e(i)\subseteq e(m), e(k-i)\subseteq e(n)}} \delta_i^f \delta_{k-i}^g\right] & \text{if } e(k) \subseteq e(m)\cup e(n). \end{cases}$$

Moreover from the above argument, we have that if $k = 2^{a_1} + 2^{a_2} + \cdots + 2^{a_p} + 2^{b_1} + 2^{b_2} + \cdots + 2^{b_q}$, then the only case when $e(i) \subseteq e(m), e(k-i) \subseteq e(n)$ is $i = 2^{a_1} + 2^{a_2} + \cdots + 2^{a_p}$ and $k - i = 2^{b_1} + 2^{b_2} + \cdots + 2^{b_q}$ i.e, $e(i) = e(k) \cap e(m)$. Hence,

$$\mathsf{w}_{k,n}(h) = \begin{cases} \frac{1}{2}\binom{m+n}{k} & \text{if } e(k) \not\subseteq e(m) \cup e(n) \\ \frac{1}{2}\left[\binom{m+n}{k} - \delta_i^f \delta_{k-i}^g\right] \text{ where } e(i) = e(k) \cap e(m) & \text{if } e(k) \subseteq e(m) \cup e(n). \end{cases}$$

□

Example 6.2.2. Consider two WAPB Boolean functions $f \in \mathcal{B}_{10}$ and $g \in \mathcal{B}_4$. Then $\delta_i^f = 0$ for $i \in \{1, 3, 4, 5, 6, 7, 9\}$ and $\delta_0^f, \delta_2^f, \delta_8^f, \delta_{10}^f \in \{-1, 1\}$. Similarly, $\delta_j^g = 0$ for $j \in \{1, 2, 3\}$ and $\delta_0^g, \delta_4^g \in \{-1, 1\}$. Here two sets $e(10) = \{3, 1\}$ and $e(4) = \{2\}$ are disjoint. Then $h \in \mathcal{B}_{14}$ such that $h(x, y) = f(x) + g(y)$ is a WAPB with $\delta_k^h = 0$ for $k \in \{1, 3, 5, 7, 9, 11, 13\}$ and $\delta_0^h = -\delta_0^f \delta_0^g, \delta_2^h = -\delta_2^f \delta_0^g, \delta_4^h = -\delta_0^f \delta_4^g, \delta_6^h = -\delta_2^f \delta_6^g, \delta_8^h = -\delta_8^f \delta_0^g, \delta_{10}^h = -\delta_{10}^f \delta_0^g, \delta_{12}^h = -\delta_8^f \delta_4^g, \delta_{14}^h = -\delta_{10}^f \delta_4^g$.

The result on the direct sum of WPB Boolean functions in [106, Theorem 3] (which is stated in Proposition 6.1.2) is a direct consequece of the Theorem 6.2.2. The theorem is stated and proved using our method and notation as follows.

Theorem 6.2.3. *Let n be a positive integer with $e(n) = \{a_1, a_2, \ldots, a_p\}$ with $0 \leq a_1 < a_2 < \cdots < a_p$. Denote $n_i = 2^{a_i}$ for $i \in [1, p]$. Let $f_{n_i} \in \mathcal{B}_{n_i}$ be WPB Boolean functions with $f_{n_i}(0, 0, \ldots, 0) = 0, f_{n_i}(1, 1, \ldots, 1) = 1$ for $1 \leq i \leq p$. Then $h \in \mathcal{B}_n$ defined as*

$$\begin{aligned} h_n(x_1, x_2, \ldots, x_n) &= f_{n_1}(x_1, x_2, \ldots, x_{n_1}) + f_{n_2}(x_{n_1+1}, x_{n_1+2}, \ldots, x_{n_1+n_2}) + \cdots \\ &\quad + f_{n_p}(x_{n-n_p+1}, x_{n-n_p+2}, \ldots, x_n) \end{aligned}$$

is a WAPB, with

$$\delta_k^{h_n} = \begin{cases} 0 & \textit{if } e(k) \not\subseteq e(n) \\ -(-1)^{|e(k)|} = (-1)^{W_(k)+1} & \textit{if } e(k) \subseteq e(n), \end{cases} \tag{6.4}$$

for $k \in [0, n]$.

Proof. It is given that $\delta_0^{f_{n_i}} = -1$ and $\delta_{n_i}^{f_{n_i}} = 1$. We will prove it using induction on $p = |e(n)|$. If $|e(n)| = 1$, it is already a WAPB and satisfying Equation 6.4. Assume that it is true if $|e(n)| \leq p-1$. Let $m = n_1 + n_2 + \cdots + n_{p-1}$ i.e., $e(m) = \{a_1, a_2, \ldots, a_{p-1}\}$. As per the assumption

$$h_m(x_1, \ldots, x_m) = f_{n_1}(x_1, \ldots, x_{n_1}) + \cdots + f_{n_{p-1}}(x_{m-n_{p-1}+1}, \ldots, x_m)$$

is a WAPB Boolean function with

$$\delta_k^{h_m} = \begin{cases} 0 & \text{if } e(k) \not\subseteq e(m) \\ -(-1)^{|e(k)|} & \text{if } e(k) \subseteq e(m), \end{cases}$$

for $k \in [0, m]$. Let $n = m + n_p$ where $n_p = 2^{a_p}$ and $n_p > m$. Further, let $h_n(x_1, \ldots, x_n) = h_m(x_1, \ldots, x_m) + f_{n_p}(x_{m+1}, \ldots, x_n)$. Here, $\delta_0^{f_{n_p}} = -1$ and $\delta_{n_p}^{f_{n_p}} = 1$. As $n_p = 2^{a_p}$ and $n_p > m$, $e(n_p) = \{a_p\}$ and $e(m) \cap e(n_p) = \emptyset$. Hence, for $k \in [0, n]$ (using Theorem 6.2.2),

$$\begin{aligned} \delta_k^{h_n} &= \begin{cases} 0 & \text{if } e(k) \not\subseteq e(n) \\ -\delta_s^{f_{n_p}} \delta_{k-s}^{h_m} \text{ where } e(s) = e(k) \cap e(n_p) & \text{if } e(k) \subseteq e(n). \end{cases} \\ &= \begin{cases} 0 & \text{if } e(k) \not\subseteq e(n) \\ -\delta_0^{f_{n_p}} \delta_k^{h_m} & \text{if } e(k) \subseteq e(n) \text{ and } e(n_p) \cap e(k) = \emptyset \\ -\delta_{n_p}^{f_{n_p}} \delta_{k-n_p}^{h_m} & \text{if } e(k) \subseteq e(n) \text{ and } e(n_p) \cap e(k) = \{a_p\} \end{cases} \\ &= \begin{cases} 0 & \text{if } e(k) \not\subseteq e(n) \\ -(-1)(-(-1)^{|e(k)|}) & \text{if } e(k) \subseteq e(n) \text{ and } e(n_p) \cap e(k) = \emptyset \\ -(1)(-(-1)^{|e(k)|-1}) & \text{if } e(k) \subseteq e(n) \text{ and } e(n_p) \cap e(k) = \{a_p\} \end{cases} \\ &= \begin{cases} 0 & \text{if } e(k) \not\subseteq e(n) \\ -(-1)^{|e(k)|} & \text{if } e(k) \subseteq e(n). \end{cases} \end{aligned}$$

□

Now we will study the possibility of formation of a WPB Boolean function by the direct sum of two WAPB Boolean functions. Let m and n be two nonnegative integers such that $e(m) = \{a_1, a_2, \ldots, a_p\}$ with $0 \leq a_1 < a_2 < \cdots < a_p$ and $e(n) = \{b_1, b_2, \ldots, b_q\}$ with $0 \leq b_1 < b_2 < \cdots < b_q$. If $m + n = 2^l$ for a nonnegative integer l then $e(m) \cap e(n) = \{a_1\}$

(i.e., $a_1 = b_1$), and $e(m) \cup e(n) = \{a_1, a_1+1, a_1+2, \ldots, l-1\}$. For example, if $m = 76$ and $n = 52$ then $e(m) = \{2,3,6\}, e(n) = \{2,4,5\}$ and $m+n = 128 = 2^7$. Here, $e(m) \cap e(n) = \{2\}$ and $e(m) \cup e(n) = \{2,3,4,5,6\}$.

Theorem 6.2.4. *Let m and n be two positive integers such that $m+n = 2^l$ for a nonnegative integer l. Let $f \in \mathcal{B}_m$ and $g \in \mathcal{B}_n$ be two WAPB Boolean functions. Then $h \in \mathcal{B}_{m+n}$ defined as $h(x,y) = f(x) + g(y)$ for $x \in \mathbb{F}_2^m, y \in \mathbb{F}_2^n$ be a WPB Boolean function if there is a $c \in \{-1,1\}$ such that*

$$\begin{aligned} \frac{\delta_0^f}{\delta_m^f} &= -\frac{\delta_0^g}{\delta_n^g}; \\ \frac{\delta^f_{2^{T_1 \setminus \{a_1\}}}}{\delta^f_{2^{T_1}}} &= c \textit{ for every } T_1 \subseteq e(m) \textit{ with } a_1 \in T_1; \\ \frac{\delta^g_{2^{T_2 \setminus \{a_1\}}}}{\delta^g_{2^{T_2}}} &= -c \textit{ for every } T_2 \subseteq e(n) \textit{ with } a_1 \in T_2; \\ \frac{\delta^f_{2^{T_1}}}{\delta^f_{2^{S_1}}} &= -\frac{\delta^g_{2^{S_2}}}{\delta^g_{2^{T_2}}}. \end{aligned}$$

where $S_1 = (T_1 \setminus \{s\}) \cup (e(m) \cap \{a_1, a_1+1, \ldots, s-1\})$ and
$S_2 = (T_2 \setminus \{s\}) \cup (e(n) \cap \{a_1, a_1+1, \ldots, s-1\})$ with s be the smallest integer in $e(k)$.

Proof. Let $e(m) = \{a_1, a_2, \ldots, a_p\}$ with $0 \le a_1 < \cdots < a_p$ and $e(n) = \{b_1, b_2, \ldots, b_q\}$ with $0 \le b_1 < b_2 < \cdots < b_q$. Since $m+n = 2^l$, $e(m) \cap e(n) = \{a_1\}$ (i.e., $a_1 = b_1$), and $e(m) \cup e(n) = \{a_1, a_1+1, a_1+2, \ldots, l-1\}$.

Since $f \in \mathcal{B}_m$ and $g \in \mathcal{B}_n$ are two WAPB Boolean functions, for any $T \subseteq \mathbb{N} \cup \{0\}$ we have

$$\delta^f_{2^T} = \begin{cases} \pm 1 & \text{if } T \subseteq e(m) \\ 0 & \text{if } T \not\subseteq e(m), \end{cases} \quad \text{and} \quad \delta^g_{2^T} = \begin{cases} \pm 1 & \text{if } T \subseteq e(n) \\ 0 & \text{if } T \not\subseteq e(n). \end{cases}$$

Here, $\delta_0^h = \delta_0^f \delta_0^g \in \{-1,1\}$ and $\delta_{m+n}^h = \delta_m^f \delta_n^g \in \{-1,1\}$. For a balanced function h, we have

$$\delta_0^f \delta_0^g = -\delta_m^f \delta_n^g \implies \frac{\delta_0^f}{\delta_m^f} = -\frac{\delta_0^g}{\delta_n^g}. \tag{6.5}$$

For all other cases i.e., $k \in [0, m+n-1]$, we need to find conditions for $\delta_k^h = -\sum_{i=0}^{k} \delta_i^f \delta_{k-i}^g = 0$. We use $m = 76$ and $n = 52$ for demonstration purposes during the following steps. We consider three different cases for $k \in [1, m+n-1]$ as follows.

Case I (When $e(k) \not\subseteq e(m) \cup e(n) = \{a_1, a_1+1, a_1+2, \ldots, l-1\}$): Then $e(k)$ contains an integer $d \in [0, a_1 - 1]$.

If $e(k)$ contains an integer $d \in [0, a_1 - 1]$ then we claim that for every $i \in [0, k]$, atleast one of the $\delta_i^f, \delta_{k-i}^g$ is 0. Let there be an $i \in [0, k]$ such that $\delta_i^f \neq 0$ i.e., $e(i) \subseteq e(m)$. Since $d < a_1$, $e(k-i)$ contains d i.e., $e(k-i) \not\subseteq e(n)$ and that implies $\delta_{k-i}^g = 0$. Hence, $\sum_{i=0}^{k} \delta_i^f \delta_{k-i}^g = 0$. In this case, no condition is formed.

For example, if we take $k = 5$, then $e(k) = \{0, 2\} \not\subseteq e(m) \cup e(n) = \{2, 3, 4, 5, 6\}$ and $d = 0$. Here, $\sum_{i=0}^{5} \delta_i^f \delta_{5-i}^g = \delta_0^f \delta_5^g + \delta_1^f \delta_4^g + \delta_2^f \delta_3^g + \delta_4^f \delta_1^g + \delta_5^f \delta_0^g = 0$ as $\delta_5^g = \delta_1^f = \delta_3^g = \delta_1^g = \delta_5^f = 0$.

Case II (When $e(k) \subseteq e(m) \cup e(n)$ and $a_1 \in e(k)$): Let $T = e(k), T_1 = T \cap e(m)$ and $T_2 = T \cap e(n)$. Here, $T_1 \cap T_2 = \{a_1\}$ and $T_1 \cup T_2 = T$. Now we will find $i \in [0, k]$ such that both $\delta_i^f, \delta_{k-i}^g$ nonzero. Let $\delta_i^f \neq 0$ and $\delta_{k-i}^g \neq 0$ for some $i \in [0, k]$, i.e., $e(i) \subseteq e(m)$ and $e(k-i) \subseteq e(n)$.

Let $a_1 \notin e(i)$. Then $a_1 \in e(k-i)$ as a_1 is smallest integer in $e(m) \cup e(n)$ and $a_1 \in e(k)$. Thus, $e(i)$ and $e(k-i)$ are disjoint. That implies, $e(i) \cup e(k-i) = e(k)$ Hence, $e(k-i) = T_2$ and $e(i) = T_1 \setminus \{a_1\} = T \setminus T_2$. Similarly, if $a_1 \in e(i)$, $e(i) = T_1$ and $e(k-i) = T_2 \setminus \{a_1\}$. Hence, in this case, $-\delta_k^h = \sum_{i=0}^{k} \delta_i^f \delta_{k-i}^g = \delta_{2^{T \setminus T_2}}^f \delta_{2^{T_2}}^g + \delta_{2^{T_1}}^f \delta_{2^{T \setminus T_1}}^g \in \{-2, 0, 2\}$. Therefore, $\delta_k^h = 0$ if $\delta_{2^{T \setminus T_2}}^f \delta_{2^{T_2}}^g + \delta_{2^{T_1}}^f \delta_{2^{T \setminus T_1}}^g = 0$ i.e., if $\frac{\delta_{2^{T \setminus T_2}}^f}{\delta_{2^{T_1}}^f} = -\frac{\delta_{2^{T \setminus T_1}}^g}{\delta_{2^{T_2}}^g}$. Hence,

$$\delta_k^h = 0 \text{ if } \frac{\delta_{2^{T_1 \setminus \{a_1\}}}^f}{\delta_{2^{T_1}}^f} = -\frac{\delta_{2^{T_2 \setminus \{a_1\}}}^g}{\delta_{2^{T_2}}^g}. \tag{6.6}$$

For example, consider $k = 44$ i.e., $e(k) = \{2, 3, 5\}$, with $m = 76$ and $n = 52$. Here $e(k) \subseteq e(m) \cup e(n)$ and $a_1 = 2 \in e(k)$. Then $T_1 = \{2, 3\}$ and $T_2 = \{2, 5\}$. Hence, -$\delta_{44}^h = \sum_{i=0}^{44} \delta_i^f \delta_{44-i}^g = \delta_{2^3}^f \delta_{2^2+2^5}^g + \delta_{2^2+2^3}^f \delta_{2^5}^g = \delta_8^f \delta_{36}^g + \delta_{12}^f \delta_{32}^g \in \{-2, 0, 2\}$. To impose

$\delta^h_{44} = 0$, the WAPB functions f, g need to satisfy $\frac{\delta^f_8}{\delta^f_{12}} = -\frac{\delta^g_{32}}{\delta^g_{36}}$.

Hence, for a fixed $T_2 \subseteq e(n)$ containing a_1, the condition in Equation 6.6 is satisfied for every $T_1 \subseteq e(m)$ containing a_1. Thus, $\frac{\delta^f_{2^{T_1 \setminus \{a_1\}}}}{\delta^f_{2^{T_1}}} = c \in \{-1, 1\}$ is constant for every $T_1 \subseteq e(m)$ containing a_1. Similarly, for a fixed $T_1 \subseteq e(m)$ containing a_1, the condition in Equation 6.6 is satisfied for every $T_2 \subseteq e(n)$ containing a_1. Thus, $\frac{\delta^g_{2^{T_2 \setminus \{a_1\}}}}{\delta^g_{2^{T_2}}} = -c \in \{-1, 1\}$, is constant for every $T_2 \subseteq e(n)$ containing a_1. That is, for a $c \in \{1, -1\}$,

$$\begin{aligned} \frac{\delta^f_{2^{T_1 \setminus \{a_1\}}}}{\delta^f_{2^{T_1}}} &= c \text{ for every } T_1 \subseteq e(m) \text{ with } a_1 \in T_1; \\ \frac{\delta^g_{2^{T_2 \setminus \{a_1\}}}}{\delta^g_{2^{T_2}}} &= -c \text{ for every } T_2 \subseteq e(n) \text{ with } a_1 \in T_2. \end{aligned} \tag{6.7}$$

In the above example, $\frac{\delta^f_0}{\delta^f_4} = \frac{\delta^f_8}{\delta^f_{12}} = \frac{\delta^f_{64}}{\delta^f_{68}} = \frac{\delta^f_{72}}{\delta^f_{76}} = c$ and $\frac{\delta^g_0}{\delta^g_4} = \frac{\delta^g_{16}}{\delta^g_{20}} = \frac{\delta^g_{32}}{\delta^g_{36}} = \frac{\delta^g_{48}}{\delta^g_{52}} = -c$.

Case III (When $e(k) \subseteq e(m) \cup e(n)$ and $a_1 \notin e(k)$): Let denote $T = e(k)$, $T_1 = T \cap e(m)$ and $T_2 = T \cap e(n)$. Here, $T_1 \cap T_2 = \emptyset$ and $T_1 \cup T_2 = T$. Now we will find $i \in [0, k]$ such that both $\delta^f_i, \delta^g_{k-i}$ are nonzero. Let $\delta^f_i \neq 0$ and $\delta^g_{k-i} \neq 0$ for some $i \in [0, k]$. That is, $e(i) \subseteq e(m)$ and $e(k - i) \subseteq e(n)$.

If $a_1 \notin e(i)$ then $e(i) \cup e(k-i) = e(k)$ as $e(i)$ and $e(k-i)$ are disjoint. Hence, $e(i) = T_1$ and $e(k - i) = T_2$ i.e., $i = 2^{T_1}$ and $k - i = 2^{T_2}$. In the example, if we take $k = 24$ i.e., $e(k) = \{3, 4\}$, we have $i = 2^{T_1} = 2^3 = 8$ and $k - i = 2^{T_2} = 2^4 = 16$ where $a_1 = 2 \notin e(i)$. Then $\delta^f_8 \delta^g_{16} \neq 0$.

Further, let $a_1 \in e(i)$. Since a_1 is the smallest integer in $e(m) \cup e(n)$, $a_1 \in e(k - i)$. Let s be the smallest integer in $e(k)$. As $k = i + (k - i)$, $\{a_1, a_1 + 1, \ldots, s - 1\} \subseteq e(i) \cup e(k - i)$ and $s \notin e(i) \cup e(k - i)$. Let denote $S_1 = (T_1 \setminus \{s\}) \cup (e(m) \cap \{a_1, a_1 + 1, \ldots, s - 1\})$ and $S_2 = (T_2 \setminus \{s\}) \cup (e(n) \cap \{a_1, \ldots, s - 1\})$. Hence $i = 2^{S_1}$ and $k - i = 2^{S_2}$

In the example, let take $k = 24$ i.e., $e(k) = \{3, 4\}$. Here, $e(k) \subseteq e(76) \cup e(52)$ and $a_1 = 2 \notin e(k)$. Further, $T_1 = \{3\}, T_2 = \{4\}$. Then the smallest integer in $e(k)$ is $s = 3$. Hence $S_1 = (T_1 \setminus \{3\}) \cup (e(76) \cap \{2\}) = \{2\}$ and $S_2 = (T_2 \setminus \{3\}) \cup (e(52) \cap \{2\}) = \{2, 4\}$.

Hence, for $i = 2^{S_1} = 2^2 = 4$ and $k - i = 2^{S_2} = 2^2 + 2^4 = 20$, $\delta^f_4 \delta^g_{20} \neq 0$. Now combining above two cases for the example $k = 24$, we have $\sum_{i=0}^{k} \delta^f_i \delta^g_{k-i} = \delta^f_{2^{T_1}} \delta^g_{2^{T_2}} + \delta^f_{2^{S_1}} \delta^g_{2^{S_2}} = \delta^f_8 \delta^g_{16} + \delta^f_4 \delta^g_{20}$.

Hence combining two cases, when $e(k) \subseteq e(m) \cup e(n)$ and $a_1 \notin e(k)$, we have $-\delta^h_k = \sum_{i=0}^{k} \delta^f_i \delta^g_{k-i} = \delta^f_{2^{T_1}} \delta^g_{2^{T_2}} + \delta^f_{2^{S_1}} \delta^g_{2^{S_2}} \in \{-2, 0, 2\}$. Therefore, $\delta^h_k = 0$ if $\delta^f_{2^{T_1}} \delta^g_{2^{T_2}} + \delta^f_{2^{S_1}} \delta^g_{2^{S_2}} = 0$. Hence,

$$\sum_{i=0}^{k} \delta^f_i \delta^g_{k-i} = 0 \implies \frac{\delta^f_{2^{T_1}}}{\delta^f_{2^{S_1}}} = -\frac{\delta^g_{2^{S_2}}}{\delta^g_{2^{T_2}}}.$$

□

Example 6.2.3. Consider $m = 12$ and $n = 20$. Then $m + n = 32 = 2^5$ with $e(m) = \{2, 3\}$ and $e(n) = \{2, 4\}$. Here, $e(m) \cup e(n) = \{2, 3, 4\}$ and $a_1 = 2$. Let $f \in \mathcal{B}_{12}$ and $g \in \mathcal{B}_{20}$ be WAPB Boolean functions.

The first condition in Theorem 6.2.4 results that $\frac{\delta^f_0}{\delta^f_{12}} = -\frac{\delta^g_0}{\delta^g_{20}}$.

The second and third conditions in Theorem 6.2.4 result, for a $c \in \{1, -1\}$, that

$$\frac{\delta^f_{2^{\emptyset}}}{\delta^f_{2^{\{2\}}}} = \frac{\delta^f_{2^{\{3\}}}}{\delta^f_{2^{\{2,3\}}}} = \frac{\delta^f_0}{\delta^f_4} = \frac{\delta^f_8}{\delta^f_{12}} = c, \quad \text{and} \quad \frac{\delta^g_{2^{\emptyset}}}{\delta^g_{2^{\{2\}}}} = \frac{\delta^g_{2^{\{4\}}}}{\delta^g_{2^{\{2,4\}}}} = \frac{\delta^g_0}{\delta^g_4} = \frac{\delta^g_{16}}{\delta^g_{20}} = -c.$$

From the fourth condition in Theorem 6.2.4, we have $\frac{\delta^f_8}{\delta^f_4} = -\frac{\delta^g_4}{\delta^g_0}$ (when $e(k) = \{3\}$); $\frac{\delta^f_8}{\delta^f_4} = -\frac{\delta^g_{20}}{\delta^g_{16}}$ (when $e(k) = \{3, 4\}$); $\frac{\delta^f_0}{\delta^f_{12}} = -\frac{\delta^g_4}{\delta^g_{16}}$ (when $e(k) = \{4\}$). Now combining all equations, we have $\frac{\delta^f_0}{\delta^f_{12}} = -\frac{\delta^g_0}{\delta^g_{20}} = -\frac{\delta^g_4}{\delta^g_{16}}$ and $\frac{\delta^f_0}{\delta^f_4} = \frac{\delta^f_8}{\delta^f_{12}} = \frac{\delta^f_4}{\delta^f_8} = -\frac{\delta^g_0}{\delta^g_4} = -\frac{\delta^g_{16}}{\delta^g_{20}}$. If we consider f such that $\delta^f_0 = x$ and $\delta^f_{12} = y$ for some $x, y \in \{1, -1\}$ then $\frac{x}{\delta^f_4} = \frac{\delta^f_8}{y} = \frac{\delta^f_4}{\delta^f_8} = c \implies c = xy$. Hence $\delta^f_8 = x$ and $\delta^f_4 = y$. Furthermore, if we consider $\delta^g_0 = z \in \{1, -1\}$ then $\delta^g_{16} = z$ and $\delta^g_4 = \delta^g_{20} = -xyz$. Hence, for WAPB f, g satisfying $\delta^f_0 = \delta^f_8 = x$; $\delta^f_4 = \delta^f_{12} = y$; $\delta^g_0 = \delta^g_{16} = z$ and $\delta^g_4 = \delta^g_{20} = -xyz$ for $x, y, z \in \{1, -1\}$ results in a WPB Boolean function $h \in \mathcal{B}_{32}$.

Now we have a consequence of the possibility of existence of WPB function due to the direct sum of two WAPB Boolean functions, i.e., when $m = n = 2^{l-1}$.

Lemma 6.2.5. *Let $n = 2^{l-1}$ be a positive integer and $f, g \in \mathcal{B}_n$ be two WAPB Boolean functions. Then $h \in \mathcal{B}_{2n}$ defined as $h(x, y) = f(x) + g(y)$ for $x, y \in \mathbb{F}_2^n$ be a WPB Boolean function if $\frac{\delta_0^f}{\delta_n^f} = -\frac{\delta_0^g}{\delta_n^g}$.*

Proof. Since $f, g \in \mathcal{B}_n$ are WAPB and n is a power of 2, $\delta_0^f, \delta_n^f, \delta_0^g, \delta_n^g \in \{1, -1\}$ and $\delta_i^f = \delta_i^g = 0$ for $i \in [1, n-1]$. Hence, from Theorem 6.2.1, we have

$$\mathsf{w}_{k,2n}(h) = \begin{cases} \frac{1}{2}\binom{2n}{k} & \text{if } k \neq 0, n, 2n, \\ \frac{1}{2}\left[\binom{2n}{k} - \delta_0^f\delta_0^g\right] & \text{if } k = 0, \\ \frac{1}{2}\left[\binom{2n}{k} - (\delta_0^f\delta_n^g + \delta_n^f\delta_0^g)\right] & \text{if } k = n \\ \frac{1}{2}\left[\binom{2n}{k} - \delta_n^f\delta_n^g\right] & \text{if } k = 2n. \end{cases}$$

Here, for $k = 0$ and $2n$, the value of $\delta_0^f\delta_0^g, \delta_n^f\delta_n^g$ are aleady in $\{1, -1\}$. For $k = n$, $\delta_0^f\delta_n^g + \delta_n^f\delta_0^g \in \{-2, 0, 2\}$. For h being a WAPB, $\delta_0^f\delta_n^g + \delta_n^f\delta_0^g = 0$ i.e., $\frac{\delta_0^f}{\delta_n^f} = -\frac{\delta_0^g}{\delta_n^g}$. Further if $\frac{\delta_0^f}{\delta_n^f} = -\frac{\delta_0^g}{\delta_n^g}$, we have $\frac{\delta_0^f\delta_0^g}{\delta_n^f\delta_n^g} = -\frac{(\delta_0^g)^2}{(\delta_n^g)^2} = -1 \implies \delta_0^f\delta_0^g = -\delta_n^f\delta_n^g \implies \delta_0^h = -\delta_{2n}^h$. Hence h is balanced and results h is a WPB Boolean function.

Alternative proof: From Equation 6.1, the convolutions of the sequences $\{\delta_i^f : i \in [0, m]\}$ and $\{\delta_j^g : j \in [0, n]\}$ produces the sequence $\{\delta_0^f\delta_0^g, 0, 0, \ldots, 0, \delta_n^f\delta_n^g\}$. Hence, the direct sum of f and g, i.e., $h \in \mathcal{B}_{2n}$ such that $h_{2n}(x, y) = f_n(x) + g_n(y)$, is a WAPB. Further, h is a WPB if $\delta_0^f\delta_0^g + \delta_n^f\delta_n^g = 0$ i.e., $\frac{\delta_0^f}{\delta_n^f} = -\frac{\delta_n^g}{\delta_0^g} = -\frac{\delta_0^g}{\delta_n^g}$. □

In the above case, h will be a WPB Boolean function if one of f and g is balanced (i.e., WPB) and the other one is not balanced (i.e., a WAPB but not WPB). However, we can construct a WPB Boolean function h from two WPB Boolean functions f and g as in the following corollary which is presented in [20].

Corollary 6.2.6. *[20] Let $f, g \in \mathcal{B}_n$ be two WPB Boolean functions where $n = 2^l$. Then $h \in \mathcal{B}_{2n}$ defined as $h(x, y) = f(x) + g(y) + y_1y_2\cdots y_n$ for $x, y = (y_1, y_2, \ldots, y_n) \in \mathbb{F}_2^n$ be a WPB Boolean function.*

Proof. As $f, g \in \mathcal{B}_n$ are WPB Boolean functions, $\hat{g}(x_1, x_2, \ldots, x_n) = g(x_1, x_2, \ldots, x_n) + x_1 x_2 \cdots x_n$ is a unbalanced WAPB Boolean function. Then $\frac{\delta_0^f}{\delta_n^f} = -1$ and $\frac{\delta_0^{\hat{g}}}{\delta_n^{\hat{g}}} = 1$. Hence, from Lemma 6.2.5, we have $h(x, y) = f(x) + \hat{g}(y)$ for $x, y \in \mathbb{F}_2^n$ is a WPB Boolean function with $\delta_0^h = -\delta_0^f \delta_0^{\hat{g}} = -\delta_0^f \delta_0^g$ and $\delta_{2n}^h = -\delta_n^f \delta_n^{\hat{g}} = \delta_n^f \delta_n^g$. □

We will study some cryptographic properties of a class WPB functions recursively generated using Lemma 6.2.5 or, Corollary 6.2.6.

Theorem 6.2.7. *For $n = 2^l, l \geq 1$, let $f_n \in \mathcal{B}_n$ be defined recursively as*

$$f_n(x_1, \ldots, x_n) = f_{\frac{n}{2}}(x_1, \ldots, x_{\frac{n}{2}}) + f_{\frac{n}{2}}(x_{\frac{n}{2}+1}, \ldots, x_n) + \prod_{i=\frac{n}{2}+1}^{n} x_i, \text{ for } l \geq 2 \text{ and}$$

$f_2(x_1, x_2) = x_2$. Then

1. *f_n is WPB.*
2. *$f_n(x_1, \ldots, x_n) = \sum_{2^1|i} x_i + \sum_{2^2|i} x_{i-1}x_i + \sum_{2^3|i} x_{i-3}x_{i-2}x_{i-1}x_i + \cdots + x_{\frac{n}{2}+1} \cdots x_n$.*
3. *$\mathsf{NL}(f_n) = 2^{n-1} - \frac{1}{2}(3^{\frac{n}{2}} - 1)$.*
4. *$\mathsf{AI}(f_n) \leq 1 + \frac{n}{4}$.*

Proof.

1. Here, f_2 is WPB and using Corollary 6.2.6, it can be proved inductively that f_n for $n = 2^l, l \geq 2$ is WPB.

2. The ANF of f_n can inductively be proved with the base case $f_2 \in \mathcal{B}_2$ such that $f_2(x_1, x_2) = x_2$. Assume that

   $$f_m(x_1, \ldots, x_m) = \sum_{2^1|i} x_i + \sum_{2^2|i} x_{i-1}x_i + \sum_{2^3|i} x_{i-3}x_{i-2}x_{i-1}x_i + \cdots + x_{\frac{m}{2}+1} \cdots x_m.$$

   Then $f_{2m}(x_1, \ldots, x_{2m}) = f_m(x_1, \ldots, x_m) + f_m(x_{m+1}, \ldots, x_{2m}) + \prod_{i=m+1}^{2m} x_i$.

   As m is a power of 2, the result follows.

3. For $n = 2^l, l \geq 1$, let $l_n(x_1, \ldots, x_n) = \sum_{2|i} x_i$ and $g_n(x_1, \ldots, x_n) = f_n + l_n = g_{\frac{n}{2}}(x_1, \ldots, x_{\frac{n}{2}}) + g_{\frac{n}{2}}(x_{\frac{n}{2}+1}, \ldots, x_n) + x_{\frac{n}{2}+1} \cdots x_n$. Here, l_n and g_n are linear and nonlinear parts of f_n, respectively. Further, denote $g'_n(x_1, \ldots, x_n) = g_n(x_1, \ldots, x_n) + x_1 x_2 \cdots x_n$. That is, $g_n(x_1, \ldots, x_n) = g_{\frac{n}{2}}(x_1, \ldots, x_{\frac{n}{2}}) + g'_{\frac{n}{2}}(x_{\frac{n}{2}+1}, \ldots, x_n)$. Hence, $\mathsf{w_H}(g'_n) = \mathsf{w_H}(g_n + x_1 x_2 \cdots x_n) = \mathsf{w_H}(g_n) - 1$

   We will prove that l_n is a nearest linear function from f_n and for that reason we will find $\mathsf{w_H}(g_n)$. We can check that $\mathsf{w_H}(g_2) = 0$ and $\mathsf{w_H}(g_4) = \mathsf{w_H}(x_3 x_4) = 4$. From the ANF of $g_n, n \geq 4$, we can compute $g_n(1, 1, \ldots, 1) = (\sum_{2^2|i} 1 + \sum_{2^3|i} 1 + \cdots + 1) \bmod 2 = 1$ as every summations but the last one contains even number of 1s. Therefore, $\mathsf{w_H}(g_n)$ satisfies the recursion

$$\mathsf{w_H}(g_n) = \mathsf{w_H}(g_{\frac{n}{2}})\mathsf{w_H}(\overline{g'_{\frac{n}{2}}}) + \mathsf{w_H}(\overline{g_{\frac{n}{2}}})\mathsf{w_H}(g'_{\frac{n}{2}}) \text{ for } n = 2^l, l > 2.$$

   Denote the sequence $\mathsf{w_H}(g_n) = w_l$ for $n = 2^l$. Then $\mathsf{w_H}(\overline{g_n}) = 2^n - w_l, \mathsf{w_H}(g'_n) = w_l - 1, \mathsf{w_H}(\overline{g'_n}) = 2^n - w_l + 1$. Hence, we have

$$\begin{aligned}
w_{l+1} &= w_l(2^{2^l} - w_l + 1) + (2^{2^l} - w_{)}(w_l - 1) \\
&= -2w_l^2 + 2(2^{2^l} + 1)w_l - 2^{2^l} \\
\implies -2w_{l+1} &= 4w_l^2 - 4(2^{2^l} + 1)w_l + 2^{2^l+1} \\
&= (2w_l)^2 - 2(2^{2^l} + 1)2w_l + (2^{2^l} + 1)^2 - (2^{2^l} + 1)^2 + 2^{2^l+1} \\
&= (2^{2^l} + 1 - 2w_l)^2 - (2^{2^{l+1}} + 1) \\
\implies 2^{2^{l+1}} + 1 - 2w_{l+1} &= (2^{2^l} + 1 - 2w_l)^2 \\
\implies z_{l+1} &= z_l^2 \text{ where } z_l = 2^{2^l} + 1 - 2w_l.
\end{aligned}$$

   Here $z_2 = 2^{2^2} + 1 - 2w_2 = 2^4 + 1 - 8 = 9$. Using the back recursion, $z_1 = 3$. Then solving the nonlinear recursion [2], $z_l = z_{l-1}^2, l \geq 2$ and $z_1 = 3$, we have $z_l = 3^{2^{l-1}}$. Hence, $\mathsf{w_H}(g_n) = w_l = \frac{2^{2^l}+1-z_l}{2} = \frac{2^{2^l}+1-3^{2^{l-1}}}{2} = 2^{2^l-1} - \frac{1}{2}(3^{2^{l-1}} - 1) = 2^{n-1} -$

$\frac{1}{2}(3^{\frac{n}{2}}-1)$. As $g_n(x_1,\ldots,x_n) = f_n(x_1,\ldots,x_n) + \sum_{i|n} x_i$, $\mathsf{NL}(f_n) \leq \mathsf{w_H}(g_n) = 2^{n-1} - \frac{1}{2}(3^{\frac{n}{2}}-1)$ for $n \geq 4$.

We can check that $l_4(x_1,\ldots,x_4) = x_2 + x_4$ is a nearest linear function with distance $2^3 - \frac{1}{2}(3^2-1) = 4$. Let assume that $\mathsf{NL}(f_n) = 2^{n-1} - \frac{1}{2}(3^{\frac{n}{2}}-1)$ with a nearest linear function l_n. Then $\mathsf{NL}(f_n + x_1x_2\cdots x_n) = 2^{n-1} - \frac{1}{2}(3^{\frac{n}{2}}-1) - 1$ with a nearest linear function l_n. As f_{2n} is a direct sum of f_n and $f_n + x_1x_2\cdots x_n$, from Proposition 6.3.1, $\mathsf{NL}(f_{2n}) = 2^n[\mathsf{NL}(f_n) + \mathsf{NL}(f_n + x_1x_2\cdots x_n)] - 2\mathsf{NL}(f_n)\mathsf{NL}(f_n + x_1x_2\cdots x_n) = 2^n[2^{n-1} - \frac{1}{2}(3^{\frac{n}{2}}-1) + 2^{n-1} - \frac{1}{2}(3^{\frac{n}{2}}-1) - 1] - 2(2^{n-1} - \frac{1}{2}(3^{\frac{n}{2}}-1))(2^{n-1} - \frac{1}{2}(3^{\frac{n}{2}} - 1) - 1) = 2^n(2^n - 3^{\frac{n}{2}}) - \frac{1}{2}(2^n - 3^{\frac{n}{2}} + 1)(2^n - 3^{\frac{n}{2}} + 1) = 2^{2n-1} - \frac{1}{2}(3^n - 1)$.

4. We can verify from the ANF of f_n that $(1 + \sum_{2|i} x_i)(1 + x_4)(1 + x_8)\cdots(1 + x_n) = (1 + \sum_{2|i} x_i)\prod_{4|i}(1 + x_i)$ is an annihilator of f_n. Hence, $\mathsf{AI} \leq 1 + \frac{n}{4}$.

□

The following lemma presents a method for getting an n variable WPB Boolean function from the concatenation of two $n-1$ variable WAPB Boolean functions.

Lemma 6.2.8. *Let $n = 2^l$ and $f, g \in \mathcal{B}_{n-1}$ are WAPB functions with $\delta_i^f = -\delta_{i-1}^f, \delta_i^f = \delta_i^g$ for $i \in [1, n]$. Then $h \in \mathcal{B}_n$ defined as $h(x_1, x_2, \ldots, x_n) = x_n f(x_1, x_2, \ldots, x_{n-1}) + (1 + x_n)g(x_1, x_2, \ldots, x_{n-1})$ is WPB.*

Proof. Here, $\mathsf{w}_{k,n}(h) = \mathsf{w}_{k,n}(f) + \mathsf{w}_{k-1,n}(g)$ for $k \in [1, n-1]$, $\mathsf{w}_{0,n}(h) = \mathsf{w}_{0,n}(f)$ and $\mathsf{w}_{n,n}(h) = \mathsf{w}_{n-1,n}(g)$.

Case $k = 0, n$: $\mathsf{w}_{0,n}(h) = \mathsf{w}_{0,n}(f)$ and $\mathsf{w}_{n,n}(h) = \mathsf{w}_{n-1,n}(g)$. Hence, $\delta_0^h = \delta_0^f \in \{-1, 1\}$ and $\delta_n^h = \delta_{n-1}^g \in \{-1, 1\}$.

Case $k \in [1, n-1]$: $\mathsf{w}_{k,n}(h) = \mathsf{w}_{k,n}(f) + \mathsf{w}_{k-1,n}(g) = \frac{1}{2}\left[\binom{n-1}{k} + \delta_k^f\right] + \frac{1}{2}\left[\binom{n-1}{k-1} + \delta_{k-1}^g\right]$ $= \frac{1}{2}\left[\binom{n}{k} + \delta_k^f + \delta_{k-1}^g\right] = \frac{1}{2}\left[\binom{n}{k} + \delta_k^f + \delta_{k-1}^f\right] = \frac{1}{2}\binom{n}{k}$. Hence, $\delta_k^h = 0$ for $k \in [1, n]$ and that implies h is a WPB Boolean function on n-variables. □

Now we are defining a class of WAPB Boolean functions, which will help us to generate WPB Boolean functions.

Definition 6.2.1. An WAPB Boolean function $f \in \mathcal{B}_n$ satisfying $\delta_i^f = -\delta_{i-1}^f$ for $i \in [1, n]$ (i.e., $\delta_i^f = (-1)^i \delta_0^f$, for $i \in [0, n]$) is defined as an alternating WAPB (AWAPB) Boolean function.

From Lucas Theorem (Proposition 3.2.1), it can be checked that an n-variable AWAPB Boolean function exists if and only if n is a positive integer of the form $2^l - 1$. We present a construction of an AWAPB Boolean function in $2^l - 1$ variables from two WAPB functions in the following lemma.

Lemma 6.2.9. *Let $n = 2^l$ be a positive integer. Let $f \in \mathcal{B}_{n-1}$ AWAPB Bolean function and $g \in \mathcal{B}_n$ be an unbalanced WAPB Boolean function. Then the direct sum $h \in \mathcal{B}_{2n-1}$ defined as $h(x, y) = f(x) + g(y)$ for $x \in \mathbb{F}_2^{n-1}, y \in \mathbb{F}_2^n$ is a AWAPB Boolean function if g is not balanced i.e., $\delta_0^g = \delta_n^g$.*

Proof. Since $f \in \mathcal{B}_{n-1}$ is AWAPB and $g \in \mathcal{B}_n$ is unbalanced WAPB, $\delta_i^f = (-1)^i \delta_0^f$ for $i \in [1, n-1]$, $\delta_i^g = 0$ for $i \in [1, n-1]$ and $\delta_0^g = \delta_n^g$. As $e(n) \cap e(n-1) = \emptyset$, for $k \in [0, 2n-1]$, we have (from Theorem 6.2.2)

$$\begin{aligned}
\delta_k^h &= \begin{cases} 0 & \text{if } e(k) \not\subseteq e(2n-1) \\ -\delta_s^f \delta_{k-s}^g \text{ where } e(s) = e(k) \cap e(n-1) & \text{if } e(k) \subseteq e(2n-1) \end{cases} \\
&= \begin{cases} 0 & \text{if } e(k) \not\subseteq e(2n-1) \\ -\delta_k^f \delta_0^g = (-1)^{k+1} \delta_0^f \delta_0^g & \text{if } k < n \text{ and } e(k) \subseteq e(2n-1) \\ -\delta_{k-n}^f \delta_n^g = (-1)^{k-n+1} \delta_0^f \delta_n^g & \text{if } k \geq n \text{ and } e(k) \subseteq e(2n-1). \end{cases}
\end{aligned}$$

As $e(k) \subseteq e(2n-1) = e(2^{l+1} - 1)$ for every $k \in [0, 2n-1]$, the case $\delta_k^h = 0$ will never occur. Further, as n is an even integer for $n > 1$, $(-1)^{k-n+1} = (-1)^{k+1}$ for $k \in [n, 2n-1]$. Hence,

$$\delta_k^h = \begin{cases} 0 & \text{if } e(k) \not\subseteq e(2n-1), \text{which will never occur,} \\ (-1)^{k+1} \delta_0^f \delta_0^g & \text{if } k < n \text{ and } e(k) \subseteq e(2n-1), \\ (-1)^{k+1} \delta_0^f \delta_n^g & \text{if } k \geq n \text{ and } e(k) \subseteq e(2n-1). \end{cases} .$$

Hence, if $\delta_n^g = \delta_0^g$ (i.e., g is not balanced), $\delta_k^h = (-1)^{k+1}\delta_0^f\delta_0^g$ which implies that h is a AWAPB with $\delta_0^h = -\delta_0^f\delta_0^g$. □

If g is WPB, then it can be made as a unbalanced WPB by adding the term $\prod_{i=1}^n y_i$ with g and hence we have the following corollary.

Corollary 6.2.10. *Let $n = 2^l$ be a positive integer. Let $f \in \mathcal{B}_{n-1}$ be an AWAPB Bolean function and $g \in \mathcal{B}_n$ be a WPB Boolean function. Then $h \in \mathcal{B}_{2n-1}$ defined as $h(x, y) = f(x) + g(y) + \prod_{i=1}^n y_i$ for $x \in \mathbb{F}_2^{n-1}, y \in \mathbb{F}_2^n$ is a AWAPB Boolean function.*

Using Lemma 6.2.8 and Corollary 6.2.10, we can recursively generate AWAPB and WPB Boolean functions as illustrated in the following example.

Example 6.2.4. Consider $f_1 \in \mathcal{B}_1$ such that $f_1(x_1) = x_1$. Here f_1 is an AWAPB Boolean function with $\delta_0^f = -1$ and $\delta_1^f = 1$. Considering $f = g = f_1$ in Lemma 6.2.8, we have $f_2(x_1, x_2) = x_2x_1 + (1 + x_2)x_1 = x_1$ is a WPB in $\mathcal{B}_2$. Now considering, $f = f_1, g = f_2$ in Corollary 6.2.10, we have $f_3(x_1, x_2, x_3) = x_1 + x_2 + x_2x_3$ is AWAPB in $\mathcal{B}_3$.

Further, instead of using the same function f_3 for f and g in Lemma 6.2.8, we consider $f(x) = f_3(x)$ and $g(x) = f_3(Ax)$ where A is a permutation matrix which permutes the cordinates of the input vector x. As a result g is too an AWAPB Boolean function with $\delta_i^f = \delta_i^g$ for $i \in [0, n]$. Here, consider $g(x_1, x_2, x_3) = x_1 + x_3 + x_2x_3$. Then $f_4(x_1, x_2, x_3, x_4) = x_4f(x_1, x_2, x_3) + (1 + x_4)g(x_1, x_2, x_3) = x_1 + x_2 + x_2x_3 + x_2x_4 + x_3x_4$ is WPB.

Then considering, $f = f_3, g = f_4$ in Corollary 6.2.10, we have

$$f_7(x_1, x_2, \ldots, x_7) = f_3(x_1, x_2, x_3) + f_4(x_4, x_5, x_6, x_7) + x_4x_5x_6x_7$$

is an AWAPB Boolean function in 7 variables. Similarly, using Lemma 6.2.8 and Corollary 6.2.10 alternatively, we can generate WPB and AWAPB Boolean functions in higher number of variables.

6.3 Cryptographic properties of the Direct sum

Now we will study and collect some existing results on the cryptographic properties of the direct sum of Boolean functions. We consider $f \in \mathcal{B}_m, g \in \mathcal{B}_n$ and $h \in \mathcal{B}_{m+n}$ defined as $h(x,y) = f(x) + g(y)$ for $x \in \mathbb{F}_2^m$ and $y \in \mathbb{F}_2^n$ in the following results. We also consider $a = (a', a'') \in \mathbb{F}_2^{m+n}$ where $a' \in \mathbb{F}_2^m$ and $a'' \in \mathbb{F}_2^n$.

Proposition 6.3.1. *[93]*

1. *The Walsh transform of h given as $W_h(a) = W_f(a')W_g(a'')$.*
2. *The nonlinearity of h is given as $\mathsf{NL}(h) = 2^n\mathsf{NL}(f) + 2^m\mathsf{NL}(g) - 2\mathsf{NL}(f)\mathsf{NL}(g)$.*

Proposition 6.3.2. *[20] For $k \in [0, m+n]$,*

1. *the Walsh transform of h over $E_{m+n,k}$ is given as*
$$W_{h,k}(a) = \sum_{i=0}^{k} W_{f,i}(a')W_{g,k-i}(a'').$$
2. *a bound on nonlinearity over $E_{m+n,k}$ is given as*
$$\mathsf{NL}_k(f) \geq \sum_{i=0}^{k}\left(\binom{m}{i}\mathsf{NL}_{k-i}(g) + \binom{n}{k-i}\mathsf{NL}_i(f) - 2\mathsf{NL}_i(f)\mathsf{NL}_{k-i}(g)\right).$$

Proposition 6.3.3.

1. *[13]* $\max(\mathsf{AI}(f), \mathsf{AI}(g)) \leq \mathsf{AI}(h) \leq \min\{\max\{\deg(f), \deg(g)\}, \mathsf{AI}(f) + \mathsf{AI}(g)\}$.
2. *[20] For* $1 \leq k \leq \min\{m,n\}, \mathsf{AI}_k(h) \geq \min_{0 \leq j \leq k}\{\max\{\mathsf{AI}_j(f), \mathsf{AI}_{k-j}(g)\}\}$.

The Item 2 of Proposition 6.3.3 can be generalized as follows.

Theorem 6.3.4. *For $0 \leq k \leq m+n$,*

$$\min_{\max\{0,k-m\} \leq j \leq \min\{m,k\}}\{\max\{\mathsf{AI}_j(f), \mathsf{AI}_{k-j}(g)\}\} \leq \mathsf{AI}_k(h) \leq \deg(h).$$

Proof. Let A_k be an annihilator of h over $E_{m+n,k}$. Then $A_k(x,y)h(x,y) = 0$ for all $(x,y) \in E_{m+n,k}$ and there exists an $(x_0,y_0) \in E_{m+n,k}$ such that $A_k(x_0,y_0) = 1$. Let $W_(x_0) = j$ (i.e., $x_0 \in E_{m,j}$) for some $\max\{0, k-m\} \leq j \leq \min\{m,k\}$. As $A_k(x_0,y_0) = 1$, $h(x_0,y_0) = f(x_0) + g(y_0) = 0$ i.e., either $f(x_0) = g(y_0) = 0$ or, $f(x_0) = g(y_0) = 1$. Now fixing x_0, we have $A_k(x_0,y)(f(x_0)+g(y)) = 0$ for all $y \in E_{n,k-j}$. That implies, $A_k(x_0,y) = A_{k,x_0}(y)$ is an annihilator of g or, $1+g$ over $E_{n,k-j}$ (as $A_k(x_0,y_0) = 1$). Similarly, fixing y_0, we have $A_k(x,y_0)(f(x)+g(y_0)) = 0$ for all $x \in E_{n,j}$. That implies, $A_{k,y_0}(x)$ is an annihilator of f or, $1+f$ over $E_{n,j}$. As a summary, if A_k is an annihilator of h over $E_{m+n,k}$, then there is a $j \in [\max\{0,k-m\}, \min\{m,k\}]$ such that A_{k,y_0} is an annihilator of f or, $1+f$ over $E_{n,j}$ and A_{k,x_0} is an annihilator of g or, $1+g$ over $E_{n,k-j}$. Hence, $\deg A_k \geq \mathsf{AI}_k(h) \geq \min_{\max\{0,k-m\} \leq j \leq \min\{m,k\}} \{\max\{\mathsf{AI}_j(f), \mathsf{AI}_{k-j}(g)\}\}$.
$1+h$ is an annihilator of h but not necessarily $1+h$ is an annihilator of h in the domain $E_{n,k}$ if $1+h$ is 0 in the domain $E_{n,k}$. In our case, h is a WPB and hence is a balance function in each domain $E_{n,k}$ for $k \in [1, n-1]$. That implies, $1+h$ is not 0 in each domain $E_{n,k}$ for $k \in [1, n-1]$ and hence $1+h$ is an annihilator of h in the domain $E_{n,k}$ and $\mathsf{AI}_k(h) \leq \deg(h)$. □

6.4 Conclusion

We have studied the direct sum of two WAPB/WPB Boolean functions and conditioned when it obtains a new WAPB/WPB Boolean function. Some of the results presented in [20, 106] are consequences of our results. We have presented some constructions of WAPB/WPB Boolean functions in this direction. There is still an open problem to study the direct sum $h(x,y) = f(x) + g(y)$ in Theorem 6.2.2 when $e(m) \cap e(n) \neq \emptyset$.

Chapter 7

Bulding WAPB Boolean Functions From Permutation Group Actions

7.1 Introduction

Various classes of Boolean functions based on the actions of permutation groups are available in the literature. For example, a prominent class of Boolean functions, such as symmetric Boolean functions [16] and rotation symmetric Boolean functions [96, 97, 59], belong to this category (see Definition 7.1.1). These functions are invariant under any permutation and cyclic rotation of the input coordinates, respectively. These functions often possess exceptional properties, making them highly valuable for a wide range of practical applications. For example, rotation symmetric Boolean functions have played a crucial role in exceeding the quadratic bound of nonlinearity [59].

In this chapter, we propose a general method for building a class of WAPB Boolean functions using the action of a cyclic permutation group on $\mathbb{F}_2^n$. This generalizes the construction of WPB Boolean functions (in $n = 2^m$ variables) by Liu and Mesnager in [66] to construct WAPB Boolean functions (in any n variables). The class of WPB functions introduced in [66] are 2-rotation symmetric (as defined in Definition 7.1.2). The authors also examined the weightwise nonlinearity profile of this class of WPB functions and provided a constant lower bound on k-weightwise nonlinearity, where k is a power of 2. In our work, we propose a bound for the nonlinearity and k- weightwise nonlinearities of functions from this construction. Additionally, we explore two significant permutation groups, $\langle\psi\rangle$ and $\langle\sigma\rangle$, where ψ is a distinct binary-cycle permutation and σ is a rotation. We theoretically analyze the cryptographic properties of the WAPB functions derived from these permutations and experimentally evaluate their nonlinearity parameters for n between 4 and 10. The chapter is based on the results presented in our work [31].

7.1.1 Permutation symmetric Boolean functions

We denote $\mathbb{S}_n$ by the symmetric group on n elements. For a permutation $\pi \in \mathbb{S}_n$, the action of permutation group $P = \langle \pi \rangle$ on $\mathbb{F}_2^n$ makes a partition of $\mathbb{F}_2^n$, called orbits. The set of orbits is denoted as $\mathcal{O}$. For $x \in \mathbb{F}_2^n$, we denote the orbit containing x as $O_\pi(x) = \{y \in \mathbb{F}_2^n \mid \pi^k(y) = x \text{ for some } k \in \mathbb{Z}\}$. Now we define two special permutations as follows.

1. **Rotation (or, cyclic) permutation (σ)**: The permutation $\sigma \in \mathbb{S}_n$ is called rotation permutation if

   $$\sigma((x_1, x_2, \ldots, x_n)) = (x_n, x_1, \ldots, x_{n-1})$$

   for every $(x_1, x_2, \ldots, x_n) \in \mathbb{F}_2^n$. We denote the rotation permutation on n elements by σ (or, σ_n to specify the number n). The cyclic group $P = \langle \sigma \rangle$ is called rotation symmetric group.

2. **Distinct binary-cycle permutation(ψ)**: Let n be a positive integer with the unique binary representation as

   $$n \;=\; n_1 + n_2 + \cdots + n_w \text{ where } n_1 = 2^{a_1}, n_2 = 2^{a_2}, \ldots, n_w = 2^{a_w} \qquad (7.1)$$

   and $0 \leq a_1 < a_2 < \cdots < a_w$. A permutation $\psi \in \mathbb{S}_n$ is called a distinct binary-cycle permutation if its disjoint cycle form contains cycles of length $n_1, n_2, \ldots, n_w$. We name the permutation by distinct binary-cycle permutation as the length of each cycle is a power of 2 and the lengths of cycles are distinct. We denote the permutation on n elements by ψ (or, ψ_n to specify the number n). Without loss of generality, we consider

$$\begin{aligned}\psi &= (x_1, x_2, \ldots, x_{n_1})(x_{n_1+1}, x_{n_1+2}, \ldots, x_{n_1+n_2}) \cdots (x_{n-n_w+1}, x_{n-n_w+2}, \ldots, x_n) \\ &= \sigma_{n_1}\sigma_{n_2}\cdots\sigma_{n_w}. \end{aligned} \tag{7.2}$$

Note that, $\sigma = \psi$ when $n = 2^m$ is a power of 2. For $\pi \in \mathbb{S}_n$, we define π symmetric Boolean functions and 2-π symmetric Boolean functions as follows.

Definition 7.1.1 (π Symmetric Boolean function (πS))**.** Let $\pi \in \mathbb{S}_n$ be a permutation on n elements with order $o(\pi)$. A Boolean function f is π symmetric (in short, πS) if and only if for any $(x_1, x_2, \ldots, x_n) \in \mathbb{F}_2^n$,

$$f(\pi^k(x_1, x_2, \ldots, x_n)) = f(x_1, x_2, \ldots, x_n) \text{ for every } 1 \le k \le o(\pi)$$

where $\pi^k = \pi \circ \pi^{k-1}$ for $k > 1$ and $\pi^1 = \pi$. Therefore, πS Boolean functions have the same truth value for all vectors in every orbit obtained by the action of permutation group $P = \langle\pi\rangle$ on $\mathbb{F}_2^n$.

For $\pi = \sigma$, the σS Boolean functions are known as Rotation Symmetric Boolean functions(in short, RotS). The σS Boolean functions are very well studied functions in the literature [83, 24, 96]. However, no study on ψS Boolean functions is available in the literature. When n is power of 2, the class of σS Boolean functions and the class of ψS Boolean functions are same.

Definition 7.1.2 (2-π Symmetric Boolean function(2-πS))**.** Let $\pi \in \mathbb{S}_n$ be a permutation on n elements and $\mathcal{O}$ be the set of all orbits due to the group action of $P = \langle\pi\rangle$ on $\mathbb{F}_2^n$. A Boolean function f is 2-π symmetric (in short, 2-πS) if and only if for every orbit $\mathsf{O} \in \mathcal{O}$ with a fixed representative element ν_{O},

$$f(\pi^{2i}(\nu_{\mathsf{O}})) = f(\nu_{\mathsf{O}}); \quad f(\pi^{2i+1}(\nu_{\mathsf{O}})) = f(\nu_{\mathsf{O}}) + 1 \text{ for every } 0 \le i < \lfloor\frac{|\mathsf{O}|}{2}\rfloor.$$

Note that, if $|\mathsf{O}|$ is odd then $f(\pi^{|\mathsf{O}|-1}(\nu_{\mathsf{O}}))$ can be any value from $\mathbb{F}_2$.

Therefore, 2-πS Boolean functions have the alternative truth value for the lexicographphically ordered vectors in every orbit obtained by the action of permutation group $P = \langle \pi \rangle$ on $\mathbb{F}_2^n$. As example, a 2-RotS Boolean function (i.e., 2-σS) on $n = 5$ satisfies

$$f(00001) = f(00100) = f(10000)$$

and

$$f(00010) = f(01000) = 1 + f(00001)$$

for the orbit $\{00001, 00010, \ldots, 10000\}$ with representative 00001.

A construction of a class of 2-σS WPB Boolean functions, when n is a power of 2, is presented by Liu and Mesnager [66] as follows.

Proposition 7.1.1. *[66] For a Boolean function $f \in \mathcal{B}_n$ with n is power of 2, if $f(x^2) = f(x) + 1$ holds for all $x \in \mathbb{F}_{2^n} \setminus \{0, 1\}$, then f is WPB.*

The Boolean function proposed in Proposition 7.1.1 is 2-σS. Since, n is the power of 2 in the construction, the cardinality of all orbits in $\mathbb{F}_{2^n} \setminus \{0, 1\}$ are even. Therefore, $f(x^2) = f(x) + 1, x \in \mathbb{F}_{2^n} \setminus \{0, 1\}$ is well defined and hence, the truth value 1 and 0 can be assigned alternatively to the half of the vectors in the each orbit. This can not be assigned when n is not a power of 2 as there are some orbits with cardinality odd and hence the $f(x^2) = f(x)+1$ can not be defined. However, we propose a generalization of this concept to construct WAPB Boolean function on any n where n is a natural number in Section 7.3. When n is power of 2, the class of 2-σS Boolean functions and the class of 2-ψS Boolean functions are the same.

Moreover, in the specific case of Proposition 7.1.1, the function is also σ^2-symmetric (σ^2S), as it takes the same value on each orbit under σ^2. This property, however, is only achievable when n is a power of 2.

7.2 Construction of WAPB Boolean functions using Group action

In this section we present a construction of 2-πS WAPB Boolean functions using the group action of a cyclic permutation group $P = \langle \pi \rangle$. Let $P = \langle \pi \rangle$ be a cyclic subgroup of the symmetric group $\mathbb{S}_n$ on n elements. Let the group action of P on $\mathbb{F}_2^n$ partitions the set into g_n number of orbits. The orbit generated by $x \in \mathbb{F}_2^n$ is denoted as $O_\pi(x) = \{\pi^i(x) : 0 \leq i < o(\pi)\}$. Since $\mathsf{w_H}(\pi^i(x)) = \mathsf{w_H}(x)$ for $0 \leq i < o(\pi)$, the group action P splits each $\mathsf{E}_{k,n}$ into orbits and let $g_{k,n}$ be the number of orbits in $\mathsf{E}_{k,n}$. Denote $\nu_{k,n,i}$ be the orbit representative of i-th orbit in $\mathsf{E}_{k,n}$ with some ordering. A construction of 2-πS WAPB Boolean functions is presented in Construction 2.

Construction 2 Construction of a 2-πS WAPB Boolean function

```
Require: $\pi \in \mathbb{S}_n$
Ensure: A 2-$\pi$S WAPB Boolean function $f_\pi \in \mathcal{B}_n$
Initiate supp$(f_\pi) = \emptyset$
$t = 0$
for $k \leftarrow 0$ to $n$ do
    for $i \leftarrow 1$ to $g_{k,n}$ do
        $u = \nu_{k,n,i}$; $l = |O_\pi(u)|$
        if $l$ is even then
            for $j \leftarrow 1$ to $\frac{l}{2}$ do
                supp$(f_\pi)$.append$(u)$
                $u \leftarrow \pi \circ \pi(u)$
            end for
        else
            $u = \pi^t(u)$
            for $j \leftarrow 1$ to $\lceil \frac{l-t}{2} \rceil$ do
                supp$(f_\pi)$.append$(u)$
                $u \leftarrow \pi \circ \pi(u)$
            end for
            Update $t \leftarrow 1 - t$
        end if
    end for
end for
return $f_\pi$
```

Construction 2 ensures a balanced WAPB Boolean function. The binary variable t indicates whether the partially constructed is balanced (when $t = 0$) or has an extra 1 (when $t = 1$) during each iteration of orbits.

Example 7.2.1. Consider $n = 5$ and the permutation $\pi = \sigma$ is the rotation permutation. Then considering the orbits with representatives

$$00000, 00001, 00011, 00101, 00111, 01011, 01111, 11111$$

we have the resultant function $f_\sigma \in \mathcal{B}_5$ of Construction 2 as $\mathsf{supp}(f_\sigma) = \{00000, 00010, 01000, 00011, 01100, 10001, 01010, 01001, 00111, 11100, 10011, 10110, 11010, 01111, 11101, 10111\}$. Hence, f_σ is a 2-RotS (*i.e.*, 2-σS) WAPB Boolean function.

Given a permutation $\pi \in \mathbb{S}_n$, let $\mathcal{O}_e$ and $\mathcal{O}_o$ denote the set of orbits of even cardinality and odd cardinality of the group action $P = \langle \pi \rangle$ on $\mathbb{F}_2^n$, respectively. Furthermore, let $\mathcal{O}_k$ denote the set of orbits of cardinality k for $k \in [0, n]$. Here, $\mathcal{O}_1$ is the set of orbits of cardinality 1. Since every orbit in $\mathcal{O}_1$ contains exactly one element, abusing the notation, we also denote $\mathcal{O}_1 = \{x \in \mathbb{F}_2^n \,|\, x = \pi(x)\}$ (i.e., the set of vectors in $\mathbb{F}_2^n$ of orbit length 1).

Theorem 7.2.1. *For $\pi \in \mathbb{S}_n$ where $n \in \mathbb{Z}^+$, let $\mathcal{O}_1 = \{x \in \mathbb{F}_2^n : \pi(x) = x\}$. Then, for $c \in \mathbb{F}_2^n$ and $k \in [0, n]$,*

1. $|\{x \in \mathbb{F}_2^n \setminus \mathcal{O}_1 : c \cdot (x + \pi(x)) = 1\}| = \begin{cases} 2^{n-1} & \text{if } c \in \mathbb{F}_2^n \setminus \mathcal{O}_1 \\ 0 & \text{if } c \in \mathcal{O}_1. \end{cases}$
2. $|\{x \in \mathsf{E}_{k,n} \setminus \mathcal{O}_1 : c \cdot (x + \pi(x)) = 1\}| = \frac{1}{2}(|\mathsf{E}_{k,n}| - \mathsf{K}_k(l, n))$, *where* $l = \mathsf{w_H}(c + \pi^{-1}(c))$.

Proof. Now for any $x = (x_1, x_2, \ldots, x_n), c = (c_1, c_2, \ldots, c_n) \in \mathbb{F}_2^n$,

$$c \cdot (x + \pi(x)) = c \cdot x + c \cdot \pi(x) = c \cdot x + \pi^{-1}(c) \cdot x = (c + \pi^{-1}(c)) \cdot x. \tag{7.3}$$

Therefore, $c \cdot (x+\pi(x))$ is a linear Boolean function on n variables. Then $c \cdot (x+\pi(x))$ is the zero Boolean function if and only if $c = \pi^{-1}(c)$ i.e., $c \in \mathcal{O}_1$. Hence,

$$|\{x \in \mathbb{F}_2^n : c \cdot (x+\pi(x)) = 1\}| = \mathsf{w_H}(c \cdot (x+\pi(x))) = \begin{cases} 2^{n-1} & \text{if } c \in \mathbb{F}_2^n \setminus \mathcal{O}_1 \\ 0 & \text{if } c \in \mathcal{O}_1. \end{cases} \tag{7.4}$$

Further, if $x \in \mathcal{O}_1$, then $\pi(x) = x$ and that implies $c \cdot (x+\pi(x)) = 0$. Hence,

$$\begin{aligned} & |\{x \in \mathcal{O}_1 : c \cdot (x+\pi(x)) = 0\}| = |\mathcal{O}_1| \\ \implies & |\{x \in \mathcal{O}_1 : c \cdot (x+\pi(x)) = 1\}| = 0. \end{aligned} \tag{7.5}$$

Now, combining Equation 7.4 and Equation 7.5 we have the desired result of Item 1

$$|\{x \in \mathbb{F}_2^n \setminus \mathcal{O}_1 : c \cdot (x+\pi(x)) = 1\}| = \begin{cases} 2^{n-1} & \text{if } c \in \mathbb{F}_2^n \setminus \mathcal{O}_1 \\ 0 & \text{if } c \in \mathcal{O}_1. \end{cases}$$

Moreover, as $c \cdot (x+\pi(x)) = (c+\pi^{-1}(c)) \cdot x$ is linear in $\mathsf{E}_{k,n}$, using Proposition 4.2.2, we have

$$|\{x \in \mathsf{E}_{k,n} : c \cdot (x+\pi(x)) = 1\}| = \mathsf{w}_{k,n}((c+\pi^{-1}(c)) \cdot x) = \frac{1}{2}(|\mathsf{E}_{k,n}| - \mathsf{K}_k(l,n))$$

where $l = \mathsf{w_H}(c+\pi^{-1}(c))$. If $x \in \mathcal{O}_1$ then $c \cdot (x+\pi(x)) = 0$.
Hence, $|\{x \in \mathsf{E}_{k,n} \setminus \mathcal{O}_1 : c \cdot (x+\pi(x)) = 1\}| = \frac{1}{2}(|\mathsf{E}_{k,n}| - \mathsf{K}_k(l,n))$. □

7.2.1 Nonlinearity and k-Weightwise Nonlinearity for any $\pi \in \mathbb{S}_n$

Theorem 7.2.2. *Let n be a positive integer and $\pi \in \mathbb{S}_n$. Then $\mathsf{NL}(f_\pi) \geq 2^{n-2} - \frac{|\mathcal{O}_o|}{2}$.*

Proof. Here $\mathcal{O}_e, \mathcal{O}_o$ and $\mathcal{O}_1$ are the set of orbits of even cardinalty, odd cardinality and single elements of the group action $G = \langle \pi \rangle$ on $\mathbb{F}_2^n$ respectively. Then the Walsh spectrum of f_π at $a \in \mathbb{F}_2^n$ is as follows.

$$\begin{aligned} W_{f_\pi}(a) &= \sum_{x \in \mathbb{F}_2^n} (-1)^{f_\pi(x)+a\cdot x} = \sum_{\mathsf{O} \in \mathcal{O}} \sum_{x \in \mathsf{O}} (-1)^{f_\pi(x)+a\cdot x} && (7.6) \\ &= \sum_{\mathsf{O} \in \mathcal{O}_e} \sum_{x \in \mathsf{O}} (-1)^{f_\pi(x)+a\cdot x} + \sum_{\mathsf{O} \in \mathcal{O}_o} \sum_{x \in \mathsf{O}} (-1)^{f_\pi(x)+a\cdot x}. && (7.7) \end{aligned}$$

$$\begin{aligned}\sum_{\mathsf{O}\in\mathcal{O}_e}\sum_{x\in\mathsf{O}}(-1)^{f_\pi(x)+a\cdot x} &= \frac{1}{2}\left[\sum_{\mathsf{O}\in\mathcal{O}_e}\sum_{x\in\mathsf{O}}(-1)^{f_\pi(x)+a\cdot x}+\sum_{\mathsf{O}\in\mathcal{O}_e}\sum_{x\in\mathsf{O}}(-1)^{f_\pi(\pi(x))+a\cdot\pi(x)}\right]\\ &= \frac{1}{2}\left[\sum_{\mathsf{O}\in\mathcal{O}_e}\sum_{x\in\mathsf{O}}(-1)^{f_\pi(x)+a\cdot x}+(-1)^{f_\pi(\pi(x))+a\cdot\pi(x)}\right]\\ &= \frac{1}{2}\left[\sum_{\mathsf{O}\in\mathcal{O}_e}\sum_{x\in\mathsf{O}}(-1)^{f_\pi(x)}\left((-1)^{a\cdot x}-(-1)^{a\cdot\pi(x)}\right)\right]\\ &\qquad\qquad(\text{ as } f_\pi(\pi(x))=1+f_\pi(x)).\end{aligned}$$

There are some vectors x in even orbits such that $((-1)^{a\cdot x}-(-1)^{a\cdot\pi(x)})=0$ i.e., $a\cdot(x+\pi(x))=0$. As these vectors contributes 0 to the sum, we now remove them in the equation. Hence, we have

$$\begin{aligned}\sum_{\mathsf{O}\in\mathcal{O}_e}\sum_{x\in\mathsf{O}}(-1)^{f_\pi(x)+a\cdot x} &= \frac{1}{2}\left[\sum_{\mathsf{O}\in\mathcal{O}_e}\sum_{x\in\mathsf{O}:a\cdot(x+\pi(x))=1}(-1)^{f_\pi(x)}((-1)^{a\cdot x}-(-1)^{a\cdot\pi(x)})\right]\\ &= \frac{1}{2}\left[\sum_{\mathsf{O}\in\mathcal{O}_e}\sum_{x\in\mathsf{O}:a\cdot(x+\pi(x))=1}2\times(-1)^{f_\pi(x)+a\cdot x}\right] \qquad (7.8)\\ &= \sum_{\mathsf{O}\in\mathcal{O}_e}\sum_{x\in\mathsf{O}:a\cdot(x+\pi(x))=1}(-1)^{f_\pi(x)+a\cdot x}. \qquad (7.9)\end{aligned}$$

Similarly, $\sum_{\mathsf{O}\in\mathcal{O}_o}\sum_{x\in\mathsf{O}}(-1)^{f_\pi(x)+a\cdot x}$

$$\begin{aligned}&= \sum_{\mathsf{O}\in\mathcal{O}_o\setminus\mathcal{O}_1}\sum_{x\in\mathsf{O}}(-1)^{f_\pi(x)+a\cdot x}+\sum_{\mathsf{O}\in\mathcal{O}_1}\sum_{x\in\mathsf{O}}(-1)^{f_\pi(x)+a\cdot x}\\ &= \sum_{\mathsf{O}\in\mathcal{O}_o\setminus\mathcal{O}_1}\sum_{i=0}^{|O|-1}(-1)^{f_\pi(\pi^i(\nu_{\mathsf{O}}))+a\cdot\pi^i(\nu_{\mathsf{O}})}+\sum_{\mathsf{O}\in\mathcal{O}_1}(-1)^{f_\pi(\nu_{\mathsf{O}})+a\cdot\nu_{\mathsf{O}}} \qquad (7.10)\end{aligned}$$

where ν_{O} is the orbit representative of the orbit O. Then, for an orbit $\mathsf{O}\in\mathcal{O}_o\setminus\mathcal{O}_1$, if

$f(\pi^{|\mathsf{O}|-1}(\nu_{\mathsf{O}})) = 1 + f(\pi^{|\mathsf{O}|-2}(\nu_{\mathsf{O}})) = f(\nu_{\mathsf{O}})$, we have

$$\begin{aligned}
&\sum_{i=0}^{|O|-1}(-1)^{f_\pi(\pi^i(\nu_{\mathsf{O}}))+a\cdot\pi^i(\nu_{\mathsf{O}})}\\
&= \frac{1}{2}\left(\sum_{i=0}^{|O|-1}(-1)^{f_\pi(\pi^i(\nu_{\mathsf{O}}))+a\cdot\pi^i(\nu_{\mathsf{O}})} + \sum_{i=0}^{|O|-1}(-1)^{f_\pi(\pi^i(\nu_{\mathsf{O}}))+a\cdot\pi^i(\nu_{\mathsf{O}})}\right)\\
&= \frac{1}{2}\left(\sum_{i=0}^{|O|-1}(-1)^{f_\pi(\pi^i(\nu_{\mathsf{O}}))+a\cdot\pi^i(\nu_{\mathsf{O}})} + \sum_{i=1}^{|O|-1}(-1)^{f_\pi(\pi^i(\nu_{\mathsf{O}}))+a\cdot\pi^i(\nu_{\mathsf{O}})}\right) + \frac{(-1)^{f_\pi(\nu_{\mathsf{O}})+a\cdot\nu_{\mathsf{O}}}}{2}\\
&= \frac{1}{2}\left(\sum_{i=0}^{|O|-1}(-1)^{f_\pi(\pi^i(\nu_{\mathsf{O}}))+a\cdot\pi^i(\nu_{\mathsf{O}})} + \sum_{i=1}^{|O|-1}(-1)^{f_\pi(\pi^i(\nu_{\mathsf{O}}))+a\cdot\pi^i(\nu_{\mathsf{O}})} + (-1)^{1+f_\pi(\nu_{\mathsf{O}})+a\cdot\nu_{\mathsf{O}}}\right)\\
&\qquad\qquad + \frac{(-1)^{f_\pi(\nu_{\mathsf{O}})+a\cdot\nu_{\mathsf{O}}} - (-1)^{1+f_\pi(\nu_{\mathsf{O}})+a\cdot\nu_{\mathsf{O}}}}{2}\\
&= \frac{1}{2}\left(\sum_{i=0}^{|O|-1}(-1)^{f_\pi(\pi^i(\nu_{\mathsf{O}}))}\left((-1)^{a\cdot\pi^i(\nu_{\mathsf{O}})} - (-1)^{a\cdot\pi^{i+1}(\nu_{\mathsf{O}})}\right)\right) + (-1)^{f_\pi(\nu_{\mathsf{O}})+a\cdot\nu_{\mathsf{O}}}. \qquad (7.11)
\end{aligned}$$

If $f(\pi^{|\mathsf{O}|-1}(\nu_{\mathsf{O}})) = f(\pi^{|\mathsf{O}|-2}(\nu_{\mathsf{O}})) = 1 + f(\nu_{\mathsf{O}})$, consider $\hat{\nu_{\mathsf{O}}} = \pi^{-1}(\nu_{\mathsf{O}})$ as the representative element. We have now the earlier situation with the representative element $\hat{\nu_{\mathsf{O}}}$ i.e., $f(\pi^{|\mathsf{O}|-1}(\hat{\nu_{\mathsf{O}}})) = 1 + f(\pi^{|\mathsf{O}|-2}(\hat{\nu_{\mathsf{O}}})) = f(\hat{\nu_{\mathsf{O}}})$. Hence we will have Equation 7.11 for the representative element $\hat{\nu_{\mathsf{O}}}$. Hence, Equation 7.10 becomes

$$\begin{aligned}
&\sum_{\mathsf{O}\in\mathcal{O}_o}\sum_{x\in\mathsf{O}}(-1)^{f_\pi(x)+a\cdot x}\\
&= \frac{1}{2}\sum_{\mathsf{O}\in\mathcal{O}_o\setminus\mathcal{O}_1}\left(\sum_{i=0}^{|O|-1}(-1)^{f_\pi(\pi^i(\nu_{\mathsf{O}}))}\left((-1)^{a\cdot\pi^i(\nu_{\mathsf{O}})} - (-1)^{a\cdot\pi^{i+1}(\nu_{\mathsf{O}})}\right)\right)\\
&\qquad\qquad + \sum_{\mathsf{O}\in\mathcal{O}_o}(-1)^{f_\pi(\nu_{\mathsf{O}})+a\cdot\nu_{\mathsf{O}}} \qquad (7.12)\\
&= \frac{1}{2}\sum_{\mathsf{O}\in\mathcal{O}_o\setminus\mathcal{O}_1}\sum_{x\in\mathsf{O}}(-1)^{f_\pi(x)}\left((-1)^{a\cdot x} - (-1)^{a\cdot\pi(x)}\right) + \sum_{\mathsf{O}\in\mathcal{O}_o}(-1)^{f_\pi(\nu_{\mathsf{O}})+a\cdot\nu_{\mathsf{O}}}\\
&= \frac{1}{2}\sum_{\mathsf{O}\in\mathcal{O}_o\setminus\mathcal{O}_1}\sum_{x\in\mathsf{O}:a(x+\pi(x))=1} 2\times(-1)^{f_\pi(x)+a\cdot x} + \sum_{\mathsf{O}\in\mathcal{O}_o}(-1)^{f_\pi(\nu_{\mathsf{O}})+a\cdot\nu_{\mathsf{O}}}\\
&= \sum_{\mathsf{O}\in\mathcal{O}_o\setminus\mathcal{O}_1}\sum_{x\in\mathsf{O}:a(x+\pi(x))=1}(-1)^{f_\pi(x)+a\cdot x} + \sum_{\mathsf{O}\in\mathcal{O}_o}(-1)^{f_\pi(\nu_{\mathsf{O}})+a\cdot\nu_{\mathsf{O}}}. \qquad (7.13)
\end{aligned}$$

Now substituting the values in Equation 7.8 and Equation 7.12 in Equation 7.6, we have:

$$\begin{aligned}
W_{f_\pi}(a) &= \sum_{\mathsf{O}\in\mathcal{O}\setminus\mathcal{O}_1} \sum_{x\in\mathsf{O}:a(x+\pi(x))=1} (-1)^{f_\pi(x)+a\cdot x} + \sum_{\mathsf{O}\in\mathcal{O}_o} (-1)^{f_\pi(\nu_\mathsf{O})+a\cdot\nu_\mathsf{O}} \\
\implies |W_{f_\pi}(a)| &\le |\sum_{\mathsf{O}\in\mathcal{O}\setminus\mathcal{O}_1} \sum_{x\in\mathsf{O}:a(x+\pi(x))=1} (-1)^{f_\pi(x)+a\cdot x}| + |\sum_{\mathsf{O}\in\mathcal{O}_o} (-1)^{f_\pi(\nu_\mathsf{O})+a\cdot\nu_\mathsf{O}}| \\
&\le |\sum_{\mathsf{O}\in\mathcal{O}\setminus\mathcal{O}_1} \sum_{x\in\mathsf{O}:a(x+\pi(x))=1} (-1)^{f_\pi(x)+a\cdot x}| + |\mathcal{O}_o| \\
\implies |W_{f_\pi}(a)| &\le \begin{cases} 2^{n-1} + |\mathcal{O}_o| & \text{if } a \in \mathbb{F}_2^n \setminus \mathcal{O}_1 \\ |\mathcal{O}_o| & \text{if } a \in \mathcal{O}_1. \end{cases} \qquad \text{(from Theorem 7.2.1).}
\end{aligned}$$

Hence, the nonlinearity of f_π satisfies

$$\begin{aligned}
\mathsf{NL}(f_\pi) &= 2^{n-1} - \frac{1}{2}\max_{a\in\mathbb{F}_2^n} |W_{f_\pi}(a)| \ \ge\ 2^{n-1} - \frac{1}{2}\max_{a\in\mathbb{F}_2^n}\{2^{n-1} + |\mathcal{O}_o|, |\mathcal{O}_o|\} \\
&\qquad\qquad\qquad\qquad\qquad = 2^{n-1} - 2^{n-2} - \frac{|\mathcal{O}_o|}{2} \\
\implies \mathsf{NL}(f_\pi) &\ge 2^{n-2} - \frac{|\mathcal{O}_o|}{2}.
\end{aligned}$$

□

For a permutation $\pi \in \mathbb{S}_n$, we denote $\mathcal{O}_{e,k}, \mathcal{O}_{o,k}$ and $\mathcal{O}_{1,k}$ by the sets of all orbits of cardinality even, odd and one of the group action $P = \langle\pi\rangle$ on $\mathsf{E}_{k,n}$, respectively.

Theorem 7.2.3. *Let $\pi \in \mathbb{S}_n$ for some $n \in \mathbb{Z}^+$. Then*

$$\mathsf{NL}_k(f_\pi) \ge \frac{1}{4}\left(\binom{n}{k} - 2|\mathcal{O}_{o,k}| + \min_{\substack{2\le l\le n \\ l \text{ even}}} \mathsf{K}_k(l,n)\right).$$

Proof. The restricted Walsh spectrum of f_π at $a \in \mathbb{F}_2^n$ is

$$W_{f_\pi,k}(a) = \sum_{x\in\mathsf{E}_{k,n}} (-1)^{f_\pi(x)+a\cdot x} = \sum_{\mathsf{O}\in\mathcal{O}_{e,k}} \sum_{x\in\mathsf{O}} (-1)^{f_\pi(x)+a\cdot x} \tag{7.14}$$

$$+ \sum_{\mathsf{O}\in\mathcal{O}_{o,k}} \sum_{x\in\mathsf{O}} (-1)^{f_\pi(x)+a\cdot x}. \tag{7.15}$$

Following a similar process as in the proof of Theorem 7.2.2, we have

$$\sum_{\mathsf{O}\in\mathcal{O}_{e,k}} \sum_{x\in\mathsf{O}} (-1)^{f_\pi(x)+a\cdot x} = \sum_{\mathsf{O}\in\mathcal{O}_{e,k}} \sum_{\substack{x\in\mathsf{O} \\ a\cdot(x+\pi(x))=1}} (-1)^{f_\pi(x)+a\cdot x} \tag{7.16}$$

and

$$\sum_{\mathsf{O}\in\mathcal{O}_{o,k}}\sum_{x\in\mathsf{O}}(-1)^{f_\pi(x)+a\cdot x} = \sum_{\mathsf{O}\in\mathcal{O}_{o,k}\setminus\mathcal{O}_{1,k}}\ \sum_{\substack{x\in\mathsf{O}\\ a\cdot(x+\pi(x))=1}} (-1)^{f_\pi(x)+a\cdot x} + \sum_{\mathsf{O}\in\mathcal{O}_{o,k}} (-1)^{f_\pi(\nu_{\mathsf{O}})+a\cdot\nu_{\mathsf{O}}}, \tag{7.17}$$

where ν_{O} is the orbit representative of the orbit O. Now substituting the values in Equation 7.16 and Equation 7.17 in Equation 7.14, we have

$$\begin{aligned} W_{f_\pi,k}(a) &= \sum_{\mathsf{O}\in\mathcal{O}_{e,k}}\ \sum_{\substack{x\in\mathsf{O}\\ a\cdot(x+\pi(x))=1}} (-1)^{f_\pi(x)+a\cdot x} + \sum_{\mathsf{O}\in\mathcal{O}_{o,k}\setminus\mathcal{O}_{1,k}}\ \sum_{\substack{x\in\mathsf{O}\\ a\cdot(x+\pi(x))=1}} (-1)^{f_\pi(x)+a\cdot x} \\ &\quad + \sum_{\mathsf{O}\in\mathcal{O}_{o,k}} (-1)^{f_\pi(\nu_{\mathsf{O}})+a\cdot\nu_{\mathsf{O}}} \\ &= \sum_{\mathsf{O}\in\mathcal{O}_{k}\setminus\mathcal{O}_{1,k}}\ \sum_{\substack{x\in\mathsf{O}\\ a\cdot(x+\pi(x))=1}} (-1)^{f_\pi(x)+a\cdot x} + \sum_{\mathsf{O}\in\mathcal{O}_{o,k}} (-1)^{f_\pi(\nu_{\mathsf{O}})+a\cdot\nu_{\mathsf{O}}} \\ \implies |W_{f_\pi,k}(a)| &\leq |\sum_{\mathsf{O}\in\mathcal{O}_{k}\setminus\mathcal{O}_{1,k}}\ \sum_{\substack{x\in\mathsf{O}\\ a\cdot(x+\pi(x))=1}} (-1)^{f_\pi(x)+a\cdot x}| + |\sum_{\mathsf{O}\in\mathcal{O}_{o,k}} (-1)^{f_\pi(\nu_{\mathsf{O}})+a\cdot\nu_{\mathsf{O}}}| \\ &\leq \frac{1}{2}(|\mathsf{E}_{k,n}| - \mathsf{K}_k(l,n)) + |\mathcal{O}_{o,k}|, \end{aligned}$$

where the last equation comes from Theorem 7.2.1. Hence, the restricted nonlinearity of f_π satisfies

$$\begin{aligned} \mathsf{NL}_k(f_\pi) &= \frac{|\mathsf{E}_{k,n}|}{2} - \frac{1}{2}\max_{a\in\mathbb{F}_2^n}|W_{f_\pi,k}(a)| \\ &\geq \frac{|\mathsf{E}_{k,n}|}{2} - \frac{1}{4}\max_{a\in\mathbb{F}_2^n}(|\mathsf{E}_{k,n}| - \mathsf{K}_k(l,n) + 2|\mathcal{O}_{o,k}|) \\ \implies \mathsf{NL}_k(f_\pi) &\geq \frac{1}{4}\left(|\mathsf{E}_{k,n}| - 2|\mathcal{O}_{o,k}| + \min_{a\in\mathbb{F}_2^n}\mathsf{K}_k(l,n)\right) \\ &= \frac{1}{4}\left(\binom{n}{k} - 2|\mathcal{O}_{o,k}| + \min_{0\leq l\leq n}\mathsf{K}_k(l,n)|\right). \end{aligned}$$

Further, $l = \mathsf{w_H}(a+\pi^{-1}(a))$ is always even as $\mathsf{w_H}(a) = \mathsf{w_H}(\pi^{-1}(a))$. Hence, we have

$$\mathsf{NL}_k(f_\pi) \geq \frac{1}{4}\left(\binom{n}{k} - 2|\mathcal{O}_{o,k}| + \min_{\substack{0\leq l\leq n\\ l \text{ even}}}\mathsf{K}_k(l,n)\right).$$

Using Theorem 4.2.4[Item 4], we have further,

$$\mathsf{NL}_k(f_\pi) \geq \frac{1}{4}\left(\binom{n}{k} - 2|\mathcal{O}_{o,k}| + \min_{\substack{2\leq l\leq n\\ l \text{ even}}} \mathsf{K}_k(l,n)\right).$$

□

In the following sections we study 2-πS WAPB Boolean functions for two different permutations $\psi, \sigma \in \mathbb{S}_n$ for any positive integer n. Here, σ is the rotation permutation and ψ is the distinct binary-cycle permutation as defined in Section 7.3. Both functions generalize the Liu-Mesnager construction [66] of WPB Boolean functions in $n = 2^m$ variables.

7.3 Construction and Study of 2-ψS WAPB Boolean functions

In this section, we present a class of 2-ψS WAPB Boolean function which is a special case of the construction presented in Section 7.2. This construction extends the idea of Liu-Mesnager construction [66] to generate WAPB Boolean functions. As Liu-Mesnager construction outputs a WPB Boolean function, the form of n (the number of variable) needs to be a power of 2. However, in our case, the number of variables n can be any positive integer for generating WAPB Boolean functions.

Let n be a positive integer with binary representation as $n = n_1+n_2+\cdots+n_w$ as defined in Equation 7.1. We denote $\mathsf{w_H}(n) = w$ i.e., the number of 1's in the binary representation of n. For $x = (x_1, x_2, \ldots, x_n)$, we have

$$\psi(x) \quad = \quad (\sigma_{n_1}(x_1,\ldots,x_{n_1}), \sigma_{n_2}(x_{n_1+1},\ldots,x_{n_1+n_2}),\ldots,\sigma_{n_w}(x_{n-n_w+1},\ldots,x_n)) \tag{7.18}$$

where σ_{n_i} is the rotation permutation on n_i elements. Here, $ord(\psi) = 2^{a_w} = n_w$. Hence, the cardinality of orbits obtained due the action of $P = \langle\psi\rangle$ on $\mathbb{F}_2^n$ are of power of 2 i.e., $|O_\psi(x)| = 2^l$ where $0 \leq l \leq a_w$ for $x \in \mathbb{F}_2^n$. Hence, there are some orbits of cardinality 1 and the remainings are of even cardinality.

Lemma 7.3.1. *Let n be a positive integer and $\psi \in \mathbb{S}_n$ as in Equation 7.2. Then there are 2^w orbits of cardinality 1 where $w = \mathsf{w_H}(n)$.*

Proof. For a vector $x \in \mathbb{F}_2^n$ is having an orbit of cardinality 1 i.e., $|O_\psi(x)| = 1$ if and only if the coordinates of x present in the cycles are of same value i.e.,

$$\begin{aligned} &x_1 = x_2 = \ldots = x_{n_1};\\ &x_{n_1+1} = x_{n_1+2} = \ldots = x_{n_1+n_2};\\ &\qquad\vdots\\ &x_{n-n_w+1} = x_{n-n_w+2} = \ldots = x_n. \end{aligned} \tag{7.19}$$

As each partition of coordinates can be either 0 or 1, there are 2^w vectors x in $\mathbb{F}_2^n$ satisfying Equation 7.19 and hence $|O_\psi(x)| = 1$. □

Since every orbit contains the vectors of same weight, we denote the weight of an orbit is the weight of vectors in the orbit i.e., $\mathsf{w_H}(O_\psi(x)) = \mathsf{w_H}(x)$ for $x \in \mathbb{F}_2^n$. Further, for $x = (x_1, x_2, \ldots, x_n), y = (y_1, y_2, \ldots, y_n) \in \mathbb{F}_2^n$, we say y covers x (i.e., $x \preceq y$), if $x_i \leq y_i$ for $1 \leq i \leq n$ i.e., $y_i = 1$ if $x_i = 1$ for $1 \leq i \leq n$. Similarly, given two positive integers n and k with binary representation $n = 2^{a_1} + 2^{a_2} + + \cdots + 2^{a_w}$ and $k = 2^{b_1} + 2^{b_2} + \cdots + 2^{b_t}$, we denote $k \preceq n$ if $\{b_1, b_2, \ldots, b_t\} \subseteq \{a_1, a_2, \ldots, a_w\}$.

Lemma 7.3.2. *Let n be a positive integer and $\psi \in \mathbb{S}_n$ as in Equation 7.2. For $k \in [0, n]$, the number of orbits of weight k and cardinality 1 is 1 if $k \preceq n$, otherwise it is 0.*

Proof. Let $k = 2^{b_1} + 2^{b_2} + \cdots + 2^{b_t}$ where $0 \leq b_1 < b_2 < \cdots < b_t$.

Case I: Let $k \preceq n$ i.e., $\{b_1, b_2, \ldots, b_t\} \subseteq \{a_1, a_2, \ldots, a_w\}$. Since the only way of writing k as sum of powers of 2 is $k = 2^{b_1}+2^{b_2}+\cdots+2^{b_t}$ and satisfying the condition in Equation 7.19, there is only one vector x with $\mathsf{w_H}(x) = k$ and $|O_\psi(x)| = 1$. In this case, the coordinates of x in the partitions of cardinality $2^{b_1}, 2^{b_2}, \ldots, 2^{b_t}$ are having value 1 and other coordinates have value 0.

Case II: Let $k \not\preceq n$, then $\{b_1, b_2, \ldots, b_t\} \not\subseteq \{a_1, a_2, \ldots, a_w\}$. If $\mathsf{w_H}(x) = k$, the nonzero coordinates of x can not be partitioned of (distinct) sizes from the set $\{2^{a_1}, 2^{a_2}, \ldots, 2^{b_w}\}$. As a result, the coordinates of x will not satisfy the Equation 7.19. Hence, $|O_\psi(x)| > 1$. Hence, in this case there is no orbit of weight k and cardinality 1. □

We denote $\mathcal{O}_o$ be the set of orbits of odd cardinality (i.e., here 1) and $\mathcal{O}_e$ be the set of orbits of even cardinality. Since the cardinality of all orbits of carinality odd is 1, abusing the notation, we also denote $\mathcal{O}_o$ as the set of all vectors belonging in the orbits of odd cardinality. Hence from Equation 7.19, $\mathcal{O}_o = \{(x_1, x_2, \cdots, x_n) \in \mathbb{F}_2^n : x_1 = x_2 = \cdots = x_{n_1};\ \ x_{n_1+1} = x_{n_1+2} = \cdots = x_{n_1+n_2};\ \cdots;\ x_{(n-n_w)+1} = x_{(n-n_w)+2} = \cdots = x_n\}$. For example, if $n = 6$, there are there are $2^{\mathsf{w_H}(6)} = 2^2 = 4$ orbits of weight 1 and $\mathcal{O}_o = \{000000, 000011, 111100, 111111\}$.

By choosing such permutation ψ for Construction 2, we have every slice $\mathsf{E}_{k,n}$, $0 \leq k \leq n$, contains at most one orbit of odd cardinality (and i.e., 1). Therefore, it becomes easy to construct 2-ψS WAPB Boolean functions as other orbits are of even cardinality. Hence, we have the following result.

Theorem 7.3.3. *Let n be a positive integer and $\psi \in \mathbb{S}_n$ as in Equation 7.2. For a Boolean function $f_\psi \in \mathcal{B}_n$, if $f_\psi(\psi(x)) = 1 + f_\psi(x)$ holds for all $x \in \mathbb{F}_2^n \setminus \mathcal{O}_o$ where $\mathcal{O}_o$ is the set of vectors whose orbit cardinality is 1, then f_ψ is WAPB.*

Hence, when $n = 2^m$, a power of 2, $\psi = \sigma$ and Construction 2 on input $\psi \in \mathbb{S}_n$ results the 2-RotS WPB Boolean function by Liu and Mesnager [66]. A simplified version of Construction 2 is presented in Construction 7.3 for input ψ.

Construction 3 Construction of 2-ψS WAPB Boolean function using $\psi \in \mathbb{S}_n$

```
Require: $\psi \in \mathbb{S}_n$ as in Equation 7.2
Ensure: A 2-$\psi$S WAPB Boolean function $f_\psi \in \mathcal{B}_n$.
For every orbit O in $\mathbb{F}_2^n$ due to the action of $P = \langle \psi \rangle$, do the following:
if $|\mathsf{O}|$ is even then
    $f$ satisfies $f_\psi(\psi(x)) = 1 + f(x)$ for $x \in \mathsf{O}$
end if
if $|\mathsf{O}| = 1$ then
    assign $f_\psi(x) = 0$ or $1$.
end if
return $f_\psi$
```

7.3.1 Nonlinearity and k-Weightwise Nonlinearity for $2 - \psi$S WAPB Boolean Functions

Theorem 7.3.4. *The number of orbits generated due the action of ψ on $\mathbb{F}_2^n$ is*

$$g_n = \frac{1}{n_w} \sum_{k=1}^{n_w} 2^{\gcd(n_1,k)+\gcd(n_2,k)+\cdots+\gcd(n_w,k)}.$$

Proof. As $ord(\psi) = 2^{a_w} = n_w$, let denote $G =< \psi >= \{\psi_n^1, \psi_n^2, \ldots, \psi_n^{n_w}\}$ where $\psi_n^1 = \psi$ and $\psi_n^i = \psi \circ \psi_n^{i-1}$ for $i \geq 2$. From the disjoint cycle form of ψ as in Equation 7.18, we have

$$\psi(x) = (\sigma_{n_1}(x_1, \ldots, x_{n_1}), \sigma_{n_2}(x_{n_1+1}, \ldots, x_{n_1+n_2}), \ldots, \sigma_{n_w}(x_{n-n_w+1}, \ldots, x_n))$$

where σ_{n_i} is the rotation permutation on n_i elements. Hence, we denote, $\psi_n = (\sigma_{n_1}, \sigma_{n_2}, \ldots, \sigma_{n_w})$ and for positive integers k, we have $\psi_n^k = (\sigma_{n_1}^k, \sigma_{n_2}^k, \ldots, \sigma_{n_w}^k)$.

Now to apply Burnside's lemma, for every $k \in \{1, 2, \ldots, n_w\}$, we need to compute the number of fixed vectors $z \in \mathbb{F}_2^n$ by ψ_n^k i.e., $\psi_n^k(z) = z$. That is, for every $k \in \{1, 2, \ldots, n_w\}$, we need to compute the number of vectors $z \in \mathbb{F}_2^n$ such that $\rho_{n_1}^k(z_1) = z_1, \rho_{n_2}^k(z_2) = z_2, \ldots, \rho_{n_w}^k(z_w) = z_w$ where $z = (z_1, z_2, \ldots, z_w)$ and $z_1 \in \mathbb{F}_2^{n_1}, z_2 \in \mathbb{F}_2^{n_2}, \ldots, z_w \in \mathbb{F}_2^{n_w}$.

Here, the number of permutation cycles in $\sigma_{n_i}^k = \gcd(n_i, k)$ for $1 \leq i \leq w$ and $1 \leq k \leq n_w$ and the length of each permutation cycle in $\sigma_{n_i}^k$ is $\frac{n_i}{\gcd(n_i,k)}$. Therefore, the total number of permutation cycles in ψ^k is

$$\gcd(n_1, k) + \gcd(n_2, k) + \cdots + \gcd(n_w, k).$$

As every permutation cycle fixes all 0's or all 1's, each permutation cycle has two choices. Therefore, ψ^k fixes $2^{\gcd(n_1,k)+\gcd(n_2,k)+\cdots+\gcd(n_w,k)}$ number of $z \in \mathbb{F}_2^n$. Hence, by using the Burnside Lemma, the number of orbits is

$$g_n = \frac{1}{n_w} \sum_{\pi \in G} |fix_{\mathbb{F}_2^n}(\pi)| = \frac{1}{n_w} \sum_{k=1}^{n_w} 2^{\gcd(n_1,k)+\gcd(n_2,k)+\cdots+\gcd(n_w,k)}.$$ □

The representative of each orbit can be be assigned 0 or 1 and accordingly other vectors in the orbit are assigned. Hence, there are 2^{g_n} WAPB 2-ψS Boolean functions in n variables. Further, we can count the number of balanced WAPB 2-ψS Boolean functions in n variables. There are 2^w many orbits of cardinality 1 and remaining $g_n - 2^w$ orbits are having cardinality even. Further, 2^{w-1} orbits from the 2^w orbits of cardinality 1 are to be assigned 1 to make f_ψ balanced. Hence $\binom{2^w}{2^{w-1}} \times 2^{g_n - 2^w}$ balanced WAPB 2-ψS Boolean functions can be generated using Construction 7.3. Now we will study some cryptographic properties of the function $f_\psi \in \mathcal{B}_n$.

Theorem 7.3.5. *Let $\psi \in \mathbb{S}_n$ as in Equation 7.2 with $\mathsf{w_H}(n) = w$. Then $\mathsf{NL}(f_\psi) \geq 2^{n-2} - 2^{w-1}$.*

Proof. As $\mathsf{w_H}(n) = w$, from Lemma 7.3.1 there are 2^w orbits with cardinality 1 and remaining orbits are of even cardinality (i.e., $|\mathsf{O}_o| = |\mathsf{O}_1| = 2^w$). Then from Theorem 7.2.2, the nonlinearity bound of f_ψ is

$$\mathsf{NL}(f_\psi) \geq 2^{n-2} - \frac{|\mathsf{O}_o|}{2} = 2^{n-2} - 2^{w-1}.$$

□

The following corollary presents a nonlinearity bound for the 2-ψS WPB Boolean function which resembles with the WPB Boolean function by Liu and Mesnager [66].

Corollary 7.3.6. *Let* $n = 2^m$ *be a positive integer. Then* $\mathsf{NL}(f_\psi) \geq 2^{n-2} - 1$.

In Table 7.1, we have presented the maximum and minimum nonlinearity among all f_ψ for the number of variables $n = \{4, 5, \ldots, 10\}$ along with the upperbound of balanced Boolean functions and lowerbound of f_ψ as per Theorem 7.3.5. We have searched all such balanced Boolean functions for $n \leq 6$ and from $70 \times 2^{20}, 2 \times 2^{25}, 6 \times 2^{23}, 6 \times 2^{23}$ randomly chosen such Boolean functions for $n = 7, 8, 9, 10$ respectively.

n	4	5	6	7	8	9	10
Number of functions	$2^4 \times \binom{2}{1}$ $= 2^5$	$2^8 \times \binom{4}{2}$ $= 6 \times 2^9$	$2^{18} \times \binom{4}{2}$ $= 6 \times 2^{19}$	$2^{36} \times \binom{8}{4}$ $= 70 \times 2^{37}$	$2^{34} \times \binom{2}{1}$ $= 2^{35}$	$2^{68} \times \binom{4}{2}$ $= 6 \times 2^{69}$	$2^{138} \times \binom{4}{2}$ $= 6 \times 2^{139}$
Max NL by experiment	4	12	26	56	116	238	480
% functions at max NL	100	22.92	0.65	4.3×10^{-3}	4.6×10^{-3}	1.6×10^{-5}	2.4×10^{-3}
Nonlinearity upper bound of balanced functions [93]	4	12	28	58	118	244	494
Min Nonlinearity	4	6	14	28	64	144	328
% functions at min NL	100	4.17	0.26	2.3×10^{-4}	3.3×10^{-3}	3.2×10^{-5}	1.6×10^{-5}
Average NL by experiment	4.0	9.79	21.83	47.35	106.01	220.58	453.49
Nonlinearity lower bound by Theorem 7.3.5	3	6	14	28	63	144	254

Table 7.1: Percentage of $f_\psi \in \mathcal{B}_n$ satisfying lower bound and upper value of $\mathsf{NL}(f_\psi)$.

Now we will study the weightwise nonlinearity of f_ψ.

Theorem 7.3.7. *Let* $n \geq 2$ *be a positive integer and* $\psi \in \mathbb{S}_n$ *as in Equation 7.2. Then*

$$\mathsf{NL}_k(f_\psi) \geq \begin{cases} \frac{1}{4}\left(\binom{n}{k} + \min\limits_{\substack{2\leq l\leq n \\ l \text{ even}}} \mathsf{K}_k(l,n)\right) & \text{if } k \not\preceq n \\ \frac{1}{4}\left(\binom{n}{k} + \min\limits_{\substack{2\leq l\leq n \\ l \text{ even}}} \mathsf{K}_k(l,n) - 2\right) & \text{if } k \preceq n. \end{cases}$$

Proof. From Lemma 7.3.2, we have $|\mathcal{O}_{o,k}| = |\mathcal{O}_{1,k}| = 1$ if $k \preceq n$ else it is 0. Hence from Theorem 7.2.3, we have

$$\mathsf{NL}_k(f_\psi) \geq \frac{1}{4}\left(\binom{n}{k} + \min_{\substack{2\leq l\leq n \\ l \text{ even}}} \mathsf{K}_k(l,n)\right) + \varepsilon,$$

where $\varepsilon = 0$ if $k \not\preceq n$ and -2 otherwise. □

We can discard the case $l = n$ from finding minimum in $\mathsf{K}_k(l, n)$ for some cases.

Corollary 7.3.8. *Let $n \geq 2$ be a positive integer and $\psi \in \mathbb{S}_n$ as in Equation 7.2. If k is even or n is odd then:*

$$\mathsf{NL}_k(f_\psi) \geq \frac{1}{4}\left(\binom{n}{k} + \min_{\substack{2\leq l\leq n-1\\ l \text{ even}}} \mathsf{K}_k(l, n)\right) + \varepsilon,$$

where $\varepsilon = 0$ if $k \npreceq n$ and -2 otherwise. Otherwise (i.e., k is odd and n is even), $\mathsf{NL}_k(f_\psi) \geq 0$.

Proof. If n is odd then $\mathsf{K}_k(n, n)$ is not included in the minimum finding as l has to be even. Further, from Theorem 4.2.4[Item 5], if k is even then $\max_{0\leq l\leq n} \mathsf{K}_k(l, n) = \mathsf{K}_k(n, n)$. Hence we are done for the case k is even or n is odd.

If k is odd and n is even, then $\min_{\substack{2\leq l\leq n\\ l \text{ even}}} \mathsf{K}_k(l, n) = \mathsf{K}_k(n, n) = -\binom{n}{k}$. Further, in this case $k \npreceq n$. Hence, $\mathsf{NL}_k(f_\psi) \geq 0$, which is infact always true. □

k n	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
15	24	0	330	0	1215	0	1506	0	500	0	45	0	-	-	-	-	-
16	28	0	443	0	1931	0	3003	0	1502	0	228	0	8	-	-	-	-
17	32	0	580	0	3003	0	5720	0	4004	0	910	0	60	0	-	-	-
18	36	0	750	0	4550	0	10725	0	9724	0	3094	0	340	0	8	-	-
19	40	0	950	0	6650	0	18343	0	21879	0	9282	0	1530	0	76	0	-
20	45	0	1190	0	9524	0	30719	0	43758	0	25194	0	5814	0	484	0	10

Table 7.2: Values of the lower bound of $\mathsf{NL}_k(f_\psi)$ as per Theorem 7.3.7.

Table 7.2 presents the numerical values of the lower bounds of $\mathsf{NL}_k(f_\psi)$ for $n = [15, 20]$ deducted from Theorem 7.3.7. Further, substituting the results of Theorem 4.2.4[Item 1 and Item 2] in Corollary 7.3.7, we have nonlinearity bounds for particular values of k.

Theorem 7.3.9. *1. If n is odd then*

$$\mathsf{NL}_k(f_\psi) \geq \begin{cases} \dfrac{1}{2}\dbinom{n-1}{k-1} & \textit{if } k \in [0, \frac{n-1}{2}],\ \textit{odd and } k \not\preceq n \\ \dfrac{1}{2}\left(\dbinom{n-1}{k-1} - 1\right) & \textit{if } k \in [0, \frac{n-1}{2}],\ \textit{odd and } k \preceq n \\ \dfrac{1}{2}\dbinom{n-1}{k} & \textit{if } k \in [\frac{n+1}{2}, n] \textit{ and } k \not\preceq n \\ \dfrac{1}{2}\left(\dbinom{n-1}{k} - 1\right) & \textit{if } k \in [\frac{n+1}{2}, n] \textit{ and } k \preceq n. \end{cases}$$

2. If n is even and $k \in [\frac{n}{2}+1, n]$ is even then

$$\mathsf{NL}_k(f_\psi) \geq \begin{cases} \dfrac{1}{2}\dbinom{n-1}{k} & \textit{if } k \not\preceq n \\ \dfrac{1}{2}\left(\dbinom{n-1}{k} - 1\right) & \textit{if } k \preceq n. \end{cases}$$

Note that the bounds in Theorem 7.3.9 do not cover the cases when (1) n is odd with $k \in [0, \frac{n-1}{2}]$ and even; and (2) n is even with $k \in [0, \frac{n}{2}]$.

Since $\mathsf{E}_{\lfloor \frac{n}{2} \rfloor, n}$ is the largest slice among all slices $\mathsf{E}_{k,n}, k \in [0, n]$, studying the nonlinearity in this domain is useful for cryptographic purposes. The cipher FLIP also uses the domain $\mathsf{E}_{\frac{n}{2}, n}$ for its design. Using the particular case results of Theorem 4.2.4[Item 1 and Item 2] in Corollary 7.3.7, we have the nonlinearity bounds for the largest slice (*i.e.*, $k = \lfloor \frac{n}{2} \rfloor$) in the following theorem.

Theorem 7.3.10. *1. If $n = 2m+1$ is odd then*

$$\mathsf{NL}_m(f_\psi) \geq \begin{cases} \dfrac{1}{2}\dbinom{n-1}{m-1} & \textit{if } m \not\preceq n \ (\textit{i.e., } n \textit{ is not of the form } 2^t - 1); \\ \dfrac{1}{2}\left(\dbinom{n-1}{m-1} - 1\right) & \textit{if } m \preceq n \ (\textit{i.e., } n \textit{ is of the form } 2^t - 1). \end{cases}$$

2. If $n = 2m$ is even then

(a) $\mathsf{NL}_m(f_\psi) \geq \binom{n-2}{m}$ if m is even.

(b) $\mathsf{NL}_{m-1}(f_\psi) \geq \frac{1}{2}\left(\binom{n-2}{m-1} + \binom{n-2}{m-3}\right)$ if m is odd and $n \geq 10$.

Now, we study the restricted nonlinearity $\mathsf{NL}_2(f_\psi)$. We denote by NL_k^n as

$$\mathsf{NL}_k^n = \min\{\mathsf{NL}_k^n(f_\psi) \mid f_\psi \in \mathcal{B}_n \text{ constructed as in Theorem 7.3.3}\}.$$

Theorem 7.3.11. *Let $n \geq 2$ be an positive integer and $\psi \in \mathbb{S}_n$ as in Equation 7.2. Then*

$$\mathsf{NL}_2(f_\psi) \,\geq\, \lfloor \frac{(n-1)^2}{8} \rfloor.$$

Proof. From Theorem 4.2.4, we have $\min\limits_{0 \leq l \leq n} \mathsf{K}_2(l,n) = -\lfloor \frac{n}{2} \rfloor$. Using it in Theorem 7.3.7 we have,

$$\mathsf{NL}_2(f_\psi) \,\geq\, \begin{cases} \frac{1}{4}\left(\binom{n}{2} - \lfloor \frac{n}{2} \rfloor\right) & \text{if } 2 \not\preceq n \\ \frac{1}{4}\left(\binom{n}{2} - \lfloor \frac{n}{2} \rfloor - 2\right) & \text{if } 2 \preceq n. \end{cases}$$

Therefore, for $2 \not\preceq n$, we have $\mathsf{NL}_2(f_\psi) \geq \begin{cases} \frac{n(n-2)}{8} & \text{if } n \text{ is even } (i.e., n = 4t \text{ form}) \\ \frac{(n-1)^2}{8} & \text{if } n \text{ is odd } (i.e., n = 4t+1 \text{ form}), \end{cases}$

and for $2 \preceq n$, we have $\mathsf{NL}_2(f_\psi) \geq \begin{cases} \frac{n(n-2)}{8} - \frac{1}{2} & \text{if } n \text{ is even } (i.e., n = 4t+2 \text{ form}) \\ \frac{(n-1)^2}{8} - \frac{1}{2} & \text{if } n \text{ is odd } (i.e., n = 4t+3 \text{ form}). \end{cases}$

We can check that for n even, $\frac{n(n-2)}{8}$ is always an integer and for n odd, $\frac{(n-1)^2}{8}$ is an integer if and only if $2 \not\preceq n$. Hence, combining the cases, we have

$$\mathsf{NL}_2(f_\psi) \geq \begin{cases} \frac{n(n-2)}{8} & \text{if } n \text{ is even} \\ \lfloor \frac{(n-1)^2}{8} \rfloor & \text{if } n \text{ is odd.} \end{cases}$$

Further, as $\frac{(n-1)^2}{8} = \frac{n(n-2)}{8} + \frac{1}{8}$, $\lfloor \frac{(n-1)^2}{8} \rfloor = \frac{n(n-2)}{8}$ when n is even. Hence, $\mathsf{NL}_2(f_\psi) \geq \lfloor \frac{(n-1)^2}{8} \rfloor$. □

Theorem 7.3.12. *Let $n \geq 2$ be an positive integer and $\psi \in \mathbb{S}_n$ as in Equation 7.2. Then for $k \in [0,n]$ $\mathsf{NL}_k^n \leq \mathsf{NL}_{2k}^{2n}$. For generalization, $\mathsf{NL}_k^n \leq \mathsf{NL}_{2^ik}^{2^in}$ for $i \geq 0$.*

Proof. We use the technique followed in the proof of [66, Theorem-3.14] to prove it. It can be checked that $\mathsf{NL}_R(f) \leq \mathsf{NL}_E(f)$ if $R \subseteq E$.

Let $n = 2^{a_w} + 2^{a_{w-1}} + \cdots + 2^{a_1}$ for $a_w > a_{w-1} > \cdots > a_1 \geq 0$ as defined in Equation7.1. For $y \in \mathsf{E}_{k,n}$, consider the form $y = (\overline{y}_w, \overline{y}_{w-1}, \ldots, \overline{y}_1)$ where $(\overline{y}_i) = (y_{n_i-1}, y_{n_i-2}, \ldots, y_{n_{i-1}}) \in \mathbb{F}_2^{n_i}$ and $\mathsf{w_H}(y) = k$. Now define a set

$$R_k = \{(\overline{y}_w\overline{y}_w\overline{y}_{w-1}\overline{y}_{w-1}\ldots\overline{y}_1\overline{y}_1) : y = \overline{y}_w\overline{y}_{w-1}\ldots\overline{y}_1 \in \mathbb{F}_2^n \text{ and } \mathsf{w_H}(y) = k\} \subseteq \mathsf{E}_{2k,2n}.$$

It can be checked that for $x = (\overline{y}_w\overline{y}_w\overline{y}_{w-1}\overline{y}_{w-1}\dots\overline{y}_1\overline{y}_1) \in R_k$, the orbit containing x,

$$\mathsf{O}_{\psi_n}(x) = \{(\overline{z}_w\overline{z}_w\overline{z}_{w-1}\overline{z}_{w-1}\dots\overline{z}_1\overline{z}_1) : (\overline{z}_w\overline{z}_{w-1}\dots\overline{z}_1) \in \mathsf{O}_{\psi_{\frac{n}{2}}}(y),\ y = \overline{y}_w\overline{y}_{w-1}\dots\overline{y}_1\}.$$

Then for a WAPB $f \in \mathcal{B}_{2n}$ satisfying Theorem 7.3.3, we have a WAPB $g \in \mathcal{B}_n$ such that $\forall y \in \mathbb{F}_2^n$, $g(y) = f(x)$ for $x \in R$. This implies,

$$\mathsf{NL}_k^n(g) = \mathsf{NL}_{\mathsf{E}_{k,n}}(g) = \mathsf{NL}_{R_k}(f) \le \mathsf{NL}_{\mathsf{E}_{2k,2n}}(f) = \mathsf{NL}_{2k}^{2n}(f).$$

The equality $\mathsf{NL}_{\mathsf{E}_{k,n}}(g) = \mathsf{NL}_{R_k}(f)$ can be proved as follows.

$$\begin{aligned}
\mathsf{NL}_{R_k}(f) &= \frac{|R_k|}{2} - \frac{1}{2}\max_{a\in\mathbb{F}_2^n}\sum_{x\in R_k}(-1)^{f(x)+a\cdot x}\\
&= \frac{|\mathsf{E}_{k,n}|}{2} - \frac{1}{2}\max_{a\in\mathbb{F}_2^n}\sum_{y\in\mathsf{E}_{k,n}}(-1)^{g(y)+a.(\overline{y}_w\overline{y}_w\cdots\overline{y}_1\overline{y}_1)}\\
&= \frac{|\mathsf{E}_{k,n}|}{2} - \frac{1}{2}\max_{a\in\mathbb{F}_2^n}\sum_{y\in\mathsf{E}_{k,n}}(-1)^{g(y)+(a^l+a^r).y}\\
&\qquad\text{where } a^l, a^r \in \mathbb{F}_2^n \text{ with } a = (\overline{a}_w^l\overline{a}_w^r\dots\overline{a}_1^l\overline{a}_1^r)\\
&= \frac{|\mathsf{E}_{k,n}|}{2} - \frac{1}{2}\max_{b\in\mathbb{F}_2^n}\sum_{y\in\mathsf{E}_{k,n}}(-1)^{g(y)+b.y} = \mathsf{NL}_{\mathsf{E}_{k,n}}(g).
\end{aligned}$$

Let the $\mathsf{NL}_{2k}^{2n} = \min_{f\in\mathcal{B}_{2n}\text{WAPB}} \mathsf{NL}_{2k}^{2n}(f) = \mathsf{NL}_{2k}^{2n}(\hat{f})$ for a WAPB Boolean function $\hat{f} \in \mathcal{B}_{2n}$. Then there is a WAPB $\hat{g} \in \mathcal{B}_n$ such that $\mathsf{NL}_k^n(\hat{g}) \le \mathsf{NL}_{2k}^{2n}(\hat{f})$. Hence, $\mathsf{NL}_k^n \le \mathsf{NL}_k^n(\hat{g}) \le \mathsf{NL}_{2k}^{2n}(\hat{f}) = \mathsf{NL}_{2k}^{2n}$. □

Now using Theorem 7.3.11 and Theorem 7.3.12, we have the following Corollary.

Corollary 7.3.13. *Let $n \ge 2$ be an positive integer and $\psi \in \mathbb{S}_n$ as in Equation 7.2. Then*

$$\lfloor\frac{(n-1)^2}{8}\rfloor \le \mathsf{NL}_2^n \le \mathsf{NL}_4^{2n} \le \mathsf{NL}_8^{2^2n} \le \cdots \le \mathsf{NL}_{2^{i+1}}^{2^in} \le \cdots.$$

Moreover, if $n = 2^m$ for $m \ge 1$, $\dfrac{n(n-2)}{8} \le \mathsf{NL}_2^n \le \mathsf{NL}_4^{2n} \le \mathsf{NL}_8^{2^2n} \le \cdots \le \mathsf{NL}_{2^{i+1}}^{2^in} \le \cdots$.

Thus, the above theorem provides a better lower bound for the weightwise nonlinearity $\mathsf{NL}_k(f_\psi)$, as proven in the article [66], recalled in the following proposition.

Proposition 7.3.14. *[66][Theorem 3.14] For any $n = 2^m \geq 8$ and f_ψ be a WPB Boolean function as defined in 7.1.1, then*

$$\mathsf{NL}_{2^i}^{(n)}(f_\psi) \geq \begin{cases} 5, & \textit{if } 1 \leq i \leq m-3, \\ 6, & \textit{if } i = m-2, \\ 19, & \textit{if } i = m-1. \end{cases}$$

7.4 Construction and Study of 2-σS (i.e., 2-RotS) WAPB Boolean functions

In this section, we present a class of 2-σS WAPB Boolean function which is another special case of the construction presented in Section 7.2. Since $\sigma \in \mathbb{S}_n$ is the rotation permutation, these functions are also known as 2-RotS WAPB Boolean functions. When n is a power of 2, the construction matches the Liu-Mesnager construction [66] to generate WPB Boolean functions. Now to study nonlinearity bound in Theorem 7.2.2, we need to count the number of orbits of odd cardinality due to the group action of $C_n = \langle \sigma_n \rangle$ on $\mathbb{F}_2^n$. For a positive integer n, we denote n_0 as the largest odd integer that divides n i.e., $n_0 = \frac{n}{2^t}$ for largest integer t such that $\frac{n}{2^t}$ is an integer.

Proposition 7.4.1 ([96], Theorem 3). *Let $C_n = \langle \sigma_n \rangle$ for $\sigma_n \in \mathbb{S}_n$ be the cyclic group of rotation permutations acting on $\mathbb{F}_2^n$. Then the number of orbits induced on $\mathbb{F}_2^n$ is given by $g_n = \frac{1}{n} \sum\limits_{t|n} \phi(t) 2^{\frac{n}{t}}$, where $\phi(t)$ is Euler's phi-function.*

Lemma 7.4.2. *Let $d \mid n$ for positive integers d and n. Let $C_n = \langle \sigma_n \rangle$ for $\sigma_n \in \mathbb{S}_n$ and $C_d = \langle \sigma_d \rangle$ for $\sigma_d \in \mathbb{S}_d$. For $v \in \mathbb{F}_2^d$, $|\mathsf{O}_{\sigma_d}(v)| = |\mathsf{O}_{\sigma_n}(\hat{v})|$ where $\hat{v} = (v, v, \ldots, v) \in \mathbb{F}_2^n$.*

Proof. For $v \in \mathbb{F}_2^d$, we can check that $\sigma_n(\hat{v}) = \sigma_n(v, v, \ldots, v) = (\sigma_d(v), \sigma_d(v), \ldots, \sigma_d(v))$.

If $|\mathsf{O}_{\sigma_d}(v)| = k$ then $\sigma_d^k(v) = v \implies (\sigma_d^k(v), \sigma_d^k(v), \ldots, \sigma_d^k(v)) = (v, v, \ldots, v)$, that is, $\sigma_n^k(\hat{v}) = \hat{v} \implies |\mathsf{O}_{\sigma_n}(\hat{v})| \mid k$. Further, if $|\mathsf{O}_{\sigma_n}(\hat{v})| = l$ then $\sigma_n^l(\hat{v}) = \hat{v}$, that is,

$(\sigma_d^l(v), \sigma_d^l(v), \ldots, \sigma_d^l(v)) = (v, v, \ldots, v) \implies \sigma_d^l(v) = v \implies |\mathsf{O}_{\sigma_d}(v)| \mid l$. Therefore, $|\mathsf{O}_{\sigma_d}(v)| = |\mathsf{O}_{\sigma_n}(\hat{v})|$ for every $v \in \mathbb{F}_2^d$. □

Theorem 7.4.3. *For $\sigma_n \in \mathbb{S}_n$, the number of orbits of odd cardinality due to the action of $C_n = \langle \sigma_n \rangle$ on $\mathbb{F}_2^n$ is given by $g_{n,odd} = g_{n_0}$.*

Proof. For $d \mid n$, it can be easily checked that if $|\mathsf{O}_{\sigma_n}(x)| = d$ for an $x \in \mathbb{F}_2^n$ then there is a $v \in \mathbb{F}_2^d$ such that such that $x = (v, v, \ldots, v)$ and $|\mathsf{O}_{\sigma_d}(v)| = d$ in $\mathbb{F}_2^d$. Every odd divisor d of n is also a divisor of n_0 and viceversa. Hence, for $d \mid n$ and d odd, $|\mathsf{O}_{\sigma_n}(x)| = d$ for an $x \in \mathbb{F}_2^n$ if and only if there is a $v \in \mathbb{F}_2^d$ such that $x = (v, v, \ldots, v)$ and $|\mathsf{O}_{\sigma_d}(v)| = d$ if and only if $|\mathsf{O}_{\sigma_{n_0}}(y)| = d$ where $y = (v, v, \ldots, v) \in \mathbb{F}_2^{n_0}$. Therefore, $g_{n,odd} = g_{n_0}$. □

7.4.1 Study of Nonliearity and k-weightwise Nonlinearity of $2-\sigma$S WAPB Boolean Functions

Theorem 7.4.4. *Let $\sigma_n \in \mathbb{S}_n$ for a positive integer n. Then, $\mathsf{NL}(f_{\sigma_n}) \geq 2^{n-2} - \frac{g_{n_0}}{2}$.*

Proof. From Theorem 7.2.2, we have

$$\mathsf{NL}(f_{\sigma_n}) \geq 2^{n-2} - \frac{|\mathsf{O}_o|}{2} = 2^{n-2} - \frac{g_{n,odd}}{2} = 2^{n-2} - \frac{g_{n_0}}{2}.$$

□

As g_n is an even integer [21] (Proposition 3.1), the bound is always an integer. When $n = 2^m$ is a power of 2, the nonlinearity bound becomes $\mathsf{NL}(f_{\sigma_n}) \geq 2^{n-2} - 1$ as $g_{n_0} = g_1 = 2$. Since $f_{\sigma_n} = f_{\psi_n}$ when n is a power of 2, the nonlinearity bound $\mathsf{NL}(f_{\sigma_n})$ matches with the result in Corollary 7.3.6. In Table 7.3, we give the maximum and minimum nonlinearity among all f_σ for the number of variables $n = \{4, 5, 6\}$, and over a sample $n = \{7, 8, 9, 10\}$, along with the upper bound of balanced Boolean functions and lower bound of f_σ as per Theorem 7.4.4.

n	4	5	6	7	8	9	10
Max NL by experiment	4	12	26	54	116	232	480
% functions at max NL	100	2.34	6.51	8.2×10^{-3}	3.8×10^{-3}	8.5×10^{-4}	9.5×10^{-5}
Nonlinearity upper bound of balanced functions [93]	4	12	28	58	118	244	494
Min NL by experiment	4	6	14	34	64	188	334
% functions at min NL	100	0.52	1.04	3.1×10^{-6}	3.2×10^{-3}	9.5×10^{-5}	9.5×10^{-5}
Average NL by experiment	4.0	9.59	22.05	48.51	106.01	220.18	460.45
Nonlinearity lower bound by Theorem 7.4.4	3	4	14	22	63	98	252

Table 7.3: Percentage of $f_\sigma \in \mathcal{B}_n$ satisfying lower bound and upper value of $\mathsf{NL}(f_\sigma)$.

Theorem 7.4.5. *For $\sigma_n \in \mathbb{S}_n$ and $k \in [0, n]$, the number of orbits due to the action of $C_n = \langle \sigma_n \rangle$ on $\mathsf{E}_{k,n}$ is given by*

$$g_{k,n} = \frac{1}{n} \sum_{\{\sigma_n^i \in C_n : o(\sigma_n^i) | k\}} \binom{\gcd(n,i)}{\frac{k}{o(\sigma_n^i)}},$$

where $o(\sigma_n^i) = \frac{n}{\gcd(n,i)}$ is the order of σ_n^i.

Proof. For $1 \leq i \leq o(\sigma_n)$, denote $F_i = \{x \in \mathsf{E}_{k,n} : \sigma_n^i(x) = x\}$ be the set of fixed elements of σ_n^i in $\mathsf{E}_{k,n}$. To apply Burnside's Lemma for counting the number of orbits, we need to determine $|F_i|$. Observe that $o(\sigma_n^i) = \frac{n}{\gcd(n,i)}$. Hence, the number of disjoint cycles in σ_n^i is $\gcd(n,i)$, with each cycle having length $o(\sigma_n^i) = \frac{n}{\gcd(n,i)}$. A vector $x \in \mathbb{F}_2^n$ is fixed by σ_n^i if each cycle in x must be all 0 or all 1. For x being in $\mathsf{E}_{k,n}$, $o(\sigma_n^i) \mid k$ and there are $\frac{k}{o(\sigma_n^i)}$ cycles in x contain all 1 and remaining cycles contain all 0. Thus, there are $|F_i| = \binom{\gcd(n,i)}{\frac{k}{o(\sigma^i)}}$. Hence, applying Burnside's Lemma, we have

$$g_{k,n} = \frac{1}{n} \sum_{\sigma_n^i \in C_n} |F_i| = \frac{1}{n} \sum_{\{\sigma_n^i \in C_n : o(\sigma_n^i) | k\}} \binom{\gcd(n,i)}{\frac{k}{o(\sigma_n^i)}}.$$

□

Lemma 7.4.6. *For $\sigma_n \in \mathbb{S}_n$, the number of orbits of odd cardinality due to the action of $C_n = \langle \sigma_n \rangle$ on $\mathsf{E}_{k,n}$ is given by*

$$g_{k,n,odd} = \begin{cases} 0 & \textit{if } 2^e \nmid k \\ g_{\frac{k}{2^e}, n_0} & \textit{if } 2^e \mid k, \end{cases}$$

where n_0 is the largest odd integer that divides n and $2^e = \frac{n}{n_0}$ i.e., e is the largest integer such that 2^e divides n.

Proof. As in the proof of Theorem 7.4.3, for $d \mid n$, if $|\mathsf{O}_{\sigma_n}(x)| = d$ for an $x \in \mathsf{E}_{k,n}$ then there is a $v \in \mathbb{F}_2^d$ such that $x = (v, v, \ldots, v)$ with $|\mathsf{O}_{\sigma_d}(v)| = d$ in $\mathbb{F}_2^d$ and $\mathsf{w_H}(v) = \frac{k}{n/d} = \frac{kd}{n}$. Thus, in this case, $\frac{n}{d} \mid k$ and hence $v \in \mathsf{E}_{\frac{kd}{n},d}$.

For any odd divisor d of n, if $\frac{n}{d} \mid k$ then $2^e \mid k$. That is, if $2^e \nmid k$ then $\frac{n}{d} \nmid k$ for every odd divisor d of n and that implies $g_{k,n,odd} = 0$.

Now consider the case $2^e \mid k$. Every odd divisor d of n is also an divisor of n_0 and viceversa. Further, for any odd integer d of n, if $\frac{n}{d} \nmid k$ then $\frac{n_0}{d} \nmid \frac{k}{2^e}$ and viceversa. Hence, $|\mathsf{O}_{\sigma_n}(x)| = d$ for an $x \in \mathsf{E}_{k,n}$ with d is odd, $d \mid n$ and $\frac{n}{d} \mid k$ iff there is a $v \in \mathsf{E}_{\frac{kd}{n},d}$ such that $x = (v, v, \ldots, v)$ and $|\mathsf{O}_{\sigma_d}(v)| = d$ iff $|\mathsf{O}_{\sigma_{n_0}}(y)| = d$ where $y = (v, v, \ldots, v) \in \mathsf{E}_{\frac{k}{2^e},n_0}$. Therefore, $g_{k,n,odd} = g_{\frac{k}{2^e},n_0}$. □

Theorem 7.4.7. *Let $n \geq 2$ be a positive integer and $\sigma_n \in \mathbb{S}_n$ be the rotation permutation. Then*

$$\mathsf{NL}_k(f_{\sigma_n}) \geq \begin{cases} \frac{1}{4}\left(\binom{n}{k} + \min\limits_{\substack{2 \leq l \leq n \\ l \text{ even}}} \mathsf{K}_k(l,n)\right) & \text{if } 2^e \nmid k \\ \frac{1}{4}\left(\binom{n}{k} - 2g_{\frac{k}{2^e},n_0} + \min\limits_{\substack{2 \leq l \leq n \\ l \text{ even}}} \mathsf{K}_k(l,n)\right) & \text{if } 2^e \mid k, \end{cases}$$

where n_0 is the largest odd intger that divides n and $2^e = \frac{n}{n_0}$

Proof. The proof can directly be obtained by substituting the value of $|\mathcal{O}_{o,k}| = g_{k,n,odd}$ from Lemma 7.4.6 in Theorem 7.2.3. □

7.5 Experimental results and Comparisons

In this section, we present experimental results on the cryptographic properties, nonlinearity (NL), weightwise nonlinearity (NL_k), algebraic immunity (AI), and weightwise algebraic immunity (AI_k), of functions from the 2-ψS and 2-σS WAPB classes. Additionally, we compare these functions with notable classes of WPB Boolean functions in 8 and 16 variables with respect to the same properties. It should be noted that for $n = 8$ and $n = 16$, the WPB Boolean function classes 2-ψS and 2-σS are identical. Consequently, we study only one representative class for these dimensions and include it in the comparative analysis.

The number of 2-ψS and 2-σS WAPB functions grows prohibitively large for 7 or more variables. For this reason, our experimental evaluation considers:

- All 2-ψS and 2-σS WAPB functions for $n = 4, 5, 6$, yielding exact results.
- Large random samples of such functions for $n \in [7, 16]$.

Thus, the results reported for $n \in [7, 16]$ are based on sampling and represent lower bounds for the maximum attainable values of each property and upper bounds for the minimum attainable values. For example, the entry value $4, 5$ for $\mathsf{AI}_6(f_\psi)$ in Table 7.9 for $n = 13$ indicates the observed minimum and maximum values within the sample. This implies that the true maximum satisfies $\max \mathsf{AI}_6(f_\psi) \geq 5$, and the true minimum satisfies $\min \mathsf{AI}_6(f_\psi) \leq 4$.

7.5.1 Nonlinearity

Table 7.4 presents the maximum and minimum nonlinearity attained by 2-ψS and 2-σS functions for $n \in [4, 10]$. The table also compares these values with the known upper bound for balanced Boolean functions and the theoretical lower bounds for 2-ψS and 2-σS functions, as given by Theorem 7.3.5 and Theorem 7.4.4, respectively. For $n \leq 6$, our results are exhaustive, i.e., we have examined all balanced Boolean functions of the respective classes. For larger numeber of variables ($n = 7, 8, 9, 10$), the data are derived from extensive random sampling, with sample sizes of approximately 70×2^{20}, 2×2^{25}, 6×2^{23}, and 6×2^{23} functions, respectively. Additionally, Table 7.4 compares the nonlinearity between the 2-ψS and 2-σS functions. A further comparison of the NL values for WPB functions f_ψ and f_σ on 8 and 16 variables against other prominent classes of WPB Boolean functions is provided in Table 7.5.

n	4	5	6	7	8	9	10
Nonlinearity upper bound of any balanced functions [93]	4	12	28	58	118	244	494
Max NL(f_ψ) by experiment	4	12	26	56	116	238	480
% functions f_ψ at max NL	100	22.92	0.65	4.3×10^{-3}	4.6×10^{-3}	1.6×10^{-5}	2.4×10^{-3}
Max NL(f_σ) by experiment	4	12	26	54	116	232	480
% functions f_σ at max NL	100	2.34	6.51	8.2×10^{-3}	3.8×10^{-3}	8.5×10^{-4}	9.5×10^{-5}
Nonlinearity lower bound of f_ψ by Theorem 7.3.5	3	6	14	28	63	144	254
Min NL(f_ψ) by experiment	4	6	14	28	64	144	328
% functions f_ψ at min NL	100	4.17	0.26	2.3×10^{-4}	3.3×10^{-3}	3.2×10^{-5}	1.6×10^{-5}
Nonlinearity lower bound of f_σ by Theorem 7.4.4	3	4	14	22	63	98	252
Min NL(f_σ) by experiment	4	6	14	34	64	188	334
% functions f_σ at min NL	100	0.52	1.04	3.1×10^{-6}	3.2×10^{-3}	9.5×10^{-5}	9.5×10^{-5}
Average NL(f_σ) by experiment	4.0	9.59	22.05	48.51	106.01	220.18	460.45

Table 7.4: Experimental results of nonlinearity of 2-ψS, 2-σS WAPB Boolean functions.

Variables **Function**	8	16
Minimum [46]	8	128
[98]	$[66, 82]$	NA
[20]	88	29488
[44]	96	30704
[72]	$[110, 112]$	NA
[46]	$[112, 116]$	$[32512, 32598]$
$f_\psi = f_\sigma$	$[64, 116]$	$[31280, 32378]$
Upper bound	118	32638

Table 7.5: Comparison of nonlinearity of WPB functions in 8 and 16 variables.

7.5.2 Weightwise nonlinearity

Table 7.6 reports the maximum weightwise nonlinearity of 2-ψS and 2-σS functions for $n \in [4, 16]$. The structure of the table is as follows: for each n (except $n = 4, 8, 16$), two rows of entries are given, where the upper row corresponds to $\mathsf{NL}_k(f_\psi)$ and the lower row to $\mathsf{NL}_k(f_\sigma)$. For $n = 4, 8, 16$, only one row is presented because the two function classes coincide for these numbers of variables.

For $n = 4, 5, 6$, the results are obtained from an exhaustive search over all functions of the respective classes. For $n \in [7, 16]$, the values are derived from a large random sample. Consequently, all entries for $7 \leq n \leq 16$ represent lower bounds on the true maximum weightwise nonlinearity. In addition to comparing max NL_k between f_ψ and f_σ functions, we also provide a comparison of NL_k for the 2-ψS (equivalently, 2-σS) WPB functions in 16 variables against other notable classes of WPB Boolean functions in Table 7.7.

f_ψ f_σ n	NL_2	NL_3	NL_4	NL_5	NL_6	NL_7	NL_8	NL_9	NL_{10}	NL_{11}	NL_{12}	NL_{13}	NL_{14}
4	1												
5	3 3	3 3											
6	3 4	6 6	3 4										
7	5 6	11 12	11 12	5 6									
8	9	22	27	22	9								
9	12 12	34 33	55 55	55 55	34 33	12 12							
10	14 14	48 52	93 91	110 112	93 91	48 52	14 14						
11	18 18	68 66	145 142	207 205	207 205	145 142	68 66	18 18					
12	23 23	93 93	220 220	360 364	422 426	360 364	220 220	93 93	23 23				
13	27 28	121 120	322 319	596 592	802 799	802 799	596 592	322 319	121 120	27 28			
14	31 31	154 158	454 458	930 944	1413 1432	1628 1644	1413 1432	930 944	454 458	154 158	31 31		
15	36 38	194 193	622 620	1407 1406	2386 2383	3079 3082	3079 3082	2386 2383	1407 1406	622 620	194 193	36 38	
16	44	246	846	2083	3867	5556	6259	5556	3867	2083	846	246	44

Table 7.6: Experimental results on NL_k of 2-ψS and 2-σS functions for $2 \leq k \leq n-2$

Function	NL_2	NL_3	NL_4	NL_5	NL_6	NL_7	NL_8
Construction 1 [66]	≥ 5	≥ 144	≥ 472	≥ 1056	≥ 2184	≥ 1296	≥ 2184
Construction 2 [43]	2	6	52	226	1500	2502	3002
[20]	4	56	350	1288	3108	4774	5539
[74]	40	204	765	1814	3484	5138	5875
$f_\psi = f_\sigma$	44	246	846	2083	3867	5556	6259
Upper bound [20]	54	268	888	2150	3959	5666	6378

Table 7.7: NL_k of WPB functions in 16 variables.

7.5.3 Algebraic immunity

The experimental investigation into the algebraic immunity for all 2-ψS and 2-σS functions for $n = 4, 5, 6$ (exhaustive search) and for a large random sample of functions for $n = 7, 8, \ldots, 16$ are summarized in Table 7.8. The analysis confirms that both classes contain functions achieving the maximum possible algebraic immunity. Notably, while the probability of a random Boolean function in an odd number of variables attaining maximum AI is known to be extremely small, we nevertheless identified functions from both the 2-ψS and 2-σS families—including those with odd n—that achieve this bound.

n	4	5	6	7	8	9	10
max AI	2	3	3	4	4	5	5
AI(f_ψ)	2	$2, 3$	$2, 3$	$2, 3, 4$	$2, 3, 4$	$3, 4, 5$	$4, 5$
% of f_ψ	100	$58.33, 41.67$	$2.83, 97.17$	$0.01, 90.98, 9.01$	$0.002, 4.76, 95.24$	$0.02, 93.99, 5.99$	$0.31, 99.69$
AI(f_σ)	2	$2, 3$	$2, 3$	$3, 4$	$2, 3, 4$	$4, 5$	5
% of f_σ	100	$66.93, 33.07$	$4.69, 95.31$	$82.09, 17.91$	$0.002, 4.76, 95.24$	$97.03, 2.97$	100

n	11	12	13	14	15	16	
max AI	6	6	7	7	8	8	
AI(f_ψ)	$5, 6$	6	$6, 7$	7	$7, 8$	8	
% of f_ψ	$99.04, 0.96$	100	$99.35, 0.65$	100	$0.999, 0.001$	100	
AI(f_σ)	$5, 6$	6	$6, 7$	7	7	8	
% of f_σ	$98.19, 1.81$	100	$99.11, 0.89$	100	100	100	

Table 7.8: Experimental results of AI of 2-ψS and 2-σS functions

7.5.4 Weightwise algebraic immunity

We performed an experimental analysis of the weightwise algebraic immunity 2-ψS and 2-σS functions. The results are summarized in Table 7.9, which includes exhaustive data for all 2-ψS and 2-σS functions with $n = 4, 5, 6$, and sampled data for $n \in [8, 16]$. The table is organized as follows: for each n (except $n = 4, 8, 16$), two rows are given, the upper row corresponds to $\mathsf{AI}_k(f_\psi)$ and the lower row to $\mathsf{AI}_k(f_\sigma)$. For $n = 4, 8, 16$, only one row is provided since the two function classes coincide for these numbers of variables.

Because the data for $n \in [7, 16]$ are obtained from random samples, the reported values constitute lower bounds on the true maximum weightwise algebraic immunity. The table also allows a comparison of max AI_k between 2-ψS and 2-σS functions. Furthermore, we compare AI_k for the 2-ψS (equivalently,, 2-σS) WPB functions inn 8 and 16 variables against other prominent WPB Boolean function classes in Table 7.10 and Table 7.11. We note the following structural symmetry: if $f(x_1, x_2, \ldots, x_n) \in \mathcal{B}_n$ is a 2-ψS (respectively, 2-σS) function then $g = f(1 + x_1, 1 + x_2, \ldots, 1 + x_n) \in \mathcal{B}_n$ is also a 2-ψS (respectively, 2-σS) function. Consequently, $\mathsf{AI}_k(f) = \mathsf{AI}_{n-k}(g)$ which explains the symmetry observed between the entries for AI_k and AI_{n-k}.

Function	AI_2	AI_3	AI_4	AI_5	AI_6
[20]	1	2	2	2	2
[105]	2	2	2	2	2
[29]	1	2	3	2	2
[74]	1	2	3	2	2
[98]	2	2	3	2	2
max $f_\psi = f_\sigma$	2	2	3	2	2
Upper bound	2	3	3	3	2

Table 7.10: Comparison of AI_k of WPB functions in 8 variables.

f_ψ f_σ n	AI_2	AI_3	AI_4	AI_5	AI_6	AI_7	AI_8	AI_9	AI_{10}	AI_{11}	AI_{12}	AI_{13}	AI_{14}
4	1												
5	2 $2, 3$	2 $2, 3$											
6	$1, 2$ 2	$1, 2$ $1, 2$	$1, 2$ 2										
7	$1, 2$ 2	2 2	2 2	$1, 2$ 2									
8	2	$1, 2$	$2, 3$	$1, 2$	2								
9	2 $1, 2$	$2, 3$ $2, 3$	$2, 3$ $2, 3$	$2, 3$ $2, 3$	$2, 3$ $2, 3$	2 $1, 2$							
10	$1, 2$ 2	$1, 2, 3$ $1, 2, 3$	$2, 3$ 3	$2, 3, 4$ $3, 4$	$2, 3$ 3	$1, 2, 3$ $1, 2, 3$	$1, 2$ 2						
11	$1, 2$ $1, 2$	$2, 3$ $2, 3$	3 $3, 4$	$3, 4$ 4	$3, 4$ 4	3 $3, 4$	$2, 3$ $2, 3$	$1, 2$ $1, 2$					
12	2 2	$2, 3$ $2, 3$	$2, 3, 4$ $3, 4$	$3, 4$ $3, 4$	4 $3, 4$	$3, 4$ $3, 4$	$2, 3, 4$ $3, 4$	$2, 3$ $2, 3$	2 2				
13	2 $1, 2$	$2, 3$ $2, 3$	$3, 4$ $3, 4$	4 4	$4, 5$ 5	$4, 5$ 5	4 4	$3, 4$ $3, 4$	$2, 3$ $2, 3$	2 $1, 2$			
14	$1, 2$ 2	$2, 3$ $2, 3$	$3, 4$ $3, 4$	4 $4, 5$	$4, 5$ 5	5 5	$4, 5$ 5	4 $4, 5$	$3, 4$ $3, 4$	$2, 3$ $2, 3$	$1, 2$ 2		
15	$1, 2$ 2	$2, 3$ $2, 3$	$3, 4$ $3, 4$	$4, 5$ $4, 5$	5 5	6 6	6 6	5 5	$4, 5$ $4, 5$	$3, 4$ $3, 4$	$2, 3$ $2, 3$	$1, 2$ 2	
16	2	$2, 3$	4	$4, 5$	5	6	6	6	5	$4, 5$	4	$2, 3$	2

Table 7.9: Experimental results of AI_k of 2-ψS and 2-σS functions for $2 \leq k \leq n-2$

Function	AI_2	AI_3	AI_4	AI_5	AI_6	AI_7	AI_8	AI_9	AI_{10}	AI_{11}	AI_{12}	AI_{13}	AI_{14}
[105]	2	2	2	2	2	2	2	2	2	2	2	2	2
[20]	1	2	2	3	3	3	3	3	3	3	3	3	2
[29]	1	2	3	3	3	3	3	3	3	3	3	2	2
[98]	2	2	3	3	4	4	5	4	4	3	3	2	2
[74]	2	3	4	4	5	6	6	6	5	4	3	2	2
max $f_\psi = f_\sigma$	2	3	4	5	5	6	6	6	5	5	4	3	2
Upper bound	2	3	4	5	5	6	6	6	5	5	4	3	2

Table 7.11: Comparison of AI_k of WPB functions in 16 variables.

7.5.5 Summary

In this subsection, we summarize the experimental results concerning the nonlinearity (NL), weightwise nonlinearity (NL_k), algebraic immunity (AI), and weightwise algebraic immunity (AI_k) of 2-ψS and 2-σS functions. We compare their performance with prominent classes of WPB Boolean functions from the literature [20, 98, 72, 44, 46, 74] in 8 and 16 variables. The comparison data for other functions are drawn from [46, 43, 11] for$\mathsf{NL}, \mathsf{NL}_k, \mathsf{AI}_k$ respectively. Note that for $n = 8$ and $n = 16$, the class of 2-ψS and 2-σS functions coincide.

The key findings of the comparison are outlined below.

- **Nonlinearity :** The nonlinearity profile of 2-ψS and 2-σS functions, provided in Table 7.4, indicates that both classes attain high nonlinearity. Furthermore, as shown in Table 7.5, the nonlinearity of the 2-ψS and 2-σS functions ranks second among the compared WPB functions.

- **Weightwise nonlinearity :** Table 7.6 presents the weightwise nonlinearity profile of 2-ψS and 2-σS functions, demonstrating that both achieve high weightwise nonlinearity with similar profile characteristics. Notably, Table 7.7 confirms that the weightwise nonlinearities of 2-ψS and 2-σS functions are the highest among all compared WPB functions.

- **Algebraic immunity :** The algebraic immunity profile, displayed in Table 7.8, reveals that functions from these classes can reach the maximum or near-maximum AI. The AI profiles of 2-ψS and 2-σS appear largely similar.

- **Weightwise algebraic immunity :** Table 7.9 summarizes the AI_k profile across slices $\mathsf{E}_{k,n}$, showing that both classes can attain maximum or high weightwise algebraic immunity. The results indicate a slightly better AI_k profile for 2-σS functions compared to 2-ψS. Importantly, as evidenced in Table 7.10 and Table 7.11, the weightwise algebraic immunity of functions in these classes is superior to that of all other compared WPB functions.

From this comparative analysis, we conclude that it is possible to construct 2-ψS or 2-σS functions that are WAPB and simultaneously achieve high (and in some cases maximal) nonlinearity, weightwise nonlinearity, algebraic immunity, and weightwise algebraic immunity.

7.6 Conclusion

In this chapter, we have generalized the construction of WPB Boolean functions, initially proposed by Liu and Mesnager [66], to a broad class of weightwise almost perfectly balanced Boolean functions. Our generalization is based on the action of cyclic subgroups of the symmetric group (*i.e.*, $\langle\pi\rangle$) on n-element sets, particularly allowing us to overcome the restriction of n being a power of two. As a consequence, we have introduced and analyzed two new classes of WAPB Boolean functions: $2-\sigma S$ and $2-\psi S$ Boolean functions. Both constructions include the Liu-Mesnager construction as a special case when n is a power of 2.

We have provided lower bounds on the nonlinearity and weightwise nonlinearities for all functions from this class, based on the number of orbits of odd sizes. In particular, we have focused on the cryptographic properties of WAPB functions obtained from two specific permutations ψ and σ, demonstrating that these functions achieve good nonlinearity and weightwise nonlinearity. Specifically, we have shown that the weightwise nonlinearity bounds for the largest slices (i.e., when $k = \lfloor \frac{n}{2} \rfloor$) and for $k = 2$. Our results have improved the previously established bound on NL_2 for the Liu-Mesnager functions, and provided experimental insights into the distribution of the nonlinearity for the class of WAPB functions we have introduced.

Chapter 8

Conclusions

This thesis has focused on the construction of Boolean functions that are balanced or almost balanced over $\mathsf{E}_{k,n}$ for all $k \in [0, 1]$. Such functions are known as weightwise perfectly balanced (WPB) or weightwise almost perfectly balanced (WAPB) Boolean functions. These functions are particularly relevant in the design of stream ciphers where the input to the Boolean function is restricted to the vectors in $\mathbb{F}_2^n$ with constant Hamming weight, as seen in the architecture of FLIP cipher.

We have explored secondary and recursive constructions of WAPB Boolean functions based on the Siegenthaler's construction over a restricted domain and a lifting technique on known support of lower dimensional WAPB functions. These methods have generalized to handle arbitrary n, prior to the previous constructions that were restricted to $n = 2^m$ for $m \in \mathbb{Z}^+$. We have introduced a class of WAPB Boolean functions based on group action of a cyclic subgroup $P = \langle \psi \rangle$ of the symmetric group over $\mathbb{F}_2^n$: $2 - \psi S$ Boolean functions. This class of functions have generalized the construction of WPB Boolean functions, initially proposed by Liu and Mesnager [66], to a broad class of WAPB Boolean functions. We have analyzed direct sum constructions for WPB and WAPB Boolean functions. We have presented conditions under which the direct sum of two WAPB or WPB functions results in a WAPB or WPB function.

8.1 Open Problems and Future Work

Although this thesis has some substantial contributions to the analysis of WAPB Boolean functions, several open questions and promising research directions remain to be explored.

1. The existing upper bounds on weightwise nonlinearity *i.e.*$\mathsf{NL}_k(f) < \frac{1}{2}[\binom{n}{k} - \sqrt{\binom{n}{k}}]$ (except the case $n = 50, k = 3$, which is still open to compute) are known to be loose for many values of k for example $k = 1, n-1$. Finding the tighter upper bounds for each k or a function that achieve the $\mathsf{NL}_k(f)$ remains a challenging problem.

2. As we have defined two new classes of WAPB Boolean functions that are $2-\sigma S$ and $2-\psi S$ Boolean functions in Chapter 7. As we have computed the weightwise nonlinearity of these class of Boolean functions and have shown that the lower bound is depends on the $\min_{\substack{2\leq l\leq n\\ l \text{ even}}} \mathsf{K}_k(l,n)$ for each $k \in [2, n-2]$. Therefore, several cases for $k \in [2, n-2]$ is still remain to work due to difficulties in computing $\min_{\substack{2\leq l\leq n\\ l \text{ even}}} \mathsf{K}_k(l,n)$.

3. There are still several cases to be remain to figure out in what conditions the direct sum of WAPB or WPB can be a WAPB or WPB. For example, to study the direct sum $h(x,y) = f(x) + g(y)$ in Theorem 6.2.2 when $e(m) \cap e(n) \neq \emptyset$.

4. A comprehensive theory for weightwise algebraic immunity *i.e.*$\mathsf{AI}_k(f)$ has not been studied rigorously. New methods for designing WAPB functions with high weightwise algebraic immunity are needed to be studied.

References

[1] Carlisle Adams and Jeff Gilchrist. The CAST-256 encryption algorithm. Technical report, Network Working Group, 1999. Available at https://www.rfc-editor.org/rfc/rfc2612.

[2] Alfred V Aho and NJA Sloawe. Some doubly exponential sequences. *The Fibonacci Quarterly*, 11(4):429–437, 1973.

[3] Martin Aigner and Günter M Ziegler. Proofs from the book. *Berlin. Germany*, 1(2):7, 1999.

[4] Noga Alon and Benny Sudakov. Bipartite subgraphs and the smallest eigenvalue. *Combinatorics, Probability and Computing*, 9(1):1–12, 2000.

[5] Steve Babbage and Matthew Dodd. The mickey stream ciphers. In *New Stream Cipher Designs: The eSTREAM Finalists*, pages 191–209. Springer Berlin Heidelberg, 2008.

[6] Côme Berbain, Olivier Billet, Anne Canteaut, Nicolas Courtois, Henri Gilbert, Louis Goubin, Aline Gouget, Louis Granboulan, Cédric Lauradoux, Marine Minier, Thomas Pornin, and Hervé Sibert. Sosemanuk, a fast software-oriented stream cipher. In *New Stream Cipher Designs: The eSTREAM Finalists*, pages 98–118. Springer Berlin Heidelberg, 2008.

[7] Daniel J. Bernstein. ChaCha, a variant of Salsa20. Technical report, Workshop record of SASC, 2008. Available at https://cr.yp.to/chacha/chacha-20080120.pdf.

[8] Daniel J. Bernstein. The Salsa20 family of stream ciphers. In *New Stream Cipher Designs: The eSTREAM Finalists*, pages 84–97. Springer, 2008.

[9] Eli Biham, Ross J. Anderson, and Lars R. Knudsen. Serpent: A new block cipher proposal. In *Fast Software Encryption, 5th International Workshop, FSE '98, Paris, France, March 23-25, 1998, Proceedings*, volume 1372 of *Lecture Notes in Computer Science*, pages 222–238. Springer, 1998.

[10] Martin Boesgaard, Mette Vesterager, and Erik Zenner. The Rabbit stream cipher. In *New Stream Cipher Designs: The eSTREAM Finalists*, pages 69–83. Springer, 2008.

[11] Luca Bonamino and Pierrick Méaux. Computing the restricted algebraic immunity, and application to weightwise perfectly balanced functions. In Yongdae Kim, Atsuko Miyaji, and Mehdi Tibouchi, editors, *Cryptology and Network Security*, pages 142–169. Springer Nature Singapore, 2026.

[12] Dan Boneh and Victor Shoup. A graduate course in applied cryptography, 2020. Available at `https://r.jordan.im/download/technology/BonehShoup_0_5.pdf`.

[13] An Braeken and Bart Preneel. On the algebraic immunity of symmetric boolean functions. In *Progress in Cryptology - INDOCRYPT 2005, 6th International Conference on Cryptology in India, Bangalore, India, December 10-12, 2005, Proceedings*, volume 3797 of *Lecture Notes in Computer Science*, pages 35–48. Springer, 2005.

[14] Andries E. Brouwer, Sebastian M. Cioabă, Ferdinand Ihringer, and Matt McGinnis. The smallest eigenvalues of Hamming graphs, Johnson graphs and other distance-regular graphs with classical parameters. *Journal of Combinatorial Theory, Series B*, 133:88–121, 2018.

[15] Christophe De Cannière. Trivium: A stream cipher construction inspired by block cipher design principles. In *Information Security, 9th International Conference, ISC 2006, Samos Island, Greece, August 30 - September 2, 2006, Proceedings*, volume 4176 of *Lecture Notes in Computer Science*, pages 171–186. Springer, 2006.

[16] Anne Canteaut and Marion Videau. Symmetric Boolean functions. *IEEE Transactions on Information Theory*, 51(8):2791–2811, 2005.

[17] Claude Carlet. On the secondary constructions of resilient and bent functions. In *Coding, Cryptography and Combinatorics*, volume 23 of *Progress in Computer Science and Applied Logic*, pages 3–28. Birkhäuser, Basel, 2004.

[18] Claude Carlet. On bent and highly nonlinear balanced/resilient functions and their algebraic immunities. In *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes, 16th International Symposium, AAECC-16, Las Vegas, NV, USA, February 20-24, 2006, Proceedings*, volume 3857 of *Lecture Notes in Computer Science*, pages 1–28. Springer, 2006.

[19] Claude Carlet. *Boolean Functions for Cryptography and Coding Theory*. Cambridge University Press, 2021.

[20] Claude Carlet, Pierrick Méaux, and Yann Rotella. Boolean functions with restricted input and their robustness; application to the FLIP cipher. *IACR Trans. Symmetric Cryptol.*, 2017(3):192–227, 2017.

[21] Lavinia C. Ciungu and Miodrag C. Iovanov. On the matrix of rotation symmetric Boolean functions. *Discrete Mathematics*, 341(12):3271–3280, 2018.

[22] Nicolas T. Courtois. Fast algebraic attacks on stream ciphers with linear feedback. In *Advances in Cryptology - CRYPTO 2003, 23rd Annual International Cryptology*

Conference, Santa Barbara, California, USA, August 17-21, 2003, Proceedings, volume 2729 of *Lecture Notes in Computer Science*, pages 176–194. Springer, 2003.

[23] Nicolas T. Courtois and Willi Meier. Algebraic attacks on stream ciphers with linear feedback. In *Advances in Cryptology - EUROCRYPT 2003, International Conference on the Theory and Applications of Cryptographic Techniques, Warsaw, Poland, May 4-8, 2003, Proceedings*, volume 2656 of *Lecture Notes in Computer Science*, pages 345–359. Springer, 2003.

[24] Thomas W. Cusick and Pantelimon Stanica. Fast evaluation, weights and nonlinearity of rotation-symmetric functions. *Discrete Mathematics*, 258(1-3):289–301, 2002.

[25] Joan Daemen and Vincent Rijmen. AES proposal: Rijndael. Available at `https://www.cs.miami.edu/home/burt/learning/Csc688.012/rijndael/rijndael_doc_V2.pdf`, 1999.

[26] Deepak K. Dalai, Kishan C. Gupta, and Subhamoy Maitra. Results on algebraic immunity for cryptographically significant Boolean functions. In *Progress in Cryptology - INDOCRYPT 2004, 5th International Conference on Cryptology in India, Chennai, India, December 20-22, 2004, Proceedings*, volume 3348 of *Lecture Notes in Computer Science*, pages 92–106. Springer, 2004.

[27] Deepak K. Dalai, Subhamoy Maitra, and Sumanta Sarkar. Basic theory in construction of Boolean functions with maximum possible annihilator immunity. *Designs, Codes and Cryptography*, 40:41–58, 2006.

[28] Deepak K. Dalai and Krishna Mallick. A class of weightwise almost perfectly balanced Boolean functions with high weightwise nonlinearity. In *8th International*

Workshop on Boolean Functions and their Applications (BFA), 2023. Available at https://eprint.iacr.org/2024/422.

[29] Deepak K. Dalai and Krishna Mallick. A class of weightwise almost perfectly balanced Boolean functions. *Advances in Mathematics of Communications*, 18(2):480–504, 2024.

[30] Deepak K. Dalai and Krishna Mallick. Constructing WAPB Boolean functions from the Direct Sum of WAPB Boolean functions. In *Progress in Cryptology - INDOCRYPT 2024 - 25th International Conference on Cryptology in India, Chennai, India, December 18-21, 2024, Proceedings, Part I*, volume 15495 of *Lecture Notes in Computer Science*, pages 188–209. Springer, 2024.

[31] Deepak K. Dalai, Krishna Mallick, and Pierrick Méaux. Weightwise almost perfectly balanced functions, construction from a permutation group action view. Cryptology ePrint Archive: report 2024/2068, 2024. Available at https://eprint.iacr.org/2024/2068.

[32] Donald W. Davies. The Lorenz cipher machine SZ42. *Cryptologia*, 19(1):39–61, 1995.

[33] Whitfield Diffie and Martin E Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.

[34] Ilya Dumer and Olga Kapralova. Spherically punctured biorthogonal codes. *IEEE Transactions on Information Theory*, 59(9):6010–6017, 2013.

[35] Patrik Ekdahl and Thomas Johansson. A new version of the stream cipher SNOW. In *Selected Areas in Cryptography, 9th Annual International Workshop, SAC 2002, St.*

John's, Newfoundland, Canada, August 15-16, 2002. Revised Papers, volume 2595 of *Lecture Notes in Computer Science*, pages 47–61. Springer, 2002.

[36] Yuval Filmus. Friedgut-Kalai-Naor theorem for slices of the Boolean cube. *Chicago Journal of Theoretical Computer Science*, 2016(14), 2016.

[37] Yuval Filmus. An orthogonal basis for functions over a slice of the Boolean hypercube. *The Electronic Journal of Combinatorics*, 23:23–49, 2016.

[38] Yuval Filmus. Junta threshold for low degree Boolean functions on the slice. *The Electronic Journal of Combinatorics*, 30:55, 2023.

[39] Yuval Filmus and Ferdinand Ihringer. Boolean constant degree functions on the slice are juntas. *Discrete Mathematics*, 342(12):111614, 2019.

[40] Yuval Filmus, Guy Kindler, Elchanan Mossel, and Karl Wimmer. Invariance principle on the slice. *ACM Transactions on Computation Theory (TOCT)*, 10(3):1–37, 2018.

[41] Yuval Filmus and Elchanan Mossel. Harmonicity and invariance on slices of the Boolean cube. *Probability Theory and Related Fields*, 175:721–782, 2019.

[42] Caroline Fontaine. *E0 (Bluetooth)*. Encyclopedia of Cryptography and Security. Springer, 2005. Available at `https://doi.org/10.1007/0-387-23483-7_117`.

[43] Agnese Gini and Pierrick Méaux. On the weightwise nonlinearity of weightwise perfectly balanced functions. *Discrete Applied Mathematics*, 322:320–341, 2022.

[44] Agnese Gini and Pierrick Méaux. Weightwise Almost Perfectly Balanced Functions: Secondary constructions for all n and better weightwise nonlinearities. In *Progress in Cryptology - INDOCRYPT 2022 - 23rd International Conference on Cryptology in*

India, Kolkata, India, December 11-14, 2022, Proceedings, volume 13774 of *Lecture Notes in Computer Science*, pages 492–514. Springer, 2022.

[45] Agnese Gini and Pierrick Méaux. On the algebraic immunity of weightwise perfectly balanced functions. In *Progress in Cryptology - LATINCRYPT 2023 - 8th International Conference on Cryptology and Information Security in Latin America, Quito, Ecuador, October 3-6, 2023, Proceedings*, volume 14168 of *Lecture Notes in Computer Science*, pages 3–23. Springer, 2023.

[46] Agnese Gini and Pierrick Méaux. Weightwise perfectly balanced functions and nonlinearity. In *Codes, Cryptology and Information Security*, volume 13874 of *Lecture Notes in Computer Science*, pages 338–359. Springer, 2023.

[47] Henry W. Gould. *Combinatorial Identities: A Standardized Set of Tables Listing 500 Binomial Coefficient Summations*. The Fibonacci Quarterly, 1972. Availble at https://api.semanticscholar.org/CorpusID:273318544.

[48] Xiaoqi Guo and Sihong Su. Construction of weightwise almost perfectly balanced Boolean functions on an arbitrary number of variables. *Discrete Applied Mathematics*, 307:102–114, 2022.

[49] K Gyory. On the diophantine equation $\binom{n}{k} = x^l$. *Acta Arithmetica*, 80(3):289–295, 1997.

[50] Martin Hell, Thomas Johansson, Alexander Maximov, Willi Meier, Jonathan Sönnerup, and Hirotaka Yoshida. Grain-128AEADv2: Strengthening the initialization against key reconstruction. In *Cryptology and Network Security - 20th International Conference, CANS 2021, Vienna, Austria, December 13-15, 2021, Proceedings*, volume 13099 of *Lecture Notes in Computer Science*, pages 24–41. Springer, 2021.

[51] Martin Hell, Thomas Johansson, and Willi Meier. Grain: a stream cipher for constrained environments. *International Journal of Wireless and Mobile Computing*, 2(1):86–93, 2007.

[52] Martin Hell, Thomas Johansson, Willi Meier, Jonathan Sönnerup, and Hirotaka Yoshida. An AEAD variant of the Grain stream cipher. In *Codes, Cryptology and Information Security - Third International Conference, C2SI 2019, Rabat, Morocco, April 22-24, 2019, Proceedings - In Honor of Said El Hajji*, volume 11445 of *Lecture Notes in Computer Science*, pages 55–71. Springer, 2019.

[53] Tor Helleseth, Torleiv Kløve, and Johannes Mykkeltveit. On the covering radius of binary codes. *IEEE Transactions on Information Theory*, 24(5):627–628, 1978.

[54] Xiang-dong Hou. On the norm and covering radius of the first-order Reed-Muller codes. *IEEE Transaction on Information Theory*, 43(3):1025–1027, 1997.

[55] Michael Hughes. *Inside the Enigma*. Bloomsbury Publishing, 1997.

[56] Lin Jiao, Yonglin Hao, and Dengguo Feng. Stream cipher designs: A review. *Science China Information Sciences*, 63(3):131101, 2020.

[57] Jonathan Katz and Yehuda Lindell. *Introduction to Modern Cryptography*. Chapman and Hall/CRC Cryptography and Network Security Series, 2007.

[58] Selçuk Kavut and Melek D. Yücel. Generalized rotation symmetric and dihedral symmetric Boolean functions -9 variable Boolean functions with nonlinearity 242. In *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes, 17th International Symposium, AAECC-17, Bangalore, India, December 16-20, 2007, Proceedings*, volume 4851 of *Lecture Notes in Computer Science*, pages 321–329. Springer, 2007.

[59] Seluk Kavut, Subhamoy Maitra, and Melek D. Yucel. Search for Boolean functions with excellent profiles in the rotation symmetric class. *IEEE Transactions on Information Theory*, 53(5):1743–1751, 2007.

[60] Auguste Kerckhoffs. La cryptographie militaire. *J. Sci. Militaires*, 9(4):5–38, 1883.

[61] Andreas Klein. Linear Feedback Shift Registers. In *Stream Ciphers*, pages 17–58. Springer, 2013.

[62] Lars R Knudsen and Matthew Robshaw. *The block cipher companion*. Springer, 2011. Available at https://doi.org/10.1007/978-3-642-17342-4.

[63] Mikhail Krawtchouk. Sur une généralisation des polynômes d'Hermite. *Comptes Rendus*, 189(620-622):5–3, 1929.

[64] Adam Langley, W Chang, Nikos Mavrogiannopoulos, Joachim Strombergson, and Simon Josefsson. ChaCha20-Poly1305 cipher suites for transport layer security (TLS). Technical report, Internet Engineering Task Force (IETF), 2016.

[65] Jingjing Li and Sihong Su. Construction of weightwise perfectly balanced Boolean functions with high weightwise nonlinearity. *Discrete Applied Mathematics*, 279:218–227, 2020.

[66] Jian Liu and Sihem Mesnager. Weightwise perfectly balanced functions with high weightwise nonlinearity profile. *Designs, Codes and Cryptography*, 87(8):1797–1813, 2019.

[67] É. Lucas. Sur les congruences des nombres eulériens et des coefficients différentiels des fonctions trigonométriques suivant un module premier. *Bull. Soc. Math. France*, 6:49–54, 1878.

[68] F. J. MacWilliams and N. J. A. Sloane. *The Theory of Error-Correcting Codes*. North-Holland Publishing Company, 2nd edition, 1978.

[69] Subhamoy Maitra. Boolean functions on odd number of variables having nonlinearity greater than the bent concatenation bound. In *NATO Science for Peace and Security Series-D:Information and Communication Security, Boolean Functions in Cryptology and Information Security*, volume 18, pages 173–182. IOS Press, 2008.

[70] Subhamoy Maitra, Selcuk Kavut, and Melek D. Yücel. Balanced Boolean function on 13-variables having nonlinearity greater than the Bent concatenation bound. In *Proceedings of the conference Boolean Functions: Cryptography and Applications (BFCA), Copenhagen*, volume 8, pages 109–118, 2008.

[71] Subhamoy Maitra, Bimal Mandal, Thor Martinsen, Dibyendu Roy, and Pantelimon Stănică. Analysis on Boolean function in a restricted (biased) domain. *IEEE Transactions on Information Theory*, 66(2):1219–1231, 2019.

[72] Sara Mandujano, Juan Carlos Ku-Cauich, and Adriana Lara. Studying special operators for the application of evolutionary algorithms in the seek of optimal Boolean functions for cryptography. In *Advances in Computational Intelligence*, volume 13612 of *Lecture Notes in Computer Science*, pages 383–396. Springer, 2022.

[73] Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In *Advances in Cryptology - EUROCRYPT 1993, Workshop on the Theory and Application of of Cryptographic Techniques, Norway, May 23-27, 1993, Proceedings*, volume 765 of *Lecture Notes in Computer Science*, pages 386–397. Springer, 1993.

[74] Pierrick Méaux. Weightwise (almost) perfectly balanced functions based on total orders. *IACR Cryptol. ePrint Arch.*, page 647, 2024.

[75] Pierrick Méaux, Anthony Journault, François-Xavier Standaert, and Claude Carlet. Towards stream ciphers for efficient FHE with low-noise ciphertexts. In *Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I*, volume 9665 of *Lecture Notes in Computer Science*, pages 311–343. Springer, 2016.

[76] Alfred J. Menezes, Paul C. Van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography*. CRC press, 2018.

[77] Sihem Mesnager and Sihong Su. On constructions of weightwise perfectly balanced Boolean functions. *Cryptography and Communications*, 13(6):951–979, 2021.

[78] Sihem Mesnager, Sihong Su, and Jingjing Li. On concrete constructions of Weightwise Perfectly Balanced functions with optimal algebraic immunity and high weightwise nonlinearity. In *6th International Workshop on Boolean Functions and their Applications (BFA)*, 2021. Available at `https://boolean.w.uib.no/files/2021/08/BFA_2021_abstract_9.pdf`.

[79] Sihem Mesnager, Zhengchun Zhou, and Cunsheng Ding. On the nonlinearity of Boolean functions with restricted input. *Cryptography and Communications*, 11(1):63–76, 2019.

[80] Johannes Mykkelveit. The covering radius of the $[128, 8]$-Reed-Muller code is 56. *IEEE Transactions on Information Theory*, 26(3):359–362, 1980.

[81] Yoav Nir and Adam Langley. ChaCha20 and Poly1305 for IETF Protocols. Technical report, Internet Research Task Force(IRTF), 2015. Available at `https://doi.org/10.17487/RFC7539`.

[82] Nick J. Patterson and Douglas H. Wiedemann. The covering radius of the $[2^{15}, 16]$ Reed-Muller code is atleast 16276. *IEEE Transactions on Information Theory*, 29(3):354–356, 1983.

[83] Josef Pieprzyk and Cheng Xin Qu. Fast hashing and rotation-symmetric functions. *Journal of Universal Computer Science*, 5:20–31, 1999.

[84] Ronald L. Rivest, Matthew JB Robshaw, and Yiqun L. Yin. RC6 as the AES. Technical report, AES Candidate Conference, 2000. Available at `https://csrc.nist.rip/encryption/aes/round2/conf3/papers/rc6-statement.pdf`.

[85] Matthew Robshaw. The eSTREAM Project. In *New Stream Cipher Designs - The eSTREAM Finalists*, volume 4986 of *Lecture Notes in Computer Science*, pages 1–6. Springer, 2008.

[86] Oscar S. Rothaus. On “Bent” functions. *Journal of Combinatorial Theory, Series A*, 20(3):300–305, 1976.

[87] Rainer A Rueppel. *Analysis and design of stream ciphers*. Springer Science & Business Media, 2012.

[88] Andrew Rukhin, Juan Soto, James Nechvatal, Miles Smid, Elaine Barker, Stefan Leigh, Mark Levenson, Mark Vangel, David Banks, and Alan Heckert. A statistical test suite for random and pseudorandom number generators for cryptographic applications. Technical report, National Institute of Standards and Technology, 2010. Available at `https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=906762`.

[89] ETSI SAGE. Specification of the 3GPP confidentiality and integrity algorithms 128-EEA3 & 128-EIA3. Document 2: ZUC specification. Available at `https://www.`

gsma.com/about-us/wp-content/uploads/2014/12/eea3eia3zucv16.pdf, 2011.

[90] Peter Savický. On the bent Boolean functions that are symmetric. *European Journal of Combinatorics*, 15:407–410, 1994.

[91] Bruce Schneier, John Kelsey, Doug Whiting, David Wagner, Chris Hall, and Niels Ferguson. Twofish: A 128-bit block cipher. Technical report, NIST AES Proposal, 1998. Available at https://www.schneier.com/wp-content/uploads/2016/02/paper-twofish-paper.pdf.

[92] Jennifer Seberry, Xian-Mo Zhang, and Yuliang Zheng. On constructions and nonlinearity of correlation immune functions (extended abstract). In *Advances in Cryptology - EUROCRYPT 1993, Workshop on the Theory and Application of of Cryptographic Techniques, Norway, May 23-27, 1993, Proceedings*, volume 765 of *Lecture Notes in Computer Science*, pages 181–199. Springer, 1993.

[93] Jennifer Seberry, Xian-Mo Zhang, and Yuliang Zheng. Nonlinearity and propagation characteristics of balanced Boolean functions. *Information and Computation*, 119(1):1–13, 1995.

[94] Claude E. Shannon. Communication theory of secrecy systems. *The Bell System Technical Journal*, 28:656–715, 1949.

[95] Thomas Siegenthaler. Correlation-immunity of nonlinear combining functions for cryptographic applications. *IEEE Transactions on Information Theory*, 30(5):776–780, 1984.

[96] Pantelimon Stanica and Subhamoy Maitra. Rotation symmetric Boolean functions - count and cryptographic properties. *Discrete Applied Mathematics*, 156(10):1567–1580, 2008.

[97] Pantelimon Stănică, Subhamoy Maitra, and John A. Clark. Results on rotation symmetric bent and correlation immune Boolean functions. In *Fast Software Encryption, 11th International Workshop, FSE 2004, Delhi, India, February 5-7, 2004, Revised Papers*, volume 3017 of *Lecture Notes in Computer Science*, pages 161–177. Springer, 2004.

[98] Deng Tang and Jian Liu. A family of weightwise (almost) perfectly balanced Boolean functions with optimal algebraic immunity. *Cryptography and Communications*, 11(6):1185–1197, 2019.

[99] Edwin R. van Dam and Renata Sotirov. New bounds for the max-k-cut and chromatic number of a graph. *Linear Algebra and its Applications*, 488:216–234, 2016.

[100] Qichun Wang, Claude Carlet, Pantelimon Stanica, and Chik H. Tan. Cryptographic properties of the Hidden Weighted Bit function. *Discrete Applied Mathematics*, 174:1–10, 2014.

[101] Chuan K. Wu. *Boolean functions in cryptology*. PhD thesis, Xidian University, Xian, 1993.

[102] Hongjun Wu. *Cryptanalysis and design of stream ciphers*. PhD thesis, Katholieke Universiteit Leuven, 2008. Available at `https://cosicdatabase.esat.kuleuven.be/backend/publications/files/these/167`.

[103] Hongjun Wu. The stream cipher HC-128. In *New Stream Cipher Designs - The eSTREAM Finalists*, volume 4986 of *Lecture Notes in Computer Science*, pages 39–47. Springer, 2008.

[104] Rui Zhang and Sihong Su. A new construction of weightwise perfectly balanced Boolean functions. *Advances in Mathematics of Communications*, 17(4):757–770, 2023.

[105] Qinglan Zhao, Yu Jia, Dong Zheng, and Baodong Qin. A new construction of weightwise perfectly balanced functions with high weightwise nonlinearity. *Mathematics*, 11(5), 2023.

[106] Linya Zhu and Sihong Su. A systematic method of constructing weightwise almost perfectly balanced Boolean functions on an arbitrary number of variables. *Discrete Applied Mathematics*, 314:181–190, 2022.