Boolean Functions Related to Cryptography and Their Complexity

HIROSE Shouichi

February 1995

Boolean Functions Related to Cryptography and Their Complexity

HIROSE Shouichi

February 1995

Abstract

Cryptography has been used for more than a thousand of years to guarantee secure communications, and it is getting more and more important with the development of computers and networks. This thesis discusses properties and complexity of Boolean functions related to cryptography.

In Chapter 2, nonlinear Boolean functions are studied. Nonlinearity is a basic concept in the design and analysis of private key cryptosystems. Different types of private key cryptosystems require different types of nonlinearity, and several nonlinearity criteria have been proposed. Among them, the propagation criterion(PC) and the strict avalanche criterion(SAC) are focused on in this chapter.

First, a necessary and sufficient condition is presented for a Boolean function with n variables to satisfy the PC with respect to all but one elements in $\{0,1\}^n - \{(0,\ldots,0)\}$. A necessary and sufficient condition is also presented for a Boolean function with n variables to satisfy the PC with respect to all but linearly independent elements in $\{0,1\}^n - \{(0,\ldots,0)\}$. Second, the construction of Boolean functions with n variables is discussed that satisfy the PC with respect to all but one or three elements in $\{0,1\}^n - \{(0,\ldots,0)\}$. The methods can generate all such functions from all perfectly nonlinear Boolean functions. Third, an exact characterization of Boolean functions with n variables satisfying the PC of degree $n-2$ is obtained. Finally, relationships between the PC and the SAC are discussed.

Recently, two strong cryptanalytic attacks applicable to many private key ciphers were proposed. One is the differential cryptanalysis proposed by Biham and Shamir, and the other is the linear cryptanalysis proposed by Matsui. These attacks make use of linearity of the

ciphers and decrypt them much faster than the exhaustive search. The success of these attacks will require nonlinear Boolean functions with large numbers of inputs and outputs as components of private key ciphers in the near future.

In Chapter 3, complexity of Boolean functions satisfying the PC is discussed. First, it is shown that every Boolean function satisfying the PC of degree 1 are unate in at most two of its variables and that every Boolean function satisfying the PC of degree 2 is not unate in any one of its variables. Second, the optimal lower bound of $\lfloor \log n \rfloor - 1$ is obtained for the inversion complexity of the perfectly nonlinear Boolean functions constructed by the method of Maiorana. Third, the nearly optimal lower bound of $n^2/4-1$ is presented for the formula size of every Boolean function which satisfies the PC of degree 1. Fourth, the lower bound of $\Omega(n^2)$ is obtained for the AT^2 VLSI complexity of perfectly nonlinear Boolean functions with $n/2$ outputs each of whose output functions is constructed by the method of Maiorana. Finally, an exponential lower bound is presented for the numbers of nodes of ordered binary decision diagrams of perfectly nonlinear Boolean functions with multiple outputs constructed by the method of Nyberg.

Key management is a crucial problem when we use cryptosystems in practical cases. One method of the management is the secret sharing scheme proposed independently by Blakley and Shamir. Homogeneous Boolean functions and slice Boolean functions are considered to be practically important functions representing access structures which determine the strategy of the secret sharing scheme.

Homogeneous Boolean functions and slice Boolean functions are also important for computational complexity theory. It is one of the most difficult problems to prove a good lower bound on the circuit size complexity of some explicitly defined Boolean function. The best lower bounds proved on the circuit size complexity of explicitly defined Boolean functions are only linear. Because of the difficulty of this problem, more restricted types of circuits have been considered. For some explicitly defined homogeneous Boolean functions, even exponential lower bounds have been proved on their monotone circuit size complexity. These results, however, do not imply any nonlinear lower bound on the circuit size complexity because negation can be at least superpolynomially powerful for computing some homogeneous Boolean

functions. On the other hand, it has been proved that negation is powerless for computing slice Boolean functions.

In Chapter 4, circuit complexity of homogeneous Boolean functions and slice Boolean functions are studied. First, it is shown that there exist k-homogeneous Boolean functions with the property that the monotone circuit size complexity of its k-th slice is $\Omega({}_nC_k/\log {}_nC_k)$ and that of its $u(>k)$-th slice is $O(n\log n)$. This complexity gap is maximal when k is constant. Second, a set of homogeneous Boolean functions with circuit size complexity and monotone circuit size complexity almost equal is presented. For every Boolean function in this set, a lower bound of $\omega(n(\log n)^2)$ on the monotone circuit size complexity implies the same lower bound on the circuit size complexity.

Contents

Chapter 1

Introduction

1.1 Backgrounds

Cryptography has been used for more than a thousand of years to guarantee secure communications. And today, the development of computers and networks has been changing it drastically. Volumes of data and information are stored and processed by computers and communicated via public networks. Computers also enable complex and time-consuming cryptanalysis. Cryptography and cryptanalysis has become a science of information and data security, cryptology, and has been attracting many researchers.

Cryptography can be divided in two categories: private key cryptography and public key cryptography. The security of public key cryptography relies on some computationally difficult problems such as factoring and discrete logarithms. The security of private key cryptography relies on the fact that any efficient cryptanalysis has not yet been found.

Nonlinearity is a basic concept for the security of private key cryptography. Different types of private key ciphers require different types of nonlinearity, and several nonlinearity criteria have been proposed as design principles of private key ciphers. The propagation criterion(PC) is one of the nonlinearity criteria, which was proposed by Preneel, Leekwijk, Linden, Govaerts and Vandewalle[PLLGV91]. It is an extended notion of the perfect nonlinearity, which was defined by Meier

and Staffelbach[MS90]. The perfect nonlinearity is one of the most important nonlinearity criteria because the distance between the set of perfectly nonlinear Boolean functions and the set of affine Boolean functions is maximum. Perfectly nonlinear Boolean functions are, however, not balanced and their nonlinear order is at most one half of the number of their variables and are not suitable for the direct application to cryptography. Thus, it is valuable to investigate Boolean functions that satisfy the PC for the systematic generation of cryptographically useful Boolean functions.

Seberry, Zhang and Zheng[SZZ93] made an interesting research along this line. They presented methods for the construction of balanced Boolean functions satisfying the PC of high degrees. They proposed methods for constructing balanced Boolean functions with n variables satisfying the PC with respect to all but a few elements in $\{0,1\}^n - \{(0,\ldots,0)\}$ whose Hamming weights are large.

The first topic of this thesis is to characterize Boolean functions satisfying the PC. Exact characterizations are presented for Boolean functions satisfying the PC of degree $n-1$ and $n-2$ and for those satisfying the PC with respect to all but a few elements in $\{0,1\}^n - \{(0,\ldots,0)\}$.

Recently, two strong cryptanalytic attacks applicable to many private key ciphers were proposed. These attacks make use of linearity of the ciphers. One is the differential cryptanalysis, which was proposed by Biham and Shamir[BS93]. By this cryptanalysis, the Data Encryption Standard(DES) with 16 rounds can be decrypted with 2^{47} chosen plaintexts. The other is the linear cryptanalysis, which was proposed by Matsui [Mat94]. By this cryptanalysis, the DES with 16 rounds can be decrypted with 2^{43} known plaintexts.

The success of the differential cryptanalysis and the linear cryptanalysis will require nonlinear Boolean functions with large numbers of inputs and outputs as components of private key ciphers. Most of the existing private key block ciphers including the DES is for 64-bit blocks and the DES, for instance, has eight substitution boxes each of which has 6 input bits and 4 output bits. In the near future, it may be desired to design 128-bit-or-more block ciphers that have substitution boxes with large numbers of input and output bits. Thus, it is practically intresting to investigate the complexity of nonlinear

Boolean functions. It is also interesting from the theoretical point of view to explore the effect of nonlinearity of Boolean functions on their complexity.

The second topic of this thesis is complexity of Boolean functions satisfying the PC. The discussion focuses on lower bounds of their complexity because there exist a large number of Boolean functions satisfying the PC that are complex and difficult to compute.

Key management is an important problem when we use cryptosystems in practical cases. One method of the management is the secret sharing scheme proposed independently by Blakley[Bla79] and Shamir[Sha79]. They proposed the k-out-of-n threshold scheme, which enables us to construct from a given secret key n pieces of information with the property that the key can be recovered only from any k or more pieces of information. Their original idea can be generalized so that the secret key can be recovered only from any one of particular subsets of n pieces of information. A set of subsets of n pieces of information from which the secret key can be recovered is called an access structure.

It is natural to assume that, if the secret key can be recovered from a subset of pieces of information, it can also be recovered from any subset containing the subset. Under the assumption, every access structure can be represented by a monotone Boolean function. Homogeneous Boolean functions and slice Boolean functions are monotone and represent access structures considered to be practically important. A k-homogeneous Boolean function represents an access structure in which the secret key can be recovered from particular subsets consisting of k pieces of information. A k-slice Boolean function represents an access structure in which the secret key can be recovered from particular subsets consisting of k pieces of information or any subset consisting of $k+1$ pieces of information.

Homogeneous Boolean functions and slice Boolean functions are also important for computational complexity theory. It is one of the most difficult problems to prove a good lower bound on the circuit size complexity of some explicitly defined Boolean function. Although, for almost all Boolean functions, their circuit size complexity is exponential in the number of their inputs[Sha49], the best lower bounds proved on the circuit size complexity of explicitly defined Boolean functions

are linear[Blu84]. Because of the difficulty of proving a large lower bound on the circuit size complexity, more restricted types of circuits have been considered. Among them, monotone circuits is one of the most popular models. Good lower bounds on the monotone circuit size complexity have been obtained. For some explicitly defined homogeneous Boolean functions, even exponential lower bounds[And85, AB86] have been proved on their monotone circuit size complexity. These results do not imply any nonlinear lower bound on the circuit size complexity because negation can be at least superpolynomially powerful for computing some homogeneous Boolean functions[Raz85]. It is proved that negation is powerless for computing slice Boolean functions[Ber82, Weg85, Val86]. The monotone circuit size complexity of n-input slice Boolean functions is larger than their circuit size complexity at most by a multiplicative constant and an additive term of $O(n(\log n)^2)$[Weg85, Val86]. Thus, if a lower bound of $\omega(n(\log n)^2)$ is proved on the monotone circuit size complexity of a slice Boolean function, then the same lower bound can be obtained on its circuit size complexity. Any good lower bound has not been proved on the monotone circuit size complexity of explicitly defined slice Boolean functions.

From these facts, it is important to investigate slice Boolean functions and to find monotone Boolean functions whose circuit size complexity and monotone circuit size complexity are almost equal. The third and last topic of the thesis is circuit complexity of homogeneous Boolean functions and their slices.

1.2 Outline of the Thesis

This thesis studies properties and complexity of nonlinear Boolean functions and circuit complexity of homogeneous Boolean functions and their slices.

Chapter 2 discusses properties of nonlinearity criteria and relationships among them. It focuses on the PC, the strict avalanche criterion(SAC), and the nonlinearity. Many of the results are proved with the use of the Walsh transform of Boolean functions. First, a necessary and sufficient condition is presented for a Boolean function with n variables to satisfy the PC with respect to all but one elements in

$\{0,1\}^n - \{(0,\ldots,0)\}$. A necessary and sufficient condition is also presented for a Boolean function with n variables to satisfy the PC with respect to all but linearly independent elements in $\{0,1\}^n - \{(0,\ldots,0)\}$. Second, the construction of Boolean functions with n variables is discussed that satisfy the PC with respect to all but one or three elements in $\{0,1\}^n - \{(0,\ldots,0)\}$. The proposed methods of construction exactly characterize the Boolean functions satisfying the PC with respect to all but one elements in $\{0,1\}^n - \{(0,\ldots,0)\}$ and those satisfying the PC with respect to all but three elements in $\{0,1\}^n - \{(0,\ldots,0)\}$. Third, an exact characterization of Boolean functions with n variables satisfying the PC of degree $n-2$ is achieved. This condition says that, for every even $n \geqslant 4$, every Boolean function with n variables satisfying the PC of degree $n-2$ is perfectly nonlinear. Finally, some relationships between the PC and the SAC are presented.

In Chapter 3, complexity of Boolean functions satisfying the PC is discussed on several computation models. Investigated is the unateness, the inversion complexity, the formula size, the area-time-square tradeoff of VLSI circuits, and the numbers of nodes of OBDD's(Ordered Binary Decision Diagrams). First, some relationships are presented between the unateness and the degree of the PC. Non-unateness of Boolean functions satisfying the PC of degree more than 1 is proved. This implies that the PC does not compatible with the unateness. Second, the inversion complexity of perfectly nonlinear Boolean functions is discussed. An optimal lower bound is obtained for every perfectly nonlinear Boolean function constructed by the method of Maiorana[Rue91]. This bound implies that many $\neg$-gates are necessary to compute such functions. Third, a nearly optimal lower bound for the formula size of every Boolean function which satisfies the PC of degree 1 is presented. This lower bound is also nearly optimal for every perfectly nonlinear Boolean function. Finally, the area-time-square(AT^2) VLSI complexity[Ull84] and the OBDD[Bry86] size of perfectly nonlinear Boolean functions with multiple outputs is discussed. The results for the two complexity measures show the effect of un-correlation among the output functions to the computational complexity.

In Chapter 4, the circuit complexity of slice Boolean functions and homogeneous Boolean functions is considered. It is known that for any k-homogeneous Boolean function, its $(k+1)$-th slice is not much more

difficult to compute than its k-th slice[Dun86]. On the other hand, it has been proved that there exist k-homogeneous Boolean functions such that the monotone circuit complexity of their k-th slices is much larger than that of their $u(> k)$-th slices[Weg86]. One topic in Chapter 4 is an improvement of the latter result. An optimal lower bound is obtained on the monotone circuit size complexity of the k-th slices for constant k. The other topic is the homogeneous Boolean functions whose circuit size complexity and monotone circuit size complexity are almost equal. For these homogeneous Boolean functions with n variables, their monotone circuit size complexity is larger than their circuit size complexity at most by a constant factor and an additive term of $O(n(\log n)^2)$.

Chapter 5 is the conclusion of this thesis with some open questions.

Chapter 2

Nonlinear Boolean Functions

2.1 Introduction

This chapter discusses properties of nonlinearity criteria and relationships among them. It focuses on the propagation criterion(PC), the strict avalanche criterion(SAC), and the nonlinearity.

First, a necessary and sufficient condition is presented for a Boolean function with n variables to satisfy the PC with respect to all but one elements in $\{0,1\}^n - \{(0,\ldots,0)\}$. From this condition, it follows that, for every even $n \geqslant 2$, Boolean functions with n variables that satisfy the PC of degree $n-1$ are perfectly nonlinear, that is, satisfy the PC of degree n. It is also shown that Boolean functions with n variables that satisfy the PC with respect to all but linearly independent elements are perfectly nonlinear if $n \geqslant 2$ is even and that they satisfy the PC with respect to all but one elements in $\{0,1\}^n - \{(0,\ldots,0)\}$ if $n \geqslant 3$ is odd.

Second, we discuss the construction of Boolean functions with n variables that satisfy the PC with respect to all but one or three elements in $\{0,1\}^n - \{(0,\ldots,0)\}$.

Seberry, Zhang and Zheng[SZZ93] presented methods for the construction of balanced Boolean functions satisfying the PC of high degrees. For odd $n \geqslant 3$, they proposed a method for constructing balanced Boolean functions with n variables satisfying the PC with respect to all but one elements in $\{0,1\}^n - \{(0,\ldots,0)\}$ and constructed balanced Boolean functions satisfying the PC of degree $n-1$. For even

$n \geqslant 4$, they proposed a method for constructing balanced Boolean functions with n variables satisfying the PC with respect to all but three elements in $\{0,1\}^n - \{(0,\ldots,0)\}$ and constructed balanced Boolean functions satisfying the PC of degree about $2n/3$. This result is optimal in the sense that, for even $n \geqslant 4$, Boolean functions with n variables satisfying the PC with respect to all but less than three elements in $\{0,1\}^n - \{(0,\ldots,0)\}$ are perfectly nonlinear and that perfectly nonlinear Boolean functions are not balanced.

This chapter shows that, for every odd $n \geqslant 3$, all Boolean functions with n variables that satisfy the PC with respect to all but one elements in $\{0,1\}^n - \{(0,\ldots,0)\}$ are constructed from all perfectly nonlinear Boolean functions with $n-1$ variables. It also presents, for every even $n \geqslant 2$, a necessary and sufficient condition for a Boolean function to satisfy the PC with respect to all but three linearly dependent elements in $\{0,1\}^n - \{(0,\ldots,0)\}$. It shows that, for every even $n \geqslant 4$, all Boolean functions with n variables that satisfy the PC with respect to all but three linearly dependent elements in $\{0,1\}^n - \{(0,\ldots,0)\}$ are constructed from all perfectly nonlinear Boolean functions with $n-2$ variables.

Third, this chapter discusses Boolean functions with n variables satisfying the PC of degree $n-2$. It shows that, for every even $n \geqslant 4$, Boolean functions with n variables satisfying the PC of degree $n-2$ are perfectly nonlinear, and that, for every odd $n \geqslant 3$, they satisfy the PC with respect to all but one elements in $\{0,1\}^n - \{(0,\ldots,0)\}$.

Finally, some relationships between the PC and the SAC are presented. It is apparent from the definition that the set of Boolean functions that satisfy the PC of degree 1 coincides with that of Boolean functions that satisfy the SAC of order 0. It has been shown that the Boolean functions that satisfy the SAC of order $n-2$ are perfectly nonlinear[AT90].

This chapter shows, for every odd $n \geqslant 3$, that Boolean functions with n variables that satisfy the PC of degree $n-1$ satisfy the SAC of order 1, while those satisfying the PC of degree $n-2$ necessarily not and that there exist Boolean functions with n variables satisfying the SAC of order 2 and not satisfying the PC of degree $n-1$. For every even $n \geqslant 2$, it shows that perfectly nonlinear Boolean functions with n variables do not necessarily satisfy the SAC of order 1. It also shows

that Boolean functions with n variables that satisfy the SAC of order $n-3$ do not necessarily satisfy the PC of degree 2 for every $n \geqslant 3$.

Section 2.2 contains the definitions of nonlinearity criteria. Section 2.3 is devoted to the discussion of Boolean functions with n variables satisfying the PC with respect to all but one elements in $\{0,1\}^n - \{(0,\ldots,0)\}$, and those satisfying the PC with respect to all but linearly independent elements in $\{0,1\}^n - \{(0,\ldots,0)\}$. Section 2.4 discusses the construction of Boolean functions satisfying the PC with respect to all but one or all but three elements in $\{0,1\}^n - \{(0,\ldots,0)\}$. Section 2.5 discusses Boolean functions with n variables satisfying the PC of degree $n-2$. Section 2.6 shows the relationships between the PC and the SAC.

2.2 Preliminaries

2.2.1 Walsh Transform and Boolean Functions

Let $\mathbf{R}$ and $\mathbf{N}$ denote the set of reals and the set of integers, respectively.

Definition 2.1 The Walsh transform of a real-valued function $f : \{0,1\}^n \to \mathbf{R}$ is

$$(\mathcal{W}(f))(\omega) = \sum_{x \in \{0,1\}^n} f(x)(-1)^{\omega \cdot x},$$

where $x = (x_1,\ldots,x_n)$, $\omega = (\omega_1,\ldots,\omega_n) \in \{0,1\}^n$ and $\omega \cdot x$ denotes the dot product $\omega_1 x_1 \oplus \cdots \oplus \omega_n x_n$. □

For simplicity, $(\mathcal{W}(f))(\omega)$ is often denoted by $F(\omega)$. The inverse Walsh transform is

$$f(x) = (\mathcal{W}^{-1}(F))(x) = \frac{1}{2^n} \sum_{\omega \in \{0,1\}^n} F(\omega)(-1)^{\omega \cdot x}.$$

The Walsh transform can be represented in a matrix form[Rue91]. For $f : \{0,1\}^n \to \mathbf{R}$, let $f(i)$ denote $f(x_1,\ldots,x_n)$ when $x_1 + x_2 2 + \cdots + x_n 2^{n-1} = i$. Let $[f] = [f(0), f(1), \ldots, f(2^n-1)]$ and $[F] = [F(0), F(1), \ldots, F(2^n-1)]$. The Walsh transform is represented as

$$[F] = [f]H_n,$$

where H_n denotes the Hadamard matrix of order n. H_n is defined recursively by

$$\begin{aligned} H_0 &= [1], \\ H_n &= \begin{bmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{bmatrix}. \end{aligned}$$

H_n is a $2^n \times 2^n$ symmetric non-singular matrix, and its inverse is $2^{-n}H_n$. The inverse Walsh transform is represented as

$$[f] = 2^{-n}[F]H_n.$$

A Boolean function is a function of the form $f : \{0,1\}^n \to \{0,1\}^m$. $f : \{0,1\}^n \to \{0,1\}^m$ is called Boolean function with n inputs and m outputs. Let $\mathrm{B}_{n,m} = \{f \mid f : \{0,1\}^n \to \{0,1\}^m\}$. For simplicity, we denote $\mathrm{B}_{n,1}$ as B_n and call an Boolean function with n inputs and 1 output Boolean function with n inputs. Boolean functions with n inputs are also called Boolean functions with n variables.

A normal form of representation is defined for Boolean functions with n variables. Let $\mathrm{N} = \{1, \ldots, n\}$.

Definition 2.2 The algebraic normal form of a Boolean function $f \in \mathrm{B}_n$ is a type of representation of f such that

$$\bigoplus_{\{i_1,\ldots,i_k\} \in \wp(\mathrm{N})} a_{\{i_1,\ldots,i_k\}} x_{i_1} \cdots x_{i_k},$$

where $\wp(\mathrm{N})$ is the power set of N, and $a_{\{i_1,\ldots,i_k\}} \in \{0,1\}$ for every $\{i_1, \ldots, i_k\} \in \wp(\mathrm{N})$. □

Every Boolean function can be uniquely represented in an algebraic normal form, and any two different Boolean functions cannot be represented in a same algebraic normal form.

The Walsh transform can be applied to Boolean functions in B_n when they are considered to be real-valued functions. For the analysis of Boolean functions, it is often convenient to work with $\hat{f} : \{0,1\}^n \to \{-1,1\}$, where $\hat{f}(x) \stackrel{\text{def}}{=} (-1)^{f(x)}$. The Walsh transform of $\hat{f}$ is

$$\hat{F}(\omega) = \sum_{x \in \{0,1\}^n} \hat{f}(x)(-1)^{\omega \cdot x} = \sum_{x \in \{0,1\}^n} (-1)^{f(x) \oplus \omega \cdot x}.$$

Definition 2.3 The autocorrelation function of a Boolean function $f : \{0,1\}^n \rightarrow \{0,1\}$ is $C_f : \{0,1\}^n \rightarrow \mathbf{N}$ such that

$$C_f(z) = \sum_{x \in \{0,1\}^n} \hat{f}(x)\hat{f}(x \oplus z),$$

where $x \oplus z$ denotes $(x_1 \oplus z_1, \ldots, x_n \oplus z_n)$. □

Proposition 2.1 shows a relationship between the autocorrelation function of f and the Walsh transform of $\hat{f}$. It states that the inverse Walsh transform of $\hat{F}^2$ is C_f.

Proposition 2.1 For any Boolean function f, $C_f = \mathcal{W}^{-1}(\hat{F}^2)$. □

Proposition 2.2 shows that the sum of $\hat{F}^2(\omega)$'s is constant for every Boolean function f.

Proposition 2.2 For any $f \in \mathrm{B}_n$, $\sum_{\omega \in \{0,1\}^n} \hat{F}^2(\omega) = 2^{2n}$. □

2.2.2 Nonlinearity Criteria for Boolean Functions

For a set S, let |S| denote the number of elements in S. Let $\mathrm{V}_n = \{0,1\}^n - \{(0,\ldots,0)\}$.

Definition 2.4 A Boolean function $f \in \mathrm{B}_n$ is balanced if and only if $|\{x|f(x) = 0\}| = |\{x|f(x) = 1\}| = 2^{n-1}$. □

An affine Boolean function $h \in \mathrm{B}_n$ is a Boolean function of the form of

$$h(x_1, \ldots, x_n) = \alpha_0 \oplus \alpha_1 x_1 \oplus \cdots \oplus \alpha_n x_n,$$

where $\alpha_i \in \{0,1\}$ for $0 \leqslant i \leqslant n$. The set of affine Boolean functions with n variables is denoted as A_n. The number of affine Boolean functions with n variables is 2^{n+1}.

The distance between two Boolean functions, f and g, with the same number of variables, is $d(f,g) = |\{x \mid f(x) \neq g(x)\}|$.

The nonlinearity of $f \in \mathrm{B}_n$ is the minimum distance between f and $h \in \mathrm{A}_n$.

Definition 2.5 The nonlinearity of $f \in \mathrm{B}_n$ is $\min_{h \in \mathrm{A}_n} d(f, h)$. □

The nonlinearity of $f \in \mathrm{B}_n$ can be represented with $\hat{F}$.

Proposition 2.3 The nonlinearity of $f \in \mathrm{B}_n$ is

$$2^{n-1} - \frac{1}{2} \max_{\omega \in \{0,1\}^n} \left| \hat{F}(\omega) \right|.$$

□

Webster and Tavares [WT86] defined the strict avalanche criterion for a design principle of substitution boxes of the DES. For any $a \in \{0,1\}^n$, let $W(a)$ denote the Hamming weight of a, that is, the number of 1's in a.

Definition 2.6 A Boolean function $f \in \mathrm{B}_n$ is said to satisfy the strict avalanche criterion(SAC) if and only if $f(x) \oplus f(x \oplus a)$ is balanced for any $a \in \{0,1\}^n$ such that $W(a) = 1$. □

For a Boolean function satisfying the SAC, any 1-bit change of inputs causes the change of the output with probability 1/2.

Let $f(x_1, \ldots, x_n) \in \mathrm{B}_n$. For any $i_1, \ldots, i_m$ such that $1 \leqslant i_1 < i_2 < \cdots < i_m \leqslant n$ and $b_1, \ldots, b_m \in \{0,1\}$, let $f \mid_{x_{i_1}=b_1, \ldots, x_{i_m}=b_m} \in \mathrm{B}_{n-m}$ denote the subfunction of f obtained by substituting $b_1, \ldots, b_m$ for $x_{i_1}, \ldots, x_{i_m}$, respectively.

Forré [For90] extended the notion of the SAC and defined the SAC of higher orders. The original definition by Forré was simplified by Lloyd [Llo91].

Definition 2.7 [Llo91] A Boolean function $f \in \mathrm{B}_n$ is said to satisfy the strict avalanche criterion of order m if and only if, for any $i_1, \ldots, i_m$ such that $1 \leqslant i_1 < i_2 < \cdots < i_m \leqslant n$ and $b_1, \ldots, b_m \in \{0,1\}$, $f \mid_{x_{i_1}=b_1, \ldots, x_{i_m}=b_m} \in \mathrm{B}_{n-m}$ satisfies the SAC. □

It is obvious from the definition that the original SAC of Definition 2.6 is equivalent to the SAC of order 0. The value of a function satisfying the SAC depends on all of its variables. Lloyd[Llo91] proved that the functions satisfying the SAC of order m also satisfy the SAC of order k for every k such that $0 \leqslant k < m$.

Let $SAC_n(m)$ denote the set of $f \in B_n$ satisfying the SAC of order m. It is apparent from Definition 2.6 that every $f \in B_0 \cup B_1$ does not satisfy the SAC. Thus, $SAC_n(n-1) = SAC_n(n) = \emptyset$ for every n.

Definition 2.8 [MS90] A Boolean function $f \in B_n$ is perfectly nonlinear if and only if $f(x) \oplus f(x \oplus a)$ is balanced for any $a \in \{0,1\}^n$ such that $1 \leqslant W(a) \leqslant n$. □

For a perfectly nonlinear Boolean function, any change of inputs causes the change of the output with probability 1/2.

The following proposition directly follows from the definition of the autocorrelation function and the perfect nonlinearity.

Proposition 2.4 Let $f \in B_n$. f is perfectly nonlinear if and only if $C_f(z) = 0$ for every $z \in V_n$. □

Meier and Staffelbach[MS90] proved that the set of perfectly nonlinear Boolean functions coincides with the set of Boolean bent functions defined by Rothaus[Rot76].

Definition 2.9 Let $f \in B_n$. f is defined to be a Boolean bent function if and only if $\left|\hat{F}(\omega)\right| = 2^{n/2}$ for every $\omega \in \{0,1\}^n$. □

Proposition 2.5 Let $f \in B_n$. f is perfectly nonlinear if and only if $\left|\hat{F}(\omega)\right| = 2^{n/2}$ for every $\omega \in \{0,1\}^n$. □

Preneel, et al.[PLLGV91] extended the notion of the perfect nonlinearity and defined the propagation criterion.

Definition 2.10 A Boolean function $f \in B_n$ is said to satisfy the propagation criterion(PC) of degree k if and only if $f(x) \oplus f(x \oplus a)$ is balanced for every $a \in \{0,1\}^n$ such that $1 \leqslant W(a) \leqslant k$. □

Let $PC_n(k)$ denote the set of Boolean functions with n variables satisfying the propagation criterion of degree k. $PC_n(n)$ is the set of perfectly nonlinear Boolean functions with n variables.

Definition 2.11 A Boolean function $f \in B_n$ is said to satisfy the propagation criterion(PC) with respect to $A \subseteq V_n$ if and only if $f(x) \oplus f(x \oplus a)$ is balanced for every $a \in A$. □

Proposition 2.6 Let $f \in B_n$ and $A \subseteq V_n$. f satisfies the PC with respect to A if and only if $C_f(z) = 0$ for every $z \in A$. □

2.3 Propagation Criterion of Boolean Functions

2.3.1 Propagation Criterion of Degree $n-1$

In this section, we investigate Boolean functions that satisfy the PC of degree $n-1$.

We begin by presenting a theorem that gives a necessary and sufficient condition for $f \in \mathrm{B}_n$ to satisfy the PC with respect to all but one elements in V_n. Before presenting the theorem, we prove two simple lemmas.

For $a = (a_1, \ldots, a_n) \in \{0,1\}^n$, let $dec(a) = a_1 + 2a_2 + \cdots + 2^{n-1}a_n$.

Lemma 2.1 Let $m \geqslant 0$ be an integer. The integers $x, y \geqslant 0$ satisfying the equation

$$x^2 + y^2 = 2^m$$

is,

- for even m, $x = 2^{m/2}$ and $y = 0$, or $x = 0$ and $y = 2^{m/2}$,
- for odd m, $x = y = 2^{(m-1)/2}$.

(Proof) If one of x and y is 0, then m is even and the other is $2^{m/2}$.

If we assume that $x \neq 0$ and $y \neq 0$, then, we can represent x and y as

$$x = 2^{e_x} q_x,\ y = 2^{e_y} q_y,$$

respectively, where $e_x \geqslant 0$, $e_y \geqslant 0$, and $q_x \geqslant 1$, $q_y \geqslant 1$ are odd. Without loss of generality, it can be assumed that $e_y \geqslant e_x \geqslant 0$. Then,

$$\begin{aligned} 2^{2e_x} {q_x}^2 + 2^{2e_y} {q_y}^2 &= 2^m \\ {q_x}^2 + 2^{2(e_y - e_x)} {q_y}^2 &= 2^{m-2e_x}. \end{aligned}$$

Since ${q_x}^2 + 2^{2(e_y-e_x)}{q_y}^2 \geqslant 2$, $m - 2e_x \geqslant 1$, which implies that ${q_x}^2 + 2^{2(e_y-e_x)}{q_y}^2$ is even. Thus, $e_y - e_x = 0$ since q_x and q_y are odd. For

$${q_x}^2 + {q_y}^2 = 2^{m-2e_x},$$

since ${q_x}^2 + {q_y}^2$ is a multiple of 2 but not of 4, $m - 2e_x = 1$. Hence, $e_x = e_y = (m-1)/2$ and $q_x = q_y = 1$. This implies m is odd and $x = y = 2^{(m-1)/2}$. □

Lemma 2.2 For every $f \in \mathrm{B}_n$,

$$\left[\hat{F}^2(0), \ldots, \hat{F}^2(2^n - 1)\right] = \left[C_f(0), \ldots, C_f(2^n - 1)\right] H_n$$

(Proof) This lemma directly follows from Proposition 2.1. □

The following theorem presents a necessary and sufficient condition for a Boolean function to satisfy the PC with respect to all but one elements in V_n.

For every $b = (b_1, \ldots, b_n) \in \{0,1\}^n$, let v_b denote the $(dec(b)+1)$-th column vector of H_n, and let $l_b(x_1, \ldots, x_n) = b_1 x_1 \oplus \cdots \oplus b_n x_n$.

Theorem 2.1 Let $b \in \mathrm{V}_n$. $f \in \mathrm{B}_n$ satisfies the PC with respect to $\mathrm{V}_n - \{b\}$ if and only if,

- for even $n \geqslant 2$, $\left|\hat{F}(\omega)\right| = 2^{n/2}$ for every $\omega \in \{0,1\}^n$,
- for odd $n \geqslant 3$,

$$\left|\hat{F}(\omega)\right| = \begin{cases} 2^{(n+1)/2} & \text{if } b \cdot \omega = 0 \\ 0 & \text{if } b \cdot \omega = 1, \end{cases}$$

or

$$\left|\hat{F}(\omega)\right| = \begin{cases} 2^{(n+1)/2} & \text{if } b \cdot \omega = 1 \\ 0 & \text{if } b \cdot \omega = 0. \end{cases}$$

(Proof) $f \in \mathrm{B}_n$ satisfies the PC with respect to $\mathrm{V}_n - \{b\}$ if and only if $C_f(a) = 0$ for every $a \in \mathrm{V}_n - \{b\}$. Thus, from Lemma 2.2, $\left[\hat{F}^2\right]$ can be represented as

$$\left[\hat{F}^2\right] = C_f(0) {v_0}^{\mathrm{T}} + C_f(b) {v_b}^{\mathrm{T}},$$

where ${v_0}^{\mathrm{T}}$ and ${v_b}^{\mathrm{T}}$ are the transposes of v_0 and v_b, respectively. Let $u_0 = ({v_0}^{\mathrm{T}} + {v_b}^{\mathrm{T}})/2$ and $u_1 = ({v_0}^{\mathrm{T}} - {v_b}^{\mathrm{T}})/2$. Then, $\left[\hat{F}^2\right]$ is able to be represented as

$$\left[\hat{F}^2\right] = c_0 u_0 + c_1 u_1,$$

where $c_0 = C_f(0) + C_f(b)$ and $c_1 = C_f(0) - C_f(b)$. Since

$$u_0 = [1 \oplus l_b(0), \ldots, 1 \oplus l_b(2^n - 1)]$$

and

$$u_1 = [l_b(0), \ldots, l_b(2^n - 1)],$$

$$\hat{F}^2(\omega) = \begin{cases} c_0 & \text{if } b \cdot \omega = 0, \\ c_1 & \text{if } b \cdot \omega = 1. \end{cases}$$

Let $\left|\hat{F}(\omega)\right| = \hat{F}_0$ for every ω such that $b \cdot \omega = 0$, and $\left|\hat{F}(\omega)\right| = \hat{F}_1$ for every ω such that $b \cdot \omega = 1$. Since $\sum_{\omega \in \{0,1\}^n} \hat{F}^2(\omega) = 2^{2n}$,

$$\hat{F}_0^2 + \hat{F}_1^2 = 2^{n+1}.$$

Hence, from Lemma 2.1,

- When n is even, $\hat{F}_0 = \hat{F}_1 = 2^{n/2}$.
- When n is odd, $\hat{F}_0 = 0$, $\hat{F}_1 = 2^{(n+1)/2}$, or $\hat{F}_0 = 2^{(n+1)/2}$, $\hat{F}_1 = 0$.

The theorem has been proved. □

Boolean functions in B_n satisfying the PC with respect to $\mathrm{V}_n - \{(1, \ldots, 1)\}$ are the ones satisfying the PC of degree $n - 1$. Thus, the following two corollaries are immediately derived from Theorem 2.1.

Corollary 2.1 For even $n \geqslant 2$, $\mathrm{PC}_n(n-1) = \mathrm{PC}_n(n)$. □

Corollary 2.2 For odd $n \geqslant 3$, $f \in \mathrm{PC}_n(n-1)$ if and only if,

$$\left|\hat{F}(\omega)\right| = \begin{cases} 2^{(n+1)/2} & \text{if } W(\omega) \text{ is even} \\ 0 & \text{if } W(\omega) \text{ is odd,} \end{cases}$$

or

$$\left|\hat{F}(\omega)\right| = \begin{cases} 2^{(n+1)/2} & \text{if } W(\omega) \text{ is odd} \\ 0 & \text{if } W(\omega) \text{ is even.} \end{cases}$$

□

Corollary 2.3 Let $n \geqslant 3$ be odd and $b \in \mathrm{V}_n$. If $f \in \mathrm{B}_n$ satisfies the PC with respect to $\mathrm{V}_n - \{b\}$, then

$$f(x) \oplus f(x \oplus b) \equiv 0 \text{ or } 1.$$

(Proof) For odd $n \geqslant 3$, if $f \in \mathrm{B}_n$ satisfies the PC with respect to $\mathrm{V}_n - \{b\}$, then, from the proof of Theorem 2.1,

$$\begin{aligned} C_f(0) + C_f(b) &= 2^{n+1} \\ C_f(0) - C_f(b) &= 0, \end{aligned}$$

or

$$\begin{aligned} C_f(0) + C_f(b) &= 0 \\ C_f(0) - C_f(b) &= 2^{n+1}. \end{aligned}$$

For the former case, $C_f(b) = 2^n$, and for the latter case $C_f(b) = -2^n$. $C_f(b) = 2^n$ and $C_f(b) = -2^n$ implies that $f(x) \oplus f(x \oplus b) \equiv 0$ and $f(x) \oplus f(x \oplus b) \equiv 1$, respectively. □

From Theorem 2.1 and Proposition 2.3, the following corollary can be derived immediately.

Corollary 2.4 Let $n \geqslant 3$ be odd. If $f \in \mathrm{B}_n$ satisfies the PC with respect to all but one elements in V_n, then the nonlinearity of f is $2^{n-1} - 2^{(n-1)/2}$. □

The above corollary states that, for every odd $n \geqslant 3$, the nonlinearity of $f \in \mathrm{B}_n$ which satisfies the PC with respect to all but one elements in V_n are high and uniquely determined.

The particular case of Corollary 2.4 is as follows.

Corollary 2.5 Let $n \geqslant 3$ be odd. If $f \in \mathrm{PC}_n(n-1)$, then the nonlinearity of f is $2^{n-1} - 2^{(n-1)/2}$. □

2.3.2 Propagation Criterion with Respect to All or All but One Nonzero Elements

This section is devoted to a necessary and sufficient condition for a Boolean function in B_n to satisfy the PC with respect to all nonzero vectors for even n and with respect to all but one nonzero vectors for odd n.

Lemma 2.3 Let k be any integer such that $1 \leqslant k \leqslant n$ and $b_1, \ldots, b_k \in \{0,1\}^n$ be linearly independent. let $r_1, \ldots, r_k \in \{0,1\}$. The number of elements in $\{0,1\}^n$ satisfying

$$\begin{cases} l_{b_1}(x_1, \ldots, x_n) &= r_1 \\ \quad\vdots & \\ l_{b_k}(x_1, \ldots, x_n) &= r_k \end{cases}$$

are 2^{n-k}. □

Theorem 2.2 Let n and k be any integers such that $n \geqslant 2$ and $1 \leqslant k \leqslant n$. Let $b_1, \ldots, b_k \in \{0,1\}^n$ be linearly independent. If $f \in B_n$ satisfies the PC with respect to $V_n - \{b_1, \ldots, b_k\}$, then,

1. when n is even, $f \in PC_n(n)$,
2. when n is odd, for some i such that $1 \leqslant i \leqslant k$, f satisfies the PC with respect to $V_n - \{b_i\}$.

(Proof) From Proposition 2.6, f satisfies the PC with respect to $V_n - \{b_1, \ldots, b_k\}$ if and only if $C_f(a) = 0$ for every $a \in V_n - \{b_1, \ldots, b_k\}$. Thus from Lemma 2.2, $\left[\hat{F}^2\right]$ can be represented as

$$\left[\hat{F}^2\right] = C_f(0){v_0}^{\mathrm{T}} + C_f(b_1){v_{b_1}}^{\mathrm{T}} + \cdots + C_f(b_k){v_{b_k}}^{\mathrm{T}},$$

Let $u_0 = {v_0}^{\mathrm{T}}$ and $u_i = ({v_0}^{\mathrm{T}} + {v_{b_i}}^{\mathrm{T}})/2$ for every i such that $1 \leqslant i \leqslant k$, then we can rewrite $\left[\hat{F}^2\right]$ as

$$\left[\hat{F}^2\right] = c_0u_0 + c_1u_1 + \cdots + c_ku_k,$$

where

$$\begin{aligned} c_0 &= C_f(0) - \sum_{i=1}^{k} C_f(b_i), \\ c_i &= 2C_f(b_i). \end{aligned}$$

Since

$$\begin{aligned} u_0 &= [1, \ldots, 1], \\ u_i &= [1 \oplus l_{b_k}(0), \ldots, 1 \oplus l_{b_k}(2^n - 1)] \text{ for } 1 \leqslant i \leqslant k, \end{aligned}$$

and l_{b_i} is balanced for every $b_i \in \mathrm{V}_n$,

$$\sum_{\omega \in \{0,1\}^n} \hat{F}^2(\omega) = 2^n c_0 + 2^{n-1}(c_1 + \cdots + c_k) = 2^{2n}.$$

Thus,

$$2c_0 + c_1 + \cdots + c_k = 2^{n+1}.$$

From Lemma 2.3, there exist some $\omega \in \{0,1\}^n$ such that

$$\hat{F}^2(\omega) = c_0.$$

There also exist some $\omega \in \{0,1\}^n$ such that, for any j such that $1 \leqslant j \leqslant k$ and $i_1, \ldots, i_j$ such that $1 \leqslant i_1 < \cdots < i_j \leqslant k$,

$$\hat{F}^2(\omega) = c_0 + c_{i_1} + \cdots + c_{i_j}.$$

For the case where n is even. Since $2c_0 + c_1 + \cdots + c_k = 2^{n+1}$,

$$\begin{aligned} c_0 + (c_0 + c_1 + \ldots + c_k) &= 2^{n+1} \\ (c_0 + c_1) + (c_0 + c_2 + \ldots + c_k) &= 2^{n+1} \\ &\cdots \\ (c_0 + c_k) + (c_0 + c_1 + \ldots + c_{k-1}) &= 2^{n+1}, \end{aligned}$$

from Lemma 2.1,

$$c_0 = c_0 + c_1 = \cdots = c_0 + c_k = 2^n,$$

Thus,

$$c_0 = 2^n, c_1 = \cdots = c_k = 0.$$

Hence, for every $\omega \in \{0,1\}^n$,

$$\left|\hat{F}(\omega)\right| = 2^{n/2}.$$

For the case where n is odd. From Lemma 2.1,

$$c_0 = 0 \text{ or } 2^{n+1},$$

and, for any j such that $1 \leqslant j \leqslant k$ and $i_1, \ldots, i_j$ such that $1 \leqslant i_1 < \cdots < i_j \leqslant k$,

$$c_0 + c_{i_1} + \cdots + c_{i_j} = 0 \text{ or } 2^{n+1}.$$

(i) If we assume that $c_0 = 0$, then

$$c_0 + c_1 + \cdots + c_k = 2^{n+1}.$$

Since $c_0 + c_i = 0$ or 2^{n+1} for every i such that $1 \leqslant i \leqslant k$,

$$c_i = 0 \text{ or } 2^{n+1}.$$

Thus, only any one of $c_1, \ldots, c_k$ is 2^{n+1} and the others are all 0. Hence, for some b_i,

$$\left|\hat{F}(\omega)\right| = \begin{cases} 2^{(n+1)/2} & \text{if } b_i \cdot \omega = 0 \\ 0 & \text{if } b_i \cdot \omega = 1. \end{cases}$$

(ii) If we assume that $c_0 = 2^{n+1}$, then

$$c_0 + c_1 + \cdots + c_k = 0.$$

Since $c_0 + c_i = 0$ or 2^{n+1} for every i such that $1 \leqslant i \leqslant k$,

$$c_i = 0 \text{ or } -2^{n+1}.$$

Thus, only any one of $c_1, \ldots, c_k$ is -2^{n+1} and the others are all 0. Hence, for some b_i,

$$\left|\hat{F}(\omega)\right| = \begin{cases} 2^{(n+1)/2} & \text{if } b_i \cdot \omega = 1 \\ 0 & \text{if } b_i \cdot \omega = 0. \end{cases}$$

Hence, the theorem has been proved. □

2.4 Construction of Boolean Functions Satisfying the PC

This section gives an exact characterization of Boolean functions with an odd number of inputs that satisfy the PC with respect to all but one nonzero vectors. It also gives an exact characterization of Boolean functions with an even number of variables that satisfy the PC with respect to all but three nonzero vectors. The motivation of this research is a method in [SZZ93] to construct balanced Boolean functions with an odd number of variables that satisfy the PC with respect to all but one nonzero vectors and that to construct balanced Boolean functions with an even number of variables that satisfy the PC with respect to all but three nonzero vectors.

2.4.1 Boolean Functions with an Odd Number of Variables

This section presents, for odd $n \geqslant 3$, a spectral property of Boolean functions in B_n that satisfy the PC with respect to all but one elements in V_n. This is an exact characterization of such Boolean functions.

Seberry, et al.[SZZ93] presented a simple method that, for any odd $n \geqslant 3$, generates balanced Boolean functions in B_n satisfying the PC with respect to all but one elements in V_n from Boolean functions in $PC_{n-1}(n-1)$.

In this section, it is shown that, for every odd $n \geqslant 3$, one can construct all Boolean functions that satisfy the PC with respect to all but one elements in V_n from all Boolean functions in $PC_{n-1}(n-1)$. It also gives a construction method that is slightly different from the method of Seberry, et al. and that reflects spectral properties. Some results are presented for the number of Boolean functions satisfying the PC with respect to all but one nonzero vectors.

A lemma is proved which is a basis of the following discussion. It states that, for any $a \in V_n$, for each column v of the matrix constructed from i-th rows of H_n such that the dot product of a and the binary representation of i is equal to 0 or 1, there exists a column in H_{n-1} that is equal to v or $-v$.

We define some notations. For a matrix M, let $col(M, i)$ be the i-th column of M. For $a = (a_1, \ldots, a_n)$ and $1 \leqslant i \leqslant n$, let $\langle a \rangle_i$ denotes $(a_1, \ldots, a_i)$.

Lemma 2.4 For every $a \in \mathrm{V}_n$, let $K_n(a, 0)$ and $K_n(a, 1)$ be $2^{n-1} \times 2^n$ matrices that are constructed by removing all $(dec(\omega) + 1)$-th rows of H_n, where $a \cdot \omega = 1$ and $a \cdot \omega = 0$, respectively. Then,

- for each column v of H_{n-1}, $K_n(a, 0)$ has two columns that is equal to v, and $K_n(a, 1)$ has v and $-v$,
- for every i such that $1 \leqslant i \leqslant 2^n$,

$$col(K_n(a, 0), i) = col(K_n(a, 1), i)$$

or

$$col(K_n(a, 0), i) = -col(K_n(a, 1), i).$$

(Proof) We prove the theorem by induction. When $n = 1$, since $H_0 = [1]$ and

$$H_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix},$$

$K_1(1, 0) = [1, 1]$ and $K_1(1, 1) = [1, -1]$. The theorem is proved for $n = 1$.

For $n \geqslant 2$, we consider the following two cases: One is the case where $a_n = 0$ and the other is the case where $a_n = 1$.

For the case where $a_n = 0$. Since

$$a \cdot (\omega_1, \ldots, \omega_{n-1}, 0) = a \cdot (\omega_1, \ldots, \omega_{n-1}, 1)$$

and

$$H_n = \begin{bmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{bmatrix},$$

$$K_n(a, c) = \begin{bmatrix} K_{n-1}(\langle a \rangle_{n-1}, c) & K_{n-1}(\langle a \rangle_{n-1}, c) \\ K_{n-1}(\langle a \rangle_{n-1}, c) & -K_{n-1}(\langle a \rangle_{n-1}, c) \end{bmatrix}$$

for $c = 0, 1$. When $c = 0$, from the inductive assumption, for every column of H_{n-2}, $K_{n-1}(\langle a \rangle_{n-1}, 0)$ has exactly two columns which are equal to it. Thus, by permuting the columns of $K_n(a, 0)$,

$$\begin{bmatrix} H_{n-2} & H_{n-2} & H_{n-2} & H_{n-2} \\ H_{n-2} & H_{n-2} & -H_{n-2} & -H_{n-2} \end{bmatrix}.$$

is obtained. This implies that, for each column of H_{n-1}, $K_n(a, 0)$ has exactly two columns which are equal to it.

When $c = 1$, for every column v' of H_{n-2}, $K_{n-1}(\langle a \rangle_{n-1}, 1)$ has v' and $-v'$. Thus, for each column v of H_{n-1}, $K_n(a, 1)$ has v and $-v$.

It is also easily derived from the inductive assumption that, for every i such that $1 \leqslant i \leqslant 2^n$,

$$col(K_n(a, 0), i) = \pm col(K_n(a, 1), i).$$

For the case where $a_n = 1$. If $a = (0, \ldots, 0, 1)$, then

$$\begin{aligned} K_n(a, 0) &= \begin{bmatrix} H_{n-1} & H_{n-1} \end{bmatrix}, \\ K_n(a, 1) &= \begin{bmatrix} H_{n-1} & -H_{n-1} \end{bmatrix}. \end{aligned}$$

It is apparent that the theorem holds for this case.

If $a \neq (0, \ldots, 0, 1)$, since

$$a \cdot (\omega_1, \ldots, \omega_{n-1}, 0) = a \cdot (\omega_1, \ldots, \omega_{n-1}, 1) \oplus 1,$$

$$K_n(a, c) = \begin{bmatrix} K_{n-1}(\langle a \rangle_{n-1}, c) & K_{n-1}(\langle a \rangle_{n-1}, c) \\ K_{n-1}(\langle a \rangle_{n-1}, 1 \oplus c) & -K_{n-1}(\langle a \rangle_{n-1}, 1 \oplus c) \end{bmatrix}$$

for $c = 0, 1$. Since, for every j such that $1 \leqslant j \leqslant 2^{n-1}$,

$$col(K_{n-1}(\langle a \rangle_{n-1}, 0), j) = \pm col(K_{n-1}(\langle a \rangle_{n-1}, 1), j),$$

there exists some $2^n \times 2^n$ non-singular matrix Π such that

$$K_n(a, c)\ \Pi = \begin{bmatrix} K_{n-1}(\langle a \rangle_{n-1}, c) & K_{n-1}(\langle a \rangle_{n-1}, c) \\ K_{n-1}(\langle a \rangle_{n-1}, c) & -K_{n-1}(\langle a \rangle_{n-1}, c) \end{bmatrix}.$$

Π is a matrix that exchanges l-th and $(l + 2^{n-1})$-th columns of $K_n(a, c)$ for every l such that $1 \leqslant l \leqslant 2^{n-1}$ and

$$col(K_{n-1}(a, 1 \oplus c), l) = -col(K_{n-1}(a, c), l).$$

This is the same case as the one where $a_n = 0$. Hence, the theorem has been proved. □

An example of Lemma 2.4 is given below.

Example 2.1 Let $n = 4$ and $a = (0,1,0,1)$. Let $H_4 = [v_1^4, \ldots, v_{16}^4]$ and $H_3 = [v_1, \ldots, v_8]$. Then,

$$\begin{aligned} K_4(a,0) &= \left[\begin{array}{cccccccc} v_1^4 & v_3^4 & v_6^4 & v_8^4 & v_{10}^4 & v_{12}^4 & v_{13}^4 & v_{15}^4 \end{array} \right]^{\mathrm{T}} \\ &= \left[\begin{array}{cccccccc} v_1 & v_2 & v_5 & v_6 & v_3 & v_4 & v_7 & v_8 \\ v_5 & v_6 & v_1 & v_2 & v_7 & v_8 & v_3 & v_4 \end{array} \right], \end{aligned}$$

$$\begin{aligned} K_4(a,1) &= \left[\begin{array}{cccccccc} v_2^4 & v_4^4 & v_5^4 & v_7^4 & v_9^4 & v_{11}^4 & v_{14}^4 & v_{16}^4 \end{array} \right]^{\mathrm{T}} \\ &= \left[\begin{array}{cccccccc} v_1 & v_2 & -v_5 & -v_6 & v_3 & v_4 & -v_7 & -v_8 \\ v_5 & v_6 & -v_1 & -v_2 & v_7 & v_8 & -v_3 & -v_4 \end{array} \right]. \end{aligned}$$

For each column v_i of H_3, $K_4(a,0)$ has two columns that are equal to v_i, and $K_4(a,1)$ has a column that is equal to v_i and a column that is equal to $-v_i$.

$$col(K_4(a,0),i) = \begin{cases} col(K_4(a,1),i) & \text{for } i = 1,2,5,6,9,10,13,14 \\ -col(K_4(a,1),i) & \text{for } i = 3,4,7,8,11,12,15,16. \end{cases}$$

□

The following theorem implies an injective mapping from the set of Boolean functions in B_n that satisfy the PC with respect to all but one nonzero vectors to $\mathrm{PC}_{n-1}(n-1)$ for odd $n \geqslant 3$.

Theorem 2.3 Let $n \geqslant 3$ be odd. Let $f \in \mathrm{B}_n$ and $b \in \mathrm{V}_n$. Suppose f satisfies the PC with respect to $\mathrm{V}_n - \{b\}$. For $\alpha_1, \ldots, \alpha_{2^{n-1}} \in \{0,1\}^n$ such that $0 \leqslant dec(\alpha_1) < \cdots < dec(\alpha_{2^{n-1}}) \leqslant 2^n - 1$ and $\hat{F}(\alpha_i) \neq 0$ for $1 \leqslant i \leqslant 2^{n-1}$, let $f_W \in \mathrm{B}_{n-1}$ be defined as

$$\left[\hat{f}_W(0), \ldots, \hat{f}_W(2^{n-1}-1)\right] = \frac{1}{2^{\frac{n+1}{2}}} \left[\hat{F}(\alpha_1), \ldots, \hat{F}(\alpha_{2^{n-1}})\right].$$

Then, f_W is perfectly nonlinear.

(Proof) From the definition of the inverse Walsh transform,

$$\frac{1}{2^n}\left[\hat{F}(0),\ldots,\hat{F}(2^n-1)\right]H_n=\left[\hat{f}(0),\ldots,\hat{f}(2^n-1)\right].$$

Since f satisfies the PC with respect to $V_n-\{b\}$,

$$\left|\hat{F}(\omega)\right|=\begin{cases}2^{(n+1)/2} & \text{if } b\cdot\omega=0\\ 0 & \text{if } b\cdot\omega=1\end{cases}\quad\text{or}\quad\begin{cases}2^{(n+1)/2} & \text{if } b\cdot\omega=1\\ 0 & \text{if } b\cdot\omega=0.\end{cases}$$

Thus,

$$\begin{aligned}\frac{1}{2^n}\left[\hat{F}(\alpha_1),\ldots,\hat{F}(\alpha_{2^{n-1}})\right]K_n(b,c) &= \left[\hat{f}(0),\ldots,\hat{f}(2^n-1)\right]\\ \left[\hat{f}_W(0),\ldots,\hat{f}_W(2^{n-1}-1)\right]K_n(b,c) &= 2^{\frac{n-1}{2}}\left[\hat{f}(0),\ldots,\hat{f}(2^n-1)\right].\end{aligned}$$

where $c=0$ or $c=1$. From Lemma 2.4, for $K_n(b,c)$, there exists a non-singular $2^n\times 2^n$-matrix Π such that

$$K_n(b,c)\,\Pi=\left[\begin{array}{cc}H_{n-1} & (-1)^cH_{n-1}\end{array}\right]$$

Π exchanges columns of matrices when operated from the right of them. Hence,

$$\left[\hat{f}_W(0),\ldots,\hat{f}_W(2^{n-1}-1)\right]\left[\begin{array}{cc}H_{n-1} & (-1)^cH_{n-1}\end{array}\right]=2^{\frac{n-1}{2}}\left[\hat{f}(0),\ldots,\hat{f}(2^n-1)\right]\Pi.$$

This equation shows that, for every $\omega\in\{0,1\}^{n-1}$,

$$\left|\left(\mathcal{W}(\hat{f}_W)\right)(\omega)\right|=2^{\frac{n-1}{2}}.$$

This completes the proof. □

The following theorem states that the mapping in Theorem 2.3 is surjective.

Theorem 2.4 Let $n\geqslant 3$ be odd and $g\in B_{n-1}$. Let $\alpha_1,\ldots,\alpha_{2^{n-1}}\in\{0,1\}^n$, $b\in V_n$ and $c\in\{0,1\}$ such that $0\leqslant dec(\alpha_1)<\cdots<dec(\alpha_{2^{n-1}})\leqslant 2^n-1$ and $b\cdot\alpha_i=c$ for $1\leqslant i\leqslant 2^{n-1}$. Let $\hat{F}:\{0,1\}^n\to\mathbf{N}$ be defined as

$$\hat{F}(\omega)=\begin{cases}2^{(n+1)/2}\hat{g}(i-1) & \text{if } \omega=\alpha_i\\ 0 & \text{otherwise}\end{cases}$$

and $\hat{f} = (\mathcal{W}^{-1}(\hat{F}))$. If g is perfectly nonlinear, then $\hat{f} : \{0,1\}^n \rightarrow \{-1,1\}$ and f satisfies the PC with respect to $V_n - \{b\}$.

(Proof) Since $\hat{F}(\omega) = 0$ when $\omega \neq \alpha_i$ and $b \cdot \alpha_i = c$ for $1 \leqslant i \leqslant 2^{n-1}$,

$$\begin{aligned}\left[\hat{f}\right] &= \frac{1}{2^n}\left[\hat{F}(0), \ldots, \hat{F}(2^n - 1)\right] H_n \\ &= \frac{1}{2^n}\left[\hat{F}(\alpha_1), \ldots, \hat{F}(\alpha_{2^{n-1}})\right] K_n(b,c) \\ &= \frac{1}{2^{\frac{n-1}{2}}}\left[[\hat{g}(0), \ldots, \hat{g}(2^{n-1} - 1)\right] K_n(b,c)\end{aligned}$$

From Lemma 2.4, for $K_n(b,c)$, there exists a non-singular $2^n \times 2^n$-matrix Π such that

$$K_n(b,c)\ \Pi = \left[\ H_{n-1} \quad (-1)^c H_{n-1}\ \right].$$

Π exchanges columns of matrices when operated from the right of them. Hence,

$$\begin{aligned}\left[\hat{f}\right]\ \Pi &= \frac{1}{2^{\frac{n-1}{2}}}\left[\hat{g}(0), \ldots, \hat{g}(2^{n-1} - 1)\right]\left[\ H_{n-1} \quad (-1)^c H_{n-1}\ \right] \\ &= \frac{1}{2^{\frac{n-1}{2}}}\left[\ \hat{G} \quad (-1)^c \hat{G}\ \right].\end{aligned}$$

Since $\left|\hat{G}(\omega)\right| = 2^{\frac{n-1}{2}}$ for every $\omega \in \{0,1\}^{n-1}$, $\hat{f} : \{0,1\}^n \rightarrow \{-1,1\}$ and, from Theorem 2.1, f satisfies the PC with respect to $V_n - \{b\}$. □

From Theorem 2.3 and 2.4, it is obvious that the algorithm below generates all the Boolean functions in B_n that satisfy the PC with respect to all but one nonzero vectors from all the Boolean functions in $PC_{n-1}(n-1)$ for odd $n \geqslant 3$.

Algorithm 2.1

input $p \in PC_{n-1}(n-1)$, $b \in V_n$ for odd $n \geqslant 3$.

output $f_0, f_1 \in B_n$ that satisfy the PC with respect to $V_n - \{b\}$.

procedure

1. Let $c \in \{0,1\}$ and $\alpha_1^c, \ldots, \alpha_{2^{n-1}}^c \in \{0,1\}^n$ such that

$$0 \leqslant dec(\alpha_1^c) \leqslant \cdots \leqslant dec(\alpha_{2^{n-1}}^c) \leqslant 2^n - 1,$$

and, for $1 \leqslant i \leqslant 2^{n-1}$,

$$b \cdot \alpha_i^c = c.$$

2. Let

$$\hat{F}_c(\omega) = \begin{cases} 2^{(n+1)/2}\hat{p}(i-1) & \text{if } \omega = \alpha_i^c \\ 0 & \text{otherwise,} \end{cases}$$

where $\hat{F}_c = \mathcal{W}(\hat{f}_c)$.

3. Let $\left[\hat{f}_c\right] = \frac{1}{2^n}\left[\hat{F}_c\right] H_n$.

□

For Algorithm 2.1, since

$$\hat{F}_1(0) = \sum_{x \in \{0,1\}^n} (-1)^{f_1(x)} = 0,$$

f_1 is balanced, and f_0 is not balanced since $\hat{F}_0(0) \neq 0$. For every $p \in \mathrm{PC}_{n-1}(n-1)$ and $b \in \mathrm{V}_n$, let $Alg_n(p,b)$ denotes the set of the Boolean functions obtained by the above algorithm, which satisfy the PC with respect to $\mathrm{V}_n - \{b\}$. Since H_n is non-singular, for any different pairs (p,b) and (p',b'), $Alg_n(p,b) \cap Alg_n(p',b') = \emptyset$. Thus the following corollary can be obtained.

Corollary 2.6 For every odd $n \geqslant 3$, the number of Boolean functions in B_n which satisfy the PC with respect to all but one elements in V_n is $2(2^n - 1)|\mathrm{PC}_{n-1}(n-1)|$, and the half of them are balanced. □

In particular, for the Boolean functions with n variables satisfying the PC of degree $n-1$, the following corollary is derived.

Corollary 2.7 For every odd $n \geqslant 3$,

- $|\mathrm{PC}_n(n-1)| = 2|\mathrm{PC}_{n-1}(n-1)|$,
- the number of balanced functions in $\mathrm{PC}_n(n-1)$ is $|\mathrm{PC}_{n-1}(n-1)|$.

□

2.4.2 Boolean Functions with an Even Number of Variables

Theorem 2.2 says that, for even $n \geqslant 2$, Boolean functions which satisfy the PC with respect to all but one or two nonzero vectors are perfectly nonlinear, because less than three different nonzero vectors are always linearly independent.

Seberry, et al.[SZZ93] presented a method for constructing balanced Boolean functions satisfying the PC with respect to all but three elements in V_n for every even $n \geqslant 4$. Their result is optimal in the sense that there exist no balanced Boolean functions which satisfy the PC with respect to all but less than three nonzero vectors. Perfectly nonlinear Boolean functions are not balanced.

Proposition 2.7 [SZZ93] Let $n \geqslant 4$ be even. For any pair of $b_1, b_2 \in V_n$ such that $b_1 \neq b_2$, there exist balanced Boolean functions in B_n satisfying the PC with respect to $V_n - \{b_1, b_2, b_1 \oplus b_2\}$. □

In this section, for even $n \geqslant 4$, an exact characterization is presented of Boolean functions in B_n satisfying the PC with respect to all but three linearly dependent elements in V_n. A method of construction of such Boolean functions are also presented, and some relationships between the number of them and that of perfectly nonlinear Boolean functions are given.

First, we present two simple lemmas.

Lemma 2.5 There exist no positive integers x, y, z and m such that $x^2 + y^2 + z^2 = 2^m$.

(Proof) Suppose that x, y, z are positive integers. Then, x, y, z can be represented as

$$x = 2^{e_1} q_1,\ y = 2^{e_2} q_2,\ z = 2^{e_3} q_3,$$

where $e_1, e_2, e_3 \geqslant 0$, and q_1, q_2, q_3 are odd integers. Without loss of generality, it may be assumed that $0 \leqslant e_1 \leqslant e_2 \leqslant e_3$. If $x^2 + y^2 + z^2 = 2^m$, then

$$\begin{aligned} 2^{2e_1} {q_1}^2 + 2^{2e_2} {q_2}^2 + 2^{2e_3} {q_3}^2 &= 2^m \\ {q_1}^2 + 2^{2(e_2 - e_1)} {q_2}^2 + 2^{2(e_3 - e_1)} {q_3}^2 &= 2^{m - 2e_1}. \end{aligned}$$

Since the left-hand side of the above equation is greater than 3, $m - 2e_1 \geqslant 2$, which implies that the left-hand side is even. Thus, $e_2 - e_1 = 0$ and $e_3 - e_1 \geqslant 1$. Then,

$$q_1{}^2 + q_2{}^2 = 2^{m-2e_1} - 2^{2(e_3-e_1)}q_3{}^2.$$

Since both of q_1 and q_2 are odd, $q_1{}^2 + q_2{}^2$ is a multiple of 2 but not of 4. This contradicts that $m - 2e_1 \geqslant 2$ and $2(e_3 - e_1) \geqslant 2$. Hence, the lemma has been proved. □

Lemma 2.6 Let w, x, y, z and m be positive integers. $w^2 + x^2 + y^2 + z^2 = 2^m$ if and only if m is even and $w = x = y = z = 2^{(m-2)/2}$.

(Proof) Suppose that w, x, y, z are positive integers. Then, they are able to be represented as

$$w = 2^{e_1}q_1,\ x = 2^{e_2}q_2,\ y = 2^{e_3}q_3,\ z = 2^{e_4}q_4,$$

where $e_1, e_2, e_3, e_4 \geqslant 0$, and q_1, q_2, q_3, q_4 are odd integers. Without loss of generality, it may be assumed that $0 \leqslant e_1 \leqslant e_2 \leqslant e_3 \leqslant e_4$. Since $w^2 + x^2 + y^2 + z^2 = 2^m$,

$$\begin{aligned} 2^{2e_1}q_1{}^2 + 2^{2e_2}q_2{}^2 + 2^{2e_3}q_3{}^2 + 2^{2e_4}q_4{}^2 &= 2^m \\ q_1{}^2 + 2^{2(e_2-e_1)}q_2{}^2 + 2^{2(e_3-e_1)}q_3{}^2 + 2^{2(e_4-e_1)}q_4{}^2 &= 2^{m-2e_1}. \end{aligned}$$

Since the left-hand side of the above equation is greater than 4, $m - 2e_1 \geqslant 2$. Since the left-hand side is even, $e_2 - e_1 = 0$. Thus,

$$q_1{}^2 + q_2{}^2 + 2^{2(e_3-e_1)}q_3{}^2 + 2^{2(e_4-e_1)}q_4{}^2 = 2^{m-2e_1}.$$

Since $q_1{}^2 + q_2{}^2$ is a multiple of 2 but not of 4 and 2^{m-2e_1} is a multiple of 4, $e_3 - e_1 = e_4 - e_1 = 0$ and

$$q_1{}^2 + q_2{}^2 + q_3{}^2 + q_4{}^2 = 2^{m-2e_1}.$$

For $i = 1, 2, 3, 4$, q_i can be represented as $q_i = 2r_i + 1$, where $r_i \geqslant 0$ is an integer. Hence,

$$4\left(\sum_{i=1}^{4} r_i(r_i + 1) + 1\right) = 2^{m-2e_1}.$$

Because $\sum_{i=1}^{4} r_i(r_i+1)+1$ is odd, $m-2e_1 = 2$ and $r_1 = r_2 = r_3 = r_4 = 0$. Hence, m is even and $w = x = y = z = 2^{(m-2)/2}$. □

The following theorem presents a necessary and sufficient condition for $f \in \mathrm{B}_n$ to satisfy the PC with respect to all but three linearly dependent elements in V_n for even $n \geqslant 4$.

Theorem 2.5 Let $n \geqslant 4$ be even and $f \in \mathrm{B}_n$. Let b_1, b_2, b_3 be different elements in V_n and be linearly dependent. $f \notin \mathrm{PC}_n(n)$ satisfies the PC with respect to $\mathrm{V}_n - \{b_1, b_2, b_3\}$ if and only if

$$\left|\hat{F}(\omega)\right| = \begin{cases} 2^{n/2+1} & \text{if } b_1 \cdot \omega = b_2 \cdot \omega = b_3 \cdot \omega = 0 \\ 0 & \text{otherwise} \end{cases}$$

or

$$\left|\hat{F}(\omega)\right| = \begin{cases} 2^{n/2+1} & \text{if } b_i \cdot \omega = 0,\ b_j \cdot \omega = b_k \cdot \omega = 1 \\ & \text{for different } i,\ j,\ k \\ 0 & \text{otherwise.} \end{cases}$$

(Proof) $f \in \mathrm{B}_n$ satisfies the PC with respect to $\mathrm{V}_n - \{b_1, b_2, b_3\}$ if and only if $C_f(a) = 0$ for every $a \in \mathrm{V}_n - \{b_1, b_2, b_3\}$. From Lemma 2.2, $\left[\hat{F}^2\right]$ can be represented as

$$\left[\hat{F}^2\right] = C_f(0){v_0}^{\mathrm{T}} + C_f(b_1){v_{b_1}}^{\mathrm{T}} + C_f(b_2){v_{b_2}}^{\mathrm{T}} + C_f(b_3){v_{b_3}}^{\mathrm{T}}.$$

Let $u_0 = {v_0}^{\mathrm{T}}$ and $u_i = ({v_0}^{\mathrm{T}} + {v_{b_i}}^{\mathrm{T}})/2$ for $i = 1, 2, 3$, then $\left[\hat{F}^2\right]$ can be written as

$$\left[\hat{F}^2\right] = c_0 u_0 + c_1 u_1 + c_2 u_2 + c_3 u_3,$$

where

$$\begin{aligned} c_0 &= C_f(0) - \left(C_f(b_1) + C_f(b_2) + C_f(b_3)\right), \\ c_i &= 2C_f(b_i). \end{aligned}$$

Since

$$\begin{aligned} u_0 &= [1, \ldots, 1], \\ u_i &= [1 \oplus l_{b_i}(0), \ldots, 1 \oplus l_{b_i}(2^n - 1)] \end{aligned}$$

for $i = 1, 2, 3$, and l_{b_i} is balanced for every $b_i \in \mathrm{V}_n$,

$$\sum_{\omega \in \{0,1\}^n} \hat{F}^2(\omega) = 2^n c_0 + 2^{n-1}(c_1 + c_2 + c_3) = 2^{2n}.$$

Thus,

$$(c_0 + c_1 + c_2 + c_3) + (c_0 + c_1) + (c_0 + c_2) + (c_0 + c_3) = 2^{n+2}.$$

Since

$$\hat{F}^2(\omega) = \begin{cases} c_0 + c_i & \text{if } b_i = 0 \text{ and } b_j \cdot \omega = 1 \text{ for} \\ & i = 1,2,3 \text{ and } j \in \{1,2,3\} - \{i\} \\ c_0 + c_1 + c_2 + c_3 & \text{if } b_i \cdot \omega = 0 \text{ for } i = 1,2,3, \end{cases}$$

from Lemma 2.1, 2.5, 2.6, there are following two cases:

C-1. $c_0 + c_1 + c_2 + c_3 = c_0 + c_1 = c_0 + c_2 = c_0 + c_3 = 2^n$,

C-2. only one of $c_0 + c_1 + c_2 + c_3$, $c_0 + c_1$, $c_0 + c_2$ and $c_0 + c_3$ is 2^{n+2} and the others are 0.

For C-1, $f \in \mathrm{PC}_n(n)$.

For C-2, if $c_0 + c_1 + c_2 + c_3 = 2^{n+2}$, then $\hat{F}^2(\omega) = 2^{n+2}$ when $b_1 \cdot \omega = b_2 \cdot \omega = b_3 \cdot \omega = 0$. If $c_0 + c_i = 2^{n+2}$, then $\hat{F}^2(\omega) = 2^{n+2}$ when $b_i \cdot \omega = 0$ and $b_j \cdot \omega = b_k \cdot \omega = 1$ for different i, j, k. Hence, the theorem has been proved. □

The next corollary can be proved in the same way as Corollary 2.3.

Corollary 2.8 Let $n \geqslant 4$ be even. Let b_1, b_2, b_3 be different elements in V_n and be linearly dependent. If $f \in \mathrm{B}_n - \mathrm{PC}_n(n)$ satisfies the PC with respect to $\mathrm{V}_n - \{b_1, b_2, b_3\}$, then, for each of $i \in \{1,2,3\}$,

$$f(x) \oplus f(x \oplus b_i) \equiv 0 \text{ or } 1.$$

(Proof) For even $n \geqslant 4$, if $f \in \mathrm{B}_n - \mathrm{PC}_n(n)$ satisfies the PC with respect to $\mathrm{V}_n - \{b_1, b_2, b_3\}$, then, from the proof of Theorem 2.5,

$$\begin{aligned} C_f(0) + C_f(b_1) + C_f(b_2) + C_f(b_3) &= 2^{n+2} \\ C_f(0) + C_f(b_1) - C_f(b_2) - C_f(b_3) &= 0 \\ C_f(0) - C_f(b_1) + C_f(b_2) - C_f(b_3) &= 0 \\ C_f(0) - C_f(b_1) - C_f(b_2) + C_f(b_3) &= 0, \end{aligned}$$

or, for different $i, j, k \in \{1, 2, 3\}$,

$$\begin{aligned} C_f(0) + C_f(b_1) + C_f(b_2) + C_f(b_3) &= 0 \\ C_f(0) + C_f(b_i) - C_f(b_j) - C_f(b_k) &= 2^{n+2} \\ C_f(0) - C_f(b_i) + C_f(b_j) - C_f(b_k) &= 0 \\ C_f(0) - C_f(b_i) - C_f(b_j) + C_f(b_k) &= 0. \end{aligned}$$

For the former case, $C_f(b_1) = C_f(b_2) = C_f(b_3) = 2^n$, and for the latter case, $C_f(b_i) = 2^n$ and $C_f(b_j) = C_f(b_k) = -2^n$. $C_f(b) = 2^n$ and $C_f(b) = -2^n$ implies that $f(x) \oplus f(x \oplus b) \equiv 0$ and $f(x) \oplus f(x \oplus b) \equiv 1$, respectively. □

The following corollary can be easily derived from Theorem 2.5. This presents a spectral property of the balanced Boolean functions satisfying the PC with respect to all but three elements in V_n for every even $n \geqslant 4$.

Corollary 2.9 Let $n \geqslant 4$ be even and $f \in B_n$. Let b_1, b_2, $b_3 \in V_n$ be different and linearly dependent. f is balanced and satisfies the PC with respect to $V_n - \{b_1, b_2, b_3\}$ if and only if

$$\left|\hat{F}(\omega)\right| = \begin{cases} 2^{n/2+1} & \text{if } b_i \cdot \omega = 0,\ b_j \cdot \omega = b_k \cdot \omega = 1 \text{ for different} \\ & i, j, k \in \{1, 2, 3\} \\ 0 & \text{otherwise.} \end{cases}$$

□

Corollary 2.10 Let $n \geqslant 4$ be even. The nonlinearity of any balanced Boolean function in B_n satisfying the PC with respect to all but three elements in V_n is $2^{n-1} - 2^{n/2}$.

(Proof) This corollary directly follows from Proposition 2.3 and Theorem 2.5. □

Seberry, et al.[SZZ93] proved that the nonlinearities of balanced Boolean functions satisfying the PC with respect to all but three nonzero vectors are at least $2^{n-1} - 2^{n/2}$. Corollary 2.10 determines the nonlinearity of balanced Boolean functions satisfying the PC with respect to

all but three nonzero vectors uniquely, and shows that the lower bound of the nonlinearity of Seberry, et al. is optimal.

In the following, it is shown that, for every even $n \geqslant 4$, one can construct all Boolean functions that satisfy the PC with respect to all but three linearly dependent vectors in V_n from all Boolean functions in $\mathrm{PC}_{n-2}(n-2)$.

A lemma is proved for the basis of the following discussion.

Lemma 2.7 Let $n \geqslant 2$, $a = (a_1, \ldots, a_n), b = (b_1, \ldots, b_n) \in \mathrm{V}_n$ such that $a \neq b$ and $c, d \in \{0, 1\}$. Let $K_n(a, c; b, d)$ be a $2^{n-2} \times 2^n$ matrix that is constructed by removing all $(dec(\omega) + 1)$-th rows of H_n, where $a \cdot \omega \neq c$ or $b \cdot \omega \neq d$. Then,

- for each column v of H_{n-2}, $K_n(a, c; b, d)$ has four columns that is equal to v if $c = d = 0$, and has two columns that is equal to v and two columns that is equal to $-v$ if $c \neq 0$ or $d \neq 0$,
- for every i such that $1 \leqslant i \leqslant 2^n$,

$$col(K_n(a, c; b, d), i) = col(K_n(a, c'; b, d'), i)$$

or

$$col(K_n(a, c; b, d), i) = -col(K_n(a, c'; b, d'), i),$$

and, for $\{(c_1, d_1), (c_2, d_2), (c_3, d_3), (c_4, d_4)\} = \{0, 1\}^2$,

$$\begin{array}{rcl} col(K_n(a, c_1; b, d_1), i) & = & col(K_n(a, c_2; b, d_2), i) \\ & \Updownarrow & \\ col(K_n(a, c_3; b, d_3), i) & = & col(K_n(a, c_4; b, d_4), i). \end{array}$$

(Proof) This lemma can be proved by induction.

For every (a, c) and (b, d), $K_n(a, c; b, d) = K_n(b, d; a, c)$. When $n = 2$, since

$$H_2 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix},$$

$$
\begin{aligned}
K_n((1,0),0;(0,1),0) &= \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix}, \\
K_n((1,0),0;(0,1),1) &= \begin{bmatrix} 1 & -1 & 1 & -1 \end{bmatrix}, \\
K_n((1,0),1;(0,1),0) &= \begin{bmatrix} 1 & 1 & -1 & -1 \end{bmatrix}, \\
K_n((1,0),1;(0,1),1) &= \begin{bmatrix} 1 & -1 & -1 & 1 \end{bmatrix},
\end{aligned}
$$

and, for $a = (1,0), b = (1,1)$ and $a = (0,1), b = (1,1)$,

$$
\begin{aligned}
K_n(a,0;b,0) &= \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix}, \\
K_n(a,1;b,0) &= \begin{bmatrix} 1 & -1 & 1 & -1 \end{bmatrix}, \\
K_n(a,0;b,1) &= \begin{bmatrix} 1 & -1 & -1 & 1 \end{bmatrix}, \\
K_n(a,1;b,1) &= \begin{bmatrix} 1 & 1 & -1 & -1 \end{bmatrix}.
\end{aligned}
$$

Thus, the theorem is proved for $n = 2$ because $H_0 = [1]$.

For simplicity, let $\langle a \rangle_{n-1} = \langle a \rangle$.

For $a_n = b_n = 0$. Since

$$
\begin{aligned}
a \cdot (\omega_1, \ldots, \omega_{n-1}, 0) &= a \cdot (\omega_1, \ldots, \omega_{n-1}, 1), \\
b \cdot (\omega_1, \ldots, \omega_{n-1}, 0) &= b \cdot (\omega_1, \ldots, \omega_{n-1}, 1),
\end{aligned}
$$

and

$$
H_n = \begin{bmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{bmatrix},
$$

$$
K_n(a,c;b,d) = \begin{bmatrix} K_{n-1}(\langle a \rangle, c; \langle b \rangle, d) & K_{n-1}(\langle a \rangle, c; \langle b \rangle, d) \\ K_{n-1}(\langle a \rangle, c; \langle b \rangle, d) & -K_{n-1}(\langle a \rangle, c; \langle b \rangle, d) \end{bmatrix}.
$$

From the inductive assumption, for every column v' of H_{n-3}, $K_{n-1}(\langle a \rangle, c; \langle b \rangle, d)$ has four columns that are equal to v' if $c = d = 0$, and has two columns that are equal to v and two columns that are equal to $-v'$ if $c = 1$ or $d = 1$. Thus, for every column v of H_{n-2}, $K_n(a,c;b,d)$ has four columns that are equal to v if $c = d = 0$, and has two columns that are equal to v and two columns that are equal to $-v$ if $c = 1$ or $d = 1$.

It is apparent from the inductive assumption that, for every i such that $1 \leqslant i \leqslant 2^n$,

$$col(K_n(a,c;b,d),i) = \pm col(K_n(a,c';b,d'),i).$$

It is also apparent that, for $\{(c_1,d_1),(c_2,d_2),(c_3,d_3),(c_4,d_4)\} = \{0,1\}^2$,

$$\begin{aligned} col(K_n(a,c_1;b,d_1),i) &= col(K_n(a,c_2;b,d_2),i) \\ &\Updownarrow \\ col(K_n(a,c_3;b,d_3),i) &= col(K_n(a,c_4;b,d_4),i). \end{aligned}$$

For $a_n = 1, b_n = 0$.

$$\begin{aligned} a\cdot(\omega_1,\ldots,\omega_{n-1},0) &= \langle a\rangle\cdot\langle\omega\rangle, \\ a\cdot(\omega_1,\ldots,\omega_{n-1},1) &= \langle a\rangle\cdot\langle\omega\rangle \oplus 1. \end{aligned}$$

(i) When $a = (0,\ldots,0,1)$,

$$\begin{aligned} K_n(a,0;b,0) &= \left[\begin{array}{cc} K_{n-1}(\langle b\rangle,0) & K_{n-1}(\langle b\rangle,0) \end{array}\right], \\ K_n(a,0;b,1) &= \left[\begin{array}{cc} K_{n-1}(\langle b\rangle,1) & K_{n-1}(\langle b\rangle,1) \end{array}\right], \\ K_n(a,1;b,0) &= \left[\begin{array}{cc} K_{n-1}(\langle b\rangle,0) & -K_{n-1}(\langle b\rangle,0) \end{array}\right], \\ K_n(a,1;b,1) &= \left[\begin{array}{cc} K_{n-1}(\langle b\rangle,1) & -K_{n-1}(\langle b\rangle,1) \end{array}\right]. \end{aligned}$$

From Lemma 2.4, for every column v of H_{n-2}, $K_{n-1}(\langle b\rangle,0)$ has two columns that are equal to v, and $K_{n-1}(\langle b\rangle,1)$ has v and $-v$. Thus, $K_n(a,c;b,d)$ has four columns equal to v if $c = d = 0$, and has two columns equal to v and two columns equal to $-v$ if $c = 1$ or $d = 1$.

Since, for every j such that $1 \leqslant j \leqslant 2^{n-1}$,

$$col(K_{n-1}(\langle b\rangle,0),j) = \pm col(K_{n-1}(\langle b\rangle,1),j),$$

for every i such that $1 \leqslant i \leqslant 2^n$,

$$col(K_n(a,c;b,d),i) = \pm col(K_n(a,c';b,d'),i),$$

and also, for $\{(c_1,d_1),(c_2,d_2),(c_3,d_3),(c_4,d_4)\} = \{0,1\}^2$,

$$\begin{aligned} col(K_n(a,c_1;b,d_1),i) &= col(K_n(a,c_2;b,d_2),i) \\ &\Updownarrow \\ col(K_n(a,c_3;b,d_3),i) &= col(K_n(a,c_4;b,d_4),i). \end{aligned}$$

(ii) When $a \neq (0,\ldots,0,1)$,

$$K_n(a,c;b,d) = \begin{bmatrix} K_{n-1}(\langle a\rangle, c; \langle b\rangle, d) & K_{n-1}(\langle a\rangle, c; \langle b\rangle, d) \\ K_{n-1}(\langle a\rangle, 1\oplus c; \langle b\rangle, d) & -K_{n-1}(\langle a\rangle, 1\oplus c; \langle b\rangle, d) \end{bmatrix}.$$

From the inductive assumption, for every i such that $1 \leqslant i \leqslant 2^{n-1}$,

$$col(K_{n-1}(\langle a\rangle, c; \langle b\rangle, d), i) = \pm col(K_{n-1}(\langle a\rangle, 1\oplus c; \langle b\rangle, d), i),$$

and, for every $(c_1,d_1),(c_2,d_2) \in \{0,1\}^2$ such that $(c_1,d_1) \neq (c_2,d_2)$,

$$\begin{array}{rcl} col(K_{n-1}(\langle a\rangle, c_1; \langle b\rangle, d_1), i) & = & col(K_{n-1}(\langle a\rangle, 1\oplus c_1; \langle b\rangle, d_1), i) \\ & \Updownarrow & \\ col(K_{n-1}(\langle a\rangle, c_2; \langle b\rangle, d_2), i) & = & col(K_{n-1}(\langle a\rangle, 1\oplus c_2; \langle b\rangle, d_2), i). \end{array}$$

Thus, there exists some $2^n \times 2^n$-matrix Π, which permutes the columns of matrices, such that, for every c and d,

$$K_n(a,c;b,d)\,\Pi = \begin{bmatrix} K_{n-1}(\langle a\rangle, c; \langle b\rangle, d) & K_{n-1}(\langle a\rangle, c; \langle b\rangle, d) \\ K_{n-1}(\langle a\rangle, c; \langle b\rangle, d) & -K_{n-1}(\langle a\rangle, c; \langle b\rangle, d) \end{bmatrix}.$$

Thus, this case can be proved in the same way as the case where $a_n = b_n = 0$.

For $a_n = b_n = 1$.

(i) When $a = (0,\ldots,0,1)$,

$$\begin{array}{rcl} K_n(a,0;b,0) & = & \begin{bmatrix} K_{n-1}(\langle b\rangle, 0) & K_{n-1}(\langle b\rangle, 0) \end{bmatrix}, \\ K_n(a,0;b,1) & = & \begin{bmatrix} K_{n-1}(\langle b\rangle, 1) & K_{n-1}(\langle b\rangle, 1) \end{bmatrix}, \\ K_n(a,1;b,0) & = & \begin{bmatrix} K_{n-1}(\langle b\rangle, 1) & -K_{n-1}(\langle b\rangle, 1) \end{bmatrix}, \\ K_n(a,1;b,1) & = & \begin{bmatrix} K_{n-1}(\langle b\rangle, 0) & -K_{n-1}(\langle b\rangle, 0) \end{bmatrix}. \end{array}$$

This case can be proved in the same way as the case where $a = (0,\ldots,0,1)$ and b such that $b_n \neq 0$.

(ii) When $a \neq (0,\ldots,0,1)$ and $b \neq (0,\ldots,0,1)$,

$$K_n(a,c;b,d) = \begin{bmatrix} K_{n-1}(\langle a\rangle, c; \langle b\rangle, d) & K_{n-1}(\langle a\rangle, c; \langle b\rangle, d) \\ K_{n-1}(\langle a\rangle, 1\oplus c; \langle b\rangle, 1\oplus d) & -K_{n-1}(\langle a\rangle, 1\oplus c; \langle b\rangle, 1\oplus d) \end{bmatrix}.$$

In the same way as for the above case, it can be shown that there exists some $2^n \times 2^n$-matrix Π, which permutes the columns of matrices, such that, for every c and d,

$$K_n(a,c;b,d)\,\Pi = \begin{bmatrix} K_{n-1}(\langle a\rangle, c; \langle b\rangle, d) & K_{n-1}(\langle a\rangle, c; \langle b\rangle, d) \\ K_{n-1}(\langle a\rangle, c; \langle b\rangle, d) & -K_{n-1}(\langle a\rangle, c; \langle b\rangle, d) \end{bmatrix}.$$

This case can be proved in the same way as the case where $a_n = b_n = 0$.

This completes the proof. □

Example 2.2 Let $n = 4$ and $a = (0,1,0,1)$, $b = (1,0,0,1)$. Let $H_4 = [v_1^4,\ldots,v_{16}^4]$ and $H_2 = [v_1, v_2, v_3, v_4]$. Then,

$$\begin{aligned} K_4(a,0;b,0) &= \begin{bmatrix} v_1^4 & v_5^4 & v_{12}^4 & v_{16}^4 \end{bmatrix}^{\mathrm{T}} \\ &= \begin{bmatrix} v_1 & v_3 & v_3 & v_1 & v_2 & v_4 & v_4 & v_2 \\ v_3 & v_1 & v_1 & v_3 & v_4 & v_2 & v_2 & v_4 \end{bmatrix} \end{aligned}$$

$$\begin{aligned} K_4(a,0;b,1) &= \begin{bmatrix} v_2^4 & v_6^4 & v_{11}^4 & v_{15}^4 \end{bmatrix}^{\mathrm{T}} \\ &= \begin{bmatrix} v_1 & -v_3 & v_3 & -v_1 & v_2 & -v_4 & v_4 & -v_2 \\ v_3 & -v_1 & v_1 & -v_3 & v_4 & -v_2 & v_2 & -v_4 \end{bmatrix} \end{aligned}$$

$$\begin{aligned} K_4(a,1;b,0) &= \begin{bmatrix} v_3^4 & v_7^4 & v_{10}^4 & v_{14}^4 \end{bmatrix}^{\mathrm{T}} \\ &= \begin{bmatrix} v_1 & v_3 & -v_3 & -v_1 & v_2 & v_4 & -v_4 & -v_2 \\ v_3 & v_1 & -v_1 & -v_3 & v_4 & v_2 & -v_2 & -v_4 \end{bmatrix} \end{aligned}$$

$$
\begin{aligned}
K_4(a,1;b,1) &= \left[\begin{array}{cccc} v_4^4 & v_8^4 & v_9^4 & v_{13}^4 \end{array}\right]^{\mathrm{T}} \\
&= \left[\begin{array}{cccccccc} v_1 & -v_3 & -v_3 & v_1 & v_2 & -v_4 & -v_4 & v_2 \\ v_3 & -v_1 & -v_1 & v_3 & v_4 & -v_2 & -v_2 & v_4 \end{array}\right]
\end{aligned}
$$

For each column v_i of H_2, $K_4(a,0;b,0)$ has four columns that are equal to v_i, and each of $K_4(a,0;b,1)$, $K_4(a,1;b,0)$ and $K_4(a,1;b,1)$ has two columns that are equal to v_i and two columns that are equal to $-v_i$. □

The following theorem implies an injective mapping from the set of Boolean functions in B_n that satisfy the PC with respect to all but three linearly dependent nonzero vectors to $\mathrm{PC}_{n-2}(n-2)$ for even $n \geqslant 4$.

Theorem 2.6 Let $n \geqslant 4$ be even. Let $f \in \mathrm{B}_n$ and $b_1, b_2, b_3 \in \mathrm{V}_n$ such that b_1, b_2, b_3 are different and linearly dependent. Suppose f satisfies the PC with respect to $\mathrm{V}_n - \{b_1, b_2, b_3\}$ and is not perfectly nonlinear. For $\alpha_1, \ldots, \alpha_{2^{n-2}} \in \{0,1\}^n$ such that $1 \leqslant dec(\alpha_1) < \cdots < dec(\alpha_{2^{n-2}}) \leqslant 2^n - 1$ and $\hat{F}(\alpha_i) \neq 0$ for $1 \leqslant i \leqslant 2^{n-2}$, let $f_W \in \mathrm{B}_{n-2}$ be defined as

$$
\left[\hat{f}_W(0), \ldots, \hat{f}_W(2^{n-2}-1)\right] = \frac{1}{2^{\frac{n}{2}+1}} \left[\hat{F}(\alpha_1), \ldots, \hat{F}(\alpha_{2^{n-2}})\right].
$$

Then f_W is perfectly nonlinear.

(Proof) Since f satisfies the PC with respect to $\mathrm{V}_n - \{b_1, b_2, b_3\}$ and is not perfectly nonlinear,

$$
\left|\hat{F}(\omega)\right| = \begin{cases} 2^{n/2+1} & \text{if } b_1 \cdot \omega = b_2 \cdot \omega = b_3 \cdot \omega = 0 \\ 0 & \text{otherwise} \end{cases}
$$

or

$$
\left|\hat{F}(\omega)\right| = \begin{cases} 2^{n/2+1} & \text{if } b_i \cdot \omega = 0,\ b_j \cdot \omega = b_k \cdot \omega = 1 \text{ for different} \\ & i,j,k \in \{1,2,3\} \\ 0 & \text{otherwise.} \end{cases}
$$

Without loss of generality, we can fix $i = 1$, $j = 2$, $k = 3$. Thus,

$$
\begin{aligned}
\frac{1}{2^n}\left[\hat{F}(0), \ldots, \hat{F}(2^n-1)\right] H_n &= \left[\hat{f}\right] \\
\frac{1}{2^n}\left[\hat{F}(\alpha_1), \ldots, \hat{F}(\alpha_{2^{n-2}})\right] K_n(b_1,0;b_2,c) &= \left[\hat{f}\right] \\
\left[\hat{f}_W(0), \ldots, \hat{f}_W(2^{n-2}-1)\right] K_n(b_1,0;b_2,c) &= 2^{\frac{n}{2}-1}\left[\hat{f}\right],
\end{aligned}
$$

where $c = 0$ or $c = 1$. From Lemma 2.7, there exists a non-singular $2^n \times 2^n$-matrix Π such that

$$K_n(b_1, 0; b_2, c)\Pi = \left[\begin{array}{cccc} H_{n-2} & H_{n-2} & (-1)^c H_{n-2} & (-1)^c H_{n-2} \end{array} \right].$$

Π exchanges columns of $K_n(b_1, 0; b_2, c)$. Hence,

$$\left[\hat{f}_W\right] \left[\begin{array}{cccc} H_{n-2} & H_{n-2} & (-1)^c H_{n-2} & (-1)^c H_{n-2} \end{array} \right] = 2^{\frac{n}{2}-1} \left[\hat{f}\right] \Pi,$$

which shows that

$$\left|\left(\mathcal{W}(\hat{f}_W)\right)(\omega)\right| = 2^{\frac{n}{2}-1}$$

for every $\omega \in \{0,1\}^{n-2}$. This completes the proof. □

The following theorem states that the mapping in Theorem 2.6 is surjective.

Theorem 2.7 Let $n \geqslant 4$ be even and $g \in \mathrm{B}_{n-2}$. Let $\alpha_1, \ldots, \alpha_{2^{n-2}} \in \{0,1\}^n$, $b_1, b_2, b_3 \in \mathrm{V}_n$ and $c, d \in \{0,1\}$ such that $0 \leqslant dec(\alpha_1) < \cdots < dec(\alpha_{2^{n-2}}) \leqslant 2^n - 1$ and $b_1 \cdot \alpha_i = c$, $b_2 \cdot \alpha_i = d$ and $b_3 \cdot \alpha_i = c \oplus d$ for $1 \leqslant i \leqslant 2^{n-2}$. Let $\hat{F} : \{0,1\}^n \to \mathrm{N}$ be defined as

$$\hat{F}(\omega) = \begin{cases} 2^{n/2+1}\hat{g}(i-1) & \text{if } \omega = \alpha_i \\ 0 & \text{otherwise} \end{cases}$$

and $\hat{f} = \mathcal{W}^{-1}(\hat{F})$. If g is perfectly nonlinear, then $\hat{f} : \{0,1\}^n \to \{-1,1\}$ and f satisfies the PC with respect to $\mathrm{V}_n - \{b_1, b_2, b_3\}$.

(Proof) Since $\hat{F}(\omega) = 0$ when $\omega \neq \alpha_i$ and $b_1 \cdot \alpha_i = c$ and $b_2 \cdot \alpha_i = d$ for $1 \leqslant i \leqslant 2^{n-2}$,

$$\begin{aligned} \left[\hat{f}\right] &= \frac{1}{2^n}\left[\hat{F}(0), \ldots, \hat{F}(2^n-1)\right] H_n \\ &= \frac{1}{2^n}\left[\hat{F}(\alpha_1), \ldots, \hat{F}(\alpha_{2^{n-2}})\right] K_n(b_1, c; b_2, d) \\ &= \frac{1}{2^{\frac{n-1}{2}}}\left[\hat{g}(0), \ldots, \hat{g}(2^{n-2}-1)\right] K_n(b_1, c; b_2, d). \end{aligned}$$

From Lemma 2.7, for $K_n(b_1, c; b_2, d)$, there exists a non-singular $2^n \times 2^n$-matrix Π such that

$$K_n(b_1, c; b_2, d)\ \Pi = \left[\ H_{n-2} \quad H_{n-2} \quad (-1)^{c \vee d} H_{n-2} \quad (-1)^{c \vee d} H_{n-2}\ \right].$$

Π exchanges columns of $K_n(b_1, c; b_2, d)$. Hence,

$$\begin{aligned} \left[\hat{f}\right]\ \Pi &= \frac{1}{2^{\frac{n}{2}-1}} [\hat{g}] \left[\ H_{n-2} \quad H_{n-2} \quad (-1)^{c \vee d} H_{n-2} \quad (-1)^{c \vee d} H_{n-2}\ \right] \\ &= \frac{1}{2^{\frac{n}{2}-1}} \left[\ \hat{G} \quad \hat{G} \quad (-1)^{c \vee d} \hat{G} \quad (-1)^{c \vee d} \hat{G}\ \right]. \end{aligned}$$

Since $\left|\hat{G}(\omega)\right| = 2^{\frac{n}{2}-1}$ for every $\omega \in \{0,1\}^{n-2}$, $\hat{f} : \{0,1\}^n \to \{-1, 1\}$ and, from Theorem 2.5, f satisfies the PC with respect to $\mathrm{V}_n - \{b_1, b_2, b_3\}$. □

From Theorem 2.6 and 2.7, it is obvious that the algorithm below generates all Boolean functions in B_n that satisfy the PC with respect to all but three nonzero vectors and that is not perfectly nonlinear from all Boolean functions in $\mathrm{PC}_{n-2}(n-2)$ for even $n \geqslant 4$.

Algorithm 2.2

input $p \in \mathrm{PC}_{n-2}(n-2)$, $b_1, b_2 \in \mathrm{V}_n$ for even $n \geqslant 4$.

output $f_{(0,0)}, f_{(0,1)}, f_{(1,0)}, f_{(1,1)} \in \mathrm{B}_n$ that satisfy the PC with respect to $V_n - \{b_1, b_2, b_1 \oplus b_2\}$.

procedure

1. Let $(c, d) \in \{0,1\}^2$ and $\alpha_1^{(c,d)}, \ldots, \alpha_{2^{n-2}}^{(c,d)} \in \{0,1\}^n$ such that

$$0 \leqslant dec(\alpha_1^{(c,d)}) < \cdots < dec(\alpha_{2^{n-2}}^{(c,d)}) \leqslant 2^n - 1,$$

and, for every i such that $1 \leqslant i \leqslant 2^{n-2}$,

$$b_1 \cdot \alpha_i^{(c,d)} = c,\ b_2 \cdot \alpha_i^{(c,d)} = d.$$

2. Let

$$\hat{F}_{(c,d)}(\omega) = \begin{cases} 2^{n/2+1}\hat{p}(i-1) & \text{if } \omega = \alpha_i^{(c,d)} \\ 0 & \text{otherwise,} \end{cases}$$

where $\hat{F}_{(c,d)} = \mathcal{W}(\hat{f}_{(c,d)})$.

3. Let $\left[\hat{f}_{(c,d)}\right] = \frac{1}{2^n}\left[\hat{F}_{(c,d)}\right] H_n$.

□

For Algorithm 2.2, $\hat{F}_{(c,d)}(0) = 0$ only if $(c,d) \neq (0,0)$. Thus, $f_{(c,d)}$ is balanced if $(c,d) \neq (0,0)$ and not balanced otherwise.

The following corollary presents the relationship between the number of balanced Boolean functions satisfying the PC with respect to all but three elements in V_n and that of perfectly nonlinear Boolean functions in B_{n-2}.

Corollary 2.11 Let $n \geqslant 4$ be even. The number of balanced Boolean functions in B_n satisfying the PC with respect to all but three elements in V_n is ${}_{2^{n-1}}C_2\,|PC_{n-2}(n-2)|$. □

2.4.3 Examples

This section gives examples of Algorithm 2.1 and Algorithm2.2.

Example 2.3 Two Boolean functions in B_5 are constructed that satisfy the PC with respect to all but one nonzero vectors.

Let $p \in PC_4(4)$ be

$$p(x_1, x_2, x_3, x_4) = x_1x_2 \oplus x_3x_4.$$

Let $b = (0,1,1,1,1)$.

The elements ω's in $\{0,1\}^5$ that satisfy $b \cdot \omega = 0$ are

$$0, 1, 6, 7, 10, 11, 12, 13, 18, 19, 20, 21, 24, 25, 30, 31,$$

where each of the numbers represents $dec(\omega)$. Thus,

$$\begin{aligned} \left[\hat{F}_0\right] = {} & [0,0,8,8,8,-8,0,0,8,8,0,0,0,0,8,-8, \\ & 8,8,0,0,0,0,8,-8,0,0,-8,-8,-8,8,0,0]. \end{aligned}$$

$$\begin{aligned}\left[\hat{f}_0\right] &= \frac{1}{2^5}\left[\hat{F}_0\right] H_5 \\ &= [1,1,1,-1,1,-1,1,1,1,1,-1,1,1,-1,-1,-1, \\ &\quad 1,1,-1,1,1,-1,-1,-1,-1,-1,-1,1,-1,1,-1,-1].\end{aligned}$$

The algebraic normal form of f_0 is

$$f_0(x_1,\ldots,x_5) = x_1x_2 \oplus x_1x_3 \oplus x_2x_4 \oplus x_2x_5 \oplus x_4x_5.$$

f_1 can be generated in the same way as the above.

$$\begin{aligned}f_1(x_1,\ldots,x_5) &= x_2 \oplus x_1x_2 \oplus x_1x_3 \oplus x_2x_4 \oplus x_2x_5 \oplus x_4x_5 \\ &= x_2 \oplus f_0(x_1,\ldots,x_5).\end{aligned}$$

The truth tables of f_0 and f_1 are shown in Figure 2.1. f_0 is balanced, while f_1 is not balanced. $f_0, f_1 \in \mathrm{PC}_5(3)$ since they satisfy the PC with respect to $\mathrm{V}_5 - \{(0,1,1,1,1)\}$. For $b = (0,1,1,1,1)$,

$$\begin{aligned}f_0(x) \oplus f_0(x \oplus b) &\equiv 1, \\ f_1(x) \oplus f_1(x \oplus b) &\equiv 0.\end{aligned}$$

□

Example 2.4 Four Boolean functions in B_6 are constructed that satisfy the PC with respect to all but three nonzero vectors.

Let $p \in \mathrm{PC}_4(4)$ be

$$p(x_1,x_2,x_3,x_4) = x_1x_2 \oplus x_3x_4.$$

Let $b_1 = (1,1,1,1,0,0)$, $b_2 = (0,0,1,1,1,1)$ and $b_3 = (1,1,0,0,1,1)$.

The elements ω's in $\{0,1\}^6$ that satisfy $b_1 \cdot \omega = b_2 \cdot \omega = b_3 \cdot \omega = 0$ are

$$0,3,12,15,21,22,25,26,37,38,41,42,48,51,60,63,$$

where each of the numbers represents $dec(\omega)$. Thus,

$$\begin{aligned}\left[\hat{F}_{(0,0)}\right] &= [16,0,0,16,0,0,0,0,0,0,0,0,16,0,0,-16, \\ &\quad 0,0,0,0,0,16,16,0,0,16,-16,0,0,0,0,0, \\ &\quad 0,0,0,0,0,16,16,0,0,16,-16,0,0,0,0,0, \\ &\quad -16,0,0,-16,0,0,0,0,0,0,0,0,-16,0,0,16].\end{aligned}$$

x_1x_2 \ $x_3x_4x_5$	000	001	011	010	110	111	101	100
00			1			1		
01		1	1	1	1	1	1	
11	1				1	1	1	
10			1		1		1	1

$f_0(x_1, \ldots, x_5)$

x_1x_2 \ $x_3x_4x_5$	000	001	011	010	110	111	101	100
00			1			1		
01	1							1
11		1	1	1				1
10			1		1		1	1

$f_1(x_1, \ldots, x_5)$

$$\begin{aligned}\left[\hat{f}_0\right] &= \frac{1}{2^6}\left[\hat{F}_{(0,0)}\right] H_6 \\ &= [1,-1,1,-1,-1,-1,1,1,1,1,-1,-1,-1,1,-1,1, \\ &\quad 1,1,1,1,1,-1,-1,1,1,-1,-1,1,1,1,1,1, \\ &\quad 1,1,1,1,1,-1,-1,1,1,-1,-1,1,1,1,1,1, \\ &\quad -1,1,-1,1,1,1,-1,-1,-1,-1,1,1,1,-1,1,-1].\end{aligned}$$

The algebraic normal form of $f_{(0,0)}$ is

$$\begin{aligned}f_{(0,0)}(x_1,\ldots,x_6) &= x_1 \oplus x_3 \oplus x_1x_3 \oplus x_2x_3 \oplus x_1x_4 \oplus x_2x_4 \oplus \\ &\quad x_1x_5 \oplus x_3x_5 \oplus x_1x_6 \oplus x_3x_6 \oplus x_5x_6.\end{aligned}$$

$f_{(0,1)}$, $f_{(1,0)}$ and $f_{(1,1)}$ can be generated in the same way as the above.

$$\begin{aligned}f_{(0,1)}(x_1,\ldots,x_6) &= x_1x_3 \oplus x_2x_3 \oplus x_1x_4 \oplus x_2x_4 \oplus x_1x_5 \oplus \\ &\quad x_3x_5 \oplus x_1x_6 \oplus x_3x_6 \oplus x_5x_6 \\ &= x_1 \oplus x_3 \oplus f_{(0,0)}(x_1,\ldots,x_6). \\ f_{(1,0)}(x_1,\ldots,x_6) &= x_3 \oplus x_1x_3 \oplus x_2x_3 \oplus x_1x_4 \oplus x_2x_4 \oplus x_1x_5 \oplus \\ &\quad x_3x_5 \oplus x_1x_6 \oplus x_3x_6 \oplus x_5x_6 \\ &= x_1 \oplus f_{(0,0)}(x_1,\ldots,x_6). \\ f_{(1,1)}(x_1,\ldots,x_6) &= x_1 \oplus x_1x_3 \oplus x_2x_3 \oplus x_1x_4 \oplus x_2x_4 \oplus x_1x_5 \oplus \\ &\quad x_3x_5 \oplus x_1x_6 \oplus x_3x_6 \oplus x_5x_6 \\ &= x_3 \oplus f_{(0,0)}(x_1,\ldots,x_6).\end{aligned}$$

$f_{(0,1)}$ $f_{(1,0)}$ and $f_{(1,1)}$ are balanced, while $f_{(0,0)}$ is not balanced. The truth table of $f_{(1,1)}$ is presented in Figure 2.2. $f_{(0,0)}$, $f_{(0,1)}$, $f_{(1,0)}$, $f_{(1,1)}$ $\in \mathrm{PC}_6(3)$ since they satisfy the PC with respect to $\mathrm{V}_6 - \{b_1, b_2, b_3\}$ and the Hamming weights of b_1, b_2, b_3 are all 4. Table 2.1 shows the values of $f_{(c_1,c_2)}(x) \oplus f_{(c_1,c_2)}(x \oplus b_i)$ for $(c_1, c_2) \in \{0,1\}^2$ and $i = 1, 2, 3$. □

$x_1x_2x_3$ \ $x_4x_5x_6$	000	001	011	010	110	111	101	100
000			1			1		
001		1	1	1	1	1	1	
011	1				1	1	1	
010			1		1		1	1
110	1							1
111	1	1		1	1		1	1
101			1		1		1	1
100	1				1	1	1	

Figure 2.2: Truth table of $f_{(1,1)}$

2.5 Boolean Functions Satisfying the PC of Degree $n-2$

2.5.1 Boolean Functions with an Even Number of Variables

In this section, it is proved that, for every even $n \geqslant 4$, $\mathrm{PC}_n(n-2) = \mathrm{PC}_n(n)$.

First, we present a simple lemma. For $u \in \{0,1\}^{2^n}$ and $\alpha \in \{0,1\}^n$, let $[u]_\alpha$ denote the $(dec(\alpha)+1)$-th element of u.

Lemma 2.8 Let $n \geqslant 2$ and $b_1, \ldots, b_n, b_{n+1} \in \{0,1\}^n$. Let $b_i = (1, \ldots, 1, \overset{i}{0}, 1, \ldots, 1)$ for every i such that $1 \leqslant i \leqslant n$ and $b_{n+1} = (1, \ldots, 1)$. For $v_{b_1}, \ldots, v_{b_{n+1}}$ and $\alpha = (\alpha_1, \ldots, \alpha_n) \in \{0,1\}^n$,

- if $W(\alpha)$ is even, then

$$[v_{b_{n+1}}]_\alpha = 1,$$

$$[v_{b_i}]_\alpha = \begin{cases} 1 & \text{if } \alpha_i = 0 \\ -1 & \text{if } \alpha_i = 1, \end{cases}$$

- if $W(\alpha)$ is odd, then

$$[v_{b_{n+1}}]_\alpha = -1,$$

Table 2.1: The value of $f_{(c_1,c_2)}(x) \oplus f_{(c_1,c_2)}(x \oplus b_i)$

	b_1	b_2	b_3
$f_{(0,0)}$	0	0	0
$f_{(0,1)}$	0	1	1
$f_{(1,0)}$	1	0	1
$f_{(1,1)}$	1	1	0

$$[v_{b_i}]_\alpha = \begin{cases} 1 & \text{if } \alpha_i = 1 \\ -1 & \text{if } \alpha_i = 0, \end{cases}$$

(Proof) This lemma can be proved from the fact that,

$$[v_{b_{n+1}}]_\alpha = (-1)^{\alpha_1 \oplus \cdots \oplus \alpha_n},$$

and, for each i such that $1 \leqslant i \leqslant n$,

$$[v_{b_i}]_\alpha = (-1)^{\alpha_1 \oplus \cdots \oplus \alpha_{i-1} \oplus \alpha_{i+1} \oplus \cdots \oplus \alpha_n}.$$

□

Theorem 2.8 For every even $n \geqslant 4$, $\mathrm{PC}_n(n-2) = \mathrm{PC}_n(n)$.

(Proof) Let $b_i = (1, \ldots, 1, \overset{i}{0}, 1, \ldots, 1) \in \{0,1\}^n$ for every i such that $1 \leqslant i \leqslant n$ and $b_{n+1} = (1, \ldots, 1) \in \{0,1\}^n$. Suppose that $f \in \mathrm{PC}_n(n-2)$. Then, $C_f(a) = 0$ for every $a \in \{0,1\}^n$ such that $1 \leqslant W(a) \leqslant n-2$. Thus, $\left[\hat{F}^2\right]$ is able to be represented as

$$\left[\hat{F}^2\right] = C_f(0){v_0}^{\mathrm{T}} + C_f(b_1){v_{b_1}}^{\mathrm{T}} + \cdots + C_f(b_n){v_{b_n}}^{\mathrm{T}} + C_f(b_{n+1}){v_{b_{n+1}}}^{\mathrm{T}}.$$

Let $u_0 = {v_0}^{\mathrm{T}}$ and $u_i = ({v_0}^{\mathrm{T}} + {v_{b_i}}^{\mathrm{T}})/2$ for every $1 \leqslant i \leqslant n+1$. Then, $\left[\hat{F}^2\right]$ can be represented as

$$\left[\hat{F}^2\right] = c_0u_0 + c_1u_1 + \cdots + c_{n+1}u_{n+1},$$

where

$$\begin{aligned} c_0 &= C_f(0) - (C_f(b_1) + \cdots + C_f(b_{n+1})), \\ c_i &= 2C_f(b_i). \end{aligned}$$

From Lemma 2.8, for any odd s such that $1 \leqslant s \leqslant n$ and $i_1, \ldots, i_s$ such that $0 \leqslant i_1 < \cdots < i_s \leqslant n-1$,

$$\hat{F}^2\left(\sum_{k=1}^{s} 2^{i_k}\right) = c_0 + \sum_{k=1}^{s} c_{i_k+1},$$

and, for any even t such that $1 \leqslant t \leqslant n$ and $j_1, \ldots, j_t$ such that $0 \leqslant j_1 < \cdots < j_t \leqslant n-1$,

$$\hat{F}^2(\sum_{k=1}^{t} 2^{j_k}) = c_0 + c_1 + \cdots + c_{n+1} - \sum_{k=1}^{t} c_{j_k+1}.$$

Thus, for every pair of odd integers s and t such that $1 \leqslant s,t \leqslant n$ and $s+t \leqslant n$ and $i_1, \ldots, i_s$ and $j_1, \ldots, j_t$ such that $\{i_1, \ldots, i_s\} \cap \{j_1, \ldots, j_t\} = \emptyset$ and $0 \leqslant i_1 < \cdots < i_s \leqslant n-1$ and $0 \leqslant j_1 < \cdots < j_t \leqslant n-1$,

$$\begin{aligned} &\hat{F}^2(0) + \hat{F}^2(\sum_{k=1}^{s} 2^{i_k}) + \hat{F}^2(\sum_{l=1}^{t} 2^{j_l}) + \hat{F}^2(\sum_{k=1}^{s} 2^{i_k} + \sum_{l=1}^{t} 2^{j_l}) \\ &= 4c_0 + 2(c_1 + \cdots + c_{n+1}) \\ &= 2^{n+2}. \end{aligned}$$

Since $n+2$ is even, from Lemma 2.1, 2.5, 2.6, all of $\hat{F}^2(0)$, $\hat{F}^2(\sum_{k=1}^{s} 2^{i_k})$, $\hat{F}^2(\sum_{l=1}^{t} 2^{j_l})$, $\hat{F}^2(\sum_{k=1}^{s} 2^{i_k} + \sum_{l=1}^{t} 2^{j_l})$ are equal to 2^n, or only one of them is equal to 2^{n+2} and the others are equal to 0.

In the former case, f is perfectly nonlinear.

In the latter case, if $\hat{F}^2(0) = 2^{n+2}$, then $\hat{F}(\omega) = 0$ for every $\omega \neq 0$, which contradicts that $\sum_{\omega \in \{0,1\}^n} \hat{F}^2(\omega) = 2^{2n}$.

If $\hat{F}^2(0) = c_0 + c_1 + \cdots + c_{n+1} = 0$, then $c_0 = 2^{n+1}$. For this case,

$$\begin{aligned} &\hat{F}^2(1,0,\ldots,0) + \hat{F}^2(0,1,0,\ldots,0) + \hat{F}^2(1,1,0,\ldots,0) \\ &= (c_0 + c_1) + (c_0 + c_2) + (c_0 + c_3 + \cdots + c_{n+1}) \\ &= 2c_0 + (c_0 + c_1 + \cdots + c_{n+1}) \\ &= 2^{n+2}. \end{aligned}$$

From Lemma 2.1, 2.5, and $c_0 = 2^{n+1}$, there are following three cases:

C-1. $c_1 = 2^{n+1}$, $c_2 = -2^{n+1}$, $c_0 + c_3 + \cdots + c_{n+1} = 0$,

C-2. $c_1 = -2^{n+1}$, $c_2 = 2^{n+1}$, $c_0 + c_3 + \cdots + c_{n+1} = 0$,

C-3. $c_1 = -2^{n+1}$, $c_2 = -2^{n+1}$, $c_0 + c_3 + \cdots + c_{n+1} = 2^{n+2}$.

For C-1. Since

$$\begin{aligned}
&\hat{F}^2(0,0,1,0,\ldots,0)+\hat{F}^2(0,0,0,1,0,\ldots,0)+\hat{F}^2(1,1,1,1,0,\ldots,0)\\
&= (c_0+c_3)+(c_0+c_4)+(c_0+c_5+\cdots+c_{n+1})\\
&= 2c_0+(c_0+c_3+\cdots+c_{n+1})\\
&= 2^{n+2},
\end{aligned}$$

the same argument as the above one shows that

C-1.1. $c_3 = 2^{n+1}$, $c_4 = -2^{n+1}$, $c_0+c_5+\cdots+c_{n+1}=0$.

C-1.2. $c_3 = -2^{n+1}$, $c_4 = 2^{n+1}$, $c_0+c_5+\cdots+c_{n+1}=0$.

C-1.3. $c_3 = -2^{n+1}$, $c_4 = -2^{n+1}$, $c_0+c_5+\cdots+c_{n+1}=2^{n+2}$.

For C-1.1, 1.2 and 1.3, the following three equations can be derived, respectively,

$$\hat{F}^2(1,0,1,0,0,\ldots,0) = c_0+c_1+\cdots+c_{n+1}-(c_1+c_3) = -2^{n+2},$$

$$\hat{F}^2(1,0,0,1,0,\ldots,0) = c_0+c_1+\cdots+c_{n+1}-(c_1+c_4) = -2^{n+2},$$

$$\hat{F}^2(0,1,1,1,0,\ldots,0) = c_0+c_2+c_3+c_4 = -2^{n+2},$$

which are contradictions.

For C-2. This case can be proved in the same way as C-1.

For C-3. Since $c_0+c_3+\cdots+c_{n+1}=2^{n+2}$,

$$\begin{aligned}
&\hat{F}^2(0,0,1,0,\ldots,0)+\hat{F}^2(0,0,0,1,0,\ldots,0)+\hat{F}^2(1,1,1,1,0,\ldots,0)\\
&= (c_0+c_3)+(c_0+c_4)+(c_0+c_5+\cdots+c_{n+1})\\
&= 2c_0+(c_0+c_3+\cdots+c_{n+1})\\
&= 2^{n+3}.
\end{aligned}$$

From Lemma 2.1, 2.5, there are following three cases:

C-3.1. $c_3 = 2^{n+1}$, $c_4 = -2^{n+1}$, $c_0+c_5+\cdots+c_{n+1}=2^{n+2}$,

C-3.2. $c_3 = -2^{n+1}$, $c_4 = 2^{n+1}$, $c_0+c_5+\cdots+c_{n+1}=2^{n+2}$,

C-3.3. $c_3 = 2^{n+1}$, $c_4 = 2^{n+1}$, $c_0 + c_5 + \cdots + c_{n+1} = 0$.

For C-3.1, 3.2 and 3.3, the following three equations can be derived, respectively,

$$\hat{F}^2(1,1,0,1,0,\ldots,0) = c_0 + c_1 + c_2 + c_4 = -2^{n+2},$$

$$\hat{F}^2(1,1,1,0,0,\ldots,0) = c_0 + c_1 + c_2 + c_3 = -2^{n+2},$$

$$\hat{F}^2(0,0,1,1,0,\ldots,0) = c_0 + c_1 + \cdots + c_{n+1} - (c_3 + c_4) = -2^{n+2},$$

which are contradictions. Hence, the theorem has been proved. □

Since perfectly nonlinear Boolean functions are not balanced, the following corollary can be derived. It presents an upper bound of the degree of the PC of balanced Boolean functions with an even number of inputs.

Corollary 2.12 For every even $n \geqslant 4$, the degree of the PC of balanced Boolean functions is less than $n - 2$. □

As for the lower bound, the following has been proved.

Proposition 2.8 [SZZ93] Let $n \geqslant 4$ be even. Suppose that $n = 3t + c$, where $c = 0, 1$, or 2. Then, there exist balanced Boolean functions in B_n that satisfy the PC of degree $2t - 1$ when $c = 0, 1$ or $2t$ when $c = 2$. □

Table 2.2 shows the bounds of the degree of the PC of balanced Boolean functions. The bounds are tight for $n = 4, 6$.

2.5.2 Boolean Functions with an Odd Number of Variables

In this section, it is shown that, for every odd $n \geqslant 3$, every $f \in \mathrm{PC}_n(n-2)$ satisfies the PC with respect to all but one elements in V_n.

Table 2.2: Bounds of the degree of the PC of balanced Boolean functions

number of variables	4	6	8	10	12	14	16	18	20	22	24
upper bound	1	3	5	7	9	11	13	15	17	19	21
lower bound	1	3	4	5	7	8	9	11	12	13	15

Lemma 2.9 Let $x, y, z \geqslant 0$ be integers and $m \geqslant 0$ be an even integer. $x^2 + y^2 + z^2 = 3 \cdot 2^m$ if and only if $x = y = z = 2^{m/2}$.

(Proof) x, y, z can be represented as

$$x = 2^{e_1} q_1,\ y = 2^{e_2} q_2,\ z = 2^{e_3} q_3,$$

where $e_1, e_2, e_3 \geqslant 0$, and each of q_1, q_2, q_3 is 0 or odd.

(i) If we assume that $y = z = 0$, then $x^2 = 3 \cdot 2^m$, which contradicts that x is an integer.

(ii) Suppose that $x \neq 0$, $y \neq 0$, $z = 0$. Then, $2^{2e_1} {q_1}^2 + 2^{2e_2} {q_2}^2 = 3 \cdot 2^m$. Since, without loss of generality, we can assume that $e_2 \geqslant e_1 \geqslant 0$,

$${q_1}^2 + 2^{2(e_2 - e_1)} {q_2}^2 = 3 \cdot 2^{m - 2e_1}.$$

If $e_1 = e_2$, then ${q_1}^2 + {q_2}^2$ is a multiple of 2 but not of 4. This implies that $m - 2e_1 = 1$, which contradicts that m is even.

If $e_1 < e_2$, then the left-hand side is odd and $m - 2e_1 = 0$. Thus, ${q_1}^2 + 2^{2(e_2 - e_1)} {q_2}^2 = 3$, which implies that $2(e_2 - e_1) = 1$. This contradicts that e_1 and e_2 are integers.

(iii) Suppose that none of x, y, z is 0. Without loss of generality, we may assume that $0 \leqslant e_1 \leqslant e_2 \leqslant e_3$.

$$\begin{aligned} 2^{2e_1} {q_1}^2 + 2^{2e_2} {q_2}^2 + 2^{2e_3} {q_3}^2 &= 3 \cdot 2^m \\ {q_1}^2 + 2^{2(e_2 - e_1)} {q_2}^2 + 2^{2(e_3 - e_1)} {q_3}^2 &= 3 \cdot 2^{m - 2e_1}. \end{aligned}$$

If we assume that $e_1 \neq e_2$ and $e_1 \neq e_3$, or $e_1 = e_2 = e_3$, then the left-hand side is odd. This implies that $m - 2e_1 = 0$ and

$$q_1{}^2 + 2^{2(e_2-e_1)}q_2{}^2 + 2^{2(e_3-e_1)}q_3{}^2 = 3.$$

Since $q_1, q_2, q_3 \geqslant 1$, $e_1 = e_2 = e_3 = m/2$ and $q_1 = q_2 = q_3 = 1$.

If we assume that $e_1 = e_2$ and $e_1 \neq e_3$, then $q_1{}^2 + q_2{}^2 = 3 \cdot 2^{m-2e_1} - 2^{2(e_3-e_1)}q_3{}^2$. Since $q_1{}^2 + q_2{}^2$ is a multiple of 2 but not of 4, $m - 2e_1 = 1$ or $2(e_3 - e_1) = 1$. This situation cannot occur because m is even.

Hence, the lemma has been proved. □

Theorem 2.9 For every odd $n \geqslant 3$, if $f \in \mathrm{PC}_n(n-2)$, then, for some $b \in \{0,1\}^n$ such that $W(b) \geqslant n-1$, f satisfies the PC with respect to $\mathrm{V}_n - \{b\}$.

(Proof) Suppose that $f \in \mathrm{PC}_n(n-2)$. Then, $C_f(a) = 0$ for every $a \in \{0,1\}^n$ whose Hamming weight is at most $n-2$. $\left[\hat{F}^2\right]$ is able to be represented as

$$\left[\hat{F}^2\right] = c_0u_0 + c_1u_1 + \cdots + c_{n+1}u_{n+1},$$

where, for every $1 \leqslant i \leqslant n+1$,

$$\begin{aligned} u_0 &= {v_0}^{\mathrm{T}}, \\ u_i &= ({v_0}^{\mathrm{T}} + {v_{b_i}}^{\mathrm{T}})/2, \\ c_0 &= C_f(0) - (C_f(b_1) + \cdots + C_f(b_{n+1})), \\ c_i &= 2C_f(b_i). \end{aligned}$$

Hence, from Lemma 2.8, for every i, j such that $0 \leqslant i, j \leqslant n-1$ and $i \neq j$,

$$\hat{F}^2(2^i) = c_0 + c_{i+1},$$

$$\hat{F}^2(2^i + 2^j) = c_0 + c_1 + \cdots + c_{n+1} - (c_{i+1} + c_{j+1}).$$

Thus, for every i, j such that $0 \leqslant i, j \leqslant n-1$ and $i \neq j$,

$$\begin{aligned} &\hat{F}^2(0) + \hat{F}^2(2^i) + \hat{F}^2(2^j) + \hat{F}^2(2^i + 2^j) \\ &= 4c_0 + 2(c_1 + \cdots + c_{n+1}) \\ &= 2^{n+2}. \end{aligned}$$

Since $n+2$ is odd, from Lemma 2.1, 2.5, 2.6, two of $\hat{F}^2(0)$, $\hat{F}^2(2^i)$, $\hat{F}^2(2^j)$ and $\hat{F}^2(2^i+2^j)$ are equal to 0, and two of them are equal to 2^{n+1}.

If $\hat{F}^2(0) = c_0 + c_1 + \cdots + c_{n+1} = 0$, then $c_0 = 2^{n+1}$. For every i, j such that $0 \leqslant i, j \leqslant n-1$ and $i \neq j$, since

$$\begin{aligned}&\hat{F}^2(2^i) + \hat{F}^2(2^j) + \hat{F}^2(2^i+2^j)\\ &= 2c_0 + (c_0 + c_1 + \cdots + c_{n+1})\\ &= 2^{n+2},\end{aligned}$$

each of $\hat{F}^2(2^i)$, $\hat{F}^2(2^j)$ and $\hat{F}^2(2^i+2^j)$ is equal to 0 or 2^{n+1}. Thus, each of $c_1, \ldots, c_n$ is equal to 0 or -2^{n+1}. Since, for every i, j such that $0 \leqslant i, j \leqslant n-1$ and $i \neq j$,

$$\hat{F}^2(2^i+2^j) = -(c_{i+1} + c_{j+1})$$

is equal to 0 or 2^{n+1}, at most only one of $c_1, \ldots, c_n$ is equal to -2^{n+1}, and the others are equal to 0. If one of $c_1, \ldots, c_n$ is equal to -2^{n+1}, then $c_{n+1} = 0$, otherwise, $c_{n+1} = -2^{n+1}$, because $c_0 = 2^{n+1}$ and $c_0 + c_1 + \cdots + c_{n+1} = 0$. Hence, one of $c_1, \ldots, c_{n+1}$ is equal to -2^{n+1} and the others are equal to 0.

If $\hat{F}^2(0) = c_0 + c_1 + \cdots + c_{n+1} = 2^{n+1}$, then $c_0 = 0$. Thus, for every i such that $0 \leqslant i \leqslant n-1$, $\hat{F}^2(2^i) = c_{i+1}$, and c_{i+1} is equal to 0 or 2^{n+1}. Since, for every i, j such that $0 \leqslant i, j \leqslant n-1$ and $i \neq j$,

$$\hat{F}^2(2^i+2^j) = 2^{n+1} - (c_{i+1} + c_{j+1}),$$

at most one of $c_1, \ldots, c_n$ is 2^{n+1}. If one of $c_1, \ldots, c_n$ is equal to 2^{n+1}, then $c_{n+1} = 0$, otherwise, $c_{n+1} = 2^{n+1}$, because $c_0 = 0$ and $c_0 + c_1 + \cdots + c_{n+1} = 2^{n+1}$.

From the above discussion, there exist some i such that $1 \leqslant i \leqslant n+1$ and $C_f(a) = 0$ for every $a \in \mathrm{V}_n - \{b_i\}$. Hence, f satisfies the PC with respect to all but one elements in V_n. □

The following corollaries can be derived from the above two theorems.

Corollary 2.13 For every odd $n \geqslant 3$,

$$|\mathrm{PC}_n(n-2)| = 2(n+1)|\mathrm{PC}_{n-1}(n-1)|,$$

and the number of balanced Boolean functions in $\mathrm{PC}_n(n-2)$ is $(n+1)|\mathrm{PC}_{n-1}(n-1)|$.

(Proof) There exist $n+1$ elements in V_n whose Hamming weight is at least $n-1$. For each $b \in \mathrm{V}_n$, there exist $2|\mathrm{PC}_{n-1}(n-1)|$ Boolean functions that satisfy the PC with respect to $\mathrm{V}_n - \{b\}$, and half of them are balanced. □

Corollary 2.14 For every odd $n \geqslant 3$, the nonlinearity of Boolean functions in $\mathrm{PC}_n(n-2)$ is $2^{n-1} - 2^{(n-1)/2}$. □

2.6 Relationships Between the PC and the SAC

This section presents some relationships between $\mathrm{PC}_n(k)$ and $\mathrm{SAC}_n(m)$.

Rothaus[Rot76] presented a few methods for constructing Boolean bent functions. One of them gives Boolean bent functions of the form

$$f(x_1, \ldots, x_n) = \bigoplus_{i=1}^{m} x_i x_{m+i} \oplus g(x_1, \ldots, x_m),$$

where $n = 2m$ and g is an arbitrary Boolean function with m variables.

It is apparent, from definitions of the SAC and the PC, that Boolean functions satisfying the PC of degree 1 also satisfy the SAC of order 0. We show the relationships between $\mathrm{SAC}_n(1)$ and $\mathrm{PC}_n(k)$.

The following theorem shows that perfectly nonlinear Boolean functions do not necessarily satisfy the SAC of order 1.

Theorem 2.10 For every even $n \geqslant 2$, $\mathrm{PC}_n(n) \not\subseteq \mathrm{SAC}_n(1)$.

(Proof) Let $n = 2m$ and

$$f(x_1, \ldots, x_n) = \bigoplus_{i=1}^{m} x_i x_{m+i}.$$

Then $f \in \mathrm{PC}_n(n)$, and

$$f|_{x_n=1}(x_1, \ldots, x_{n-1}) = \bigoplus_{i=1}^{m-1} x_i x_{m+i} \oplus x_m.$$

Thus,

$$f|_{x_n=1}(x_1,\ldots,x_{n-1})\oplus \\ f|_{x_n=1}(x_1,\ldots,x_{m-1},x_m\oplus 1,x_{m+1},\ldots,x_{n-1})\equiv 1.$$

This implies that $f \notin \mathrm{SAC}_n(1)$. □

The following theorem shows that, for every odd $n \geqslant 3$, all the Boolean functions with n variables satisfying the PC of degree $n-1$ satisfy the SAC of order 1, while those satisfying the PC of degree $n-2$ necessarily not. We prove the theorem by using the following lemma.

Lemma 2.10 [For90] For any $f \in \mathrm{B}_n$, $f \in \mathrm{SAC}_n(1)$ if and only if, $f \in \mathrm{SAC}_n(0)$ and

$$\sum_{\omega\in\{0,1\}^n} \hat{F}(\omega)\hat{F}(\omega\oplus a)(-1)^{\omega_i} = 0$$

for every $a \in \{0,1\}^n$ whose Hamming weight is 1 and every i such that the i-th bit of a is 0. □

Theorem 2.11 For every odd $n \geqslant 3$,

1. $\mathrm{PC}_n(n-1) \subseteq \mathrm{SAC}_n(1)$,
2. $\mathrm{PC}_n(n-2) \not\subseteq \mathrm{SAC}_n(1)$.

(Proof) 1. Suppose that $f \in \mathrm{PC}_n(n-1)$. It is clear from the definition that $f \in \mathrm{SAC}_n(0)$. If n is odd, then $\hat{F}(\omega) = 0$ either for every $\omega \in \{0,1\}^n$ whose Hamming weight is even or for every $\omega \in \{0,1\}^n$ whose Hamming weight is odd. For any $a \in \{0,1\}^n$ whose Hamming weight is 1, $\hat{F}(\omega) = 0$ or $\hat{F}(\omega\oplus a) = 0$, because the Hamming weight of either ω or $\omega \oplus a$ is odd. Thus,

$$\sum_{\omega\in\{0,1\}^n} \hat{F}(\omega)\hat{F}(\omega\oplus a)(-1)^{\omega_i} = 0,$$

which implies that $f \in \mathrm{SAC}_n(1)$.

2. Let $m = (n-1)/2$ and $p \in \mathrm{B}_{2m}$ such that

$$p(x_1,\ldots,x_{2m}) = x_1x_{2m} \oplus x_2x_{2m-1} \oplus \cdots \oplus x_mx_{m+1}.$$

For $b = (0, 1, \ldots, 1) \in \{0,1\}^n$ and $\alpha_1, \ldots, \alpha_{2^{n-1}} \in \{0,1\}^n$ such that $0 \leqslant dec(\alpha_1) < \cdots < dec(\alpha_{2^{n-1}}) \leqslant 2^n - 1$ and $b \cdot \alpha_i = 1$ for $1 \leqslant i \leqslant 2^{n-1}$, let

$$\hat{F}(\omega) = \begin{cases} 2^{(n+1)/2}\hat{p}(i-1) & \text{if } \omega = \alpha_i \\ 0 & \text{otherwise.} \end{cases}$$

Let $\hat{f} = \mathcal{W}^{-1}(\hat{F})$. Then, $f \in \mathrm{PC}_n(n-2)$, because $f \in \mathrm{B}_n$ and f satisfies the PC with respect to $\mathrm{V}_n - \{b\}$ from Theorem 2.4.

Since

$$p(0, x_2, \ldots, x_{2m}) \oplus p(1, x_2, \ldots, x_{2m}) = x_{2m}$$

and

$$b \cdot (0, \omega_2, \ldots, \omega_n) = b \cdot (1, \omega_2, \ldots, \omega_n),$$

for $a = (1, 0, \ldots, 0) \in \{0,1\}^n$,

$$\begin{aligned} \hat{F}(\alpha_{2j-1})\hat{F}(\alpha_{2j-1} \oplus a) &= \hat{F}(\alpha_{2j-1})\hat{F}(\alpha_{2j}) \\ &= \begin{cases} 2^{n+1} & j = 1, \ldots, 2^{n-3} \\ -2^{n+1} & j = 2^{n-3}+1, \ldots, 2^{n-2}. \end{cases} \end{aligned}$$

The n-th bit of α_i is equal to 0 for $i = 1, \ldots, 2^{n-2}$ and equal to 1 for $i = 2^{n-2}+1, \ldots, 2^{n-1}$. Thus,

$$\begin{aligned} \sum_{\omega \in \{0,1\}^n} \hat{F}(\omega)\hat{F}(\omega \oplus a)(-1)^{\omega_n} &= 2^{n+1}2^{n-2} + (-2^{n+1})(-1)2^{n-2} \\ &= 2^{2n}. \end{aligned}$$

This implies that $f \notin \mathrm{SAC}_n(1)$. □

Example 2.5 We present an example of Boolean functions in B_n that satisfy the PC of degree $n-2$ and that do not satisfy the SAC of order 1.

Let $n = 5$. Let $p \in \mathrm{PC}_4(4)$ such that

$$p(x_1, x_2, x_3, x_4) = x_1x_4 \oplus x_2x_3,$$

and $b = (0, 1, 1, 1, 1)$. Let $\hat{F}(\omega_1, \ldots, \omega_5)$ be defined as in the proof of Theorem 2.11. Table 2.3 shows the $\hat{F}(\omega_1, \ldots, \omega_5)$.

Table 2.3: $\hat{F}$ of Example 2.5

$(\omega_1, \omega_2, \omega_3, \omega_4, \omega_5)$	$\hat{F}$	$(\omega_1, \omega_2, \omega_3, \omega_4, \omega_5)$	$\hat{F}$
$(0,0,0,0,0)$	8	$(0,0,0,0,1)$	0
$(1,0,0,0,0)$	8	$(1,0,0,0,1)$	0
$(0,1,0,0,0)$	0	$(0,1,0,0,1)$	8
$(1,1,0,0,0)$	0	$(1,1,0,0,1)$	-8
$(0,0,1,0,0)$	0	$(0,0,1,0,1)$	8
$(1,0,1,0,0)$	0	$(1,0,1,0,1)$	-8
$(0,1,1,0,0)$	8	$(0,1,1,0,1)$	0
$(1,1,1,0,0)$	8	$(1,1,1,0,1)$	0
$(0,0,0,1,0)$	0	$(0,0,0,1,1)$	8
$(1,0,0,1,0)$	0	$(1,0,0,1,1)$	-8
$(0,1,0,1,0)$	8	$(0,1,0,1,1)$	0
$(1,1,0,1,0)$	8	$(1,1,0,1,1)$	0
$(0,0,1,1,0)$	-8	$(0,0,1,1,1)$	0
$(1,0,1,1,0)$	-8	$(1,0,1,1,1)$	0
$(0,1,1,1,0)$	0	$(0,1,1,1,1)$	-8
$(1,1,1,1,0)$	0	$(1,1,1,1,1)$	8

$$f(x_1, \ldots, x_5) = x_2 \oplus x_1x_2 \oplus x_2x_3 \oplus x_2x_4 \oplus x_3x_4 \oplus x_1x_5.$$

$f \in \mathrm{PC}_5(3)$ because f satisfies the PC with respect to $\mathrm{V}_5 - \{(0,1,1,1,1)\}$. Let

$$\begin{aligned} f(1, x_2, x_3, x_4, x_5) &= x_2x_3 \oplus x_2x_4 \oplus x_3x_4 \oplus x_5 \\ &\stackrel{\text{def}}{=} g(x_2, x_3, x_4, x_5). \end{aligned}$$

Then,

$$g(x_2, x_3, x_4, x_5) \oplus g(x_2, x_3, x_4, x_5 \oplus 1) \equiv 1,$$

and g does not satisfy the SAC. Thus, $f \notin \mathrm{SAC}_5(1)$. □

The following theorem shows that, for every odd $n \geqslant 3$, Boolean functions with n variables satisfying the PC of degree $n-1$ do not necessarily satisfy the SAC of order 2. The proof of this theorem uses the following lemma.

Lemma 2.11 [For90] For any $f \in \mathrm{B}_n$, $f \in \mathrm{SAC}_n(2)$ if and only if $f \in \mathrm{SAC}_n(1)$ and

$$\sum_{\omega \in \{0,1\}^n} \hat{F}(\omega)\hat{F}(\omega \oplus a)(-1)^{\omega_i} = 0$$

for every $a \in \{0,1\}^n$ whose Hamming weight is 2 and every i such that the i-th bit of a is 0. □

Theorem 2.12 For every odd $n \geqslant 3$, $\mathrm{PC}_n(n-1) \not\subseteq \mathrm{SAC}_n(2)$.

(Proof) If $n = 3$, then $\mathrm{PC}_3(2) \neq \emptyset$ and $\mathrm{SAC}_3(2) = \emptyset$.

For $n \geqslant 5$, let $m = (n-1)/2$ and $p \in \mathrm{B}_{2m}$ such that

$$p(x_1, \ldots, x_{2m}) = x_1x_{2m} \oplus x_2x_{2m-1} \oplus \cdots \oplus x_mx_{m+1}.$$

For $\alpha_1, \ldots, \alpha_{2^{n-1}} \in \{0,1\}^n$ such that $0 \leqslant dec(\alpha_1) < \cdots < dec(\alpha_{2^{n-1}}) \leqslant 2^n - 1$ and the Hamming weights of them are odd,

$$\hat{F}(\omega) = \begin{cases} 2^{(n+1)/2}\hat{p}(i-1) & \text{if } \omega = \alpha_i \\ 0 & \text{otherwise.} \end{cases}$$

Let $\hat{f} = \mathcal{W}^{-1}(\hat{F})$. Then, $f \in \mathrm{PC}_n(n-1)$, because $f \in \mathrm{B}_n$ and f satisfies the PC with respect to $\mathrm{V}_n - \{(1,\ldots,1)\}$ from Theorem 2.4.

Since

$$p(0, x_2, \ldots, x_{2m}) \oplus p(1, x_2, \ldots, x_{2m}) = x_{2m}$$

and

$$\begin{aligned} W((0,0,\omega_3,\ldots,\omega_n)) &= W((1,1,\omega_3,\ldots,\omega_n)), \\ W((0,1,\omega_3,\ldots,\omega_n)) &= W((1,0,\omega_3,\ldots,\omega_n)), \end{aligned}$$

for $a = (1,1,0,\ldots,0) \in \{0,1\}^n$,

$$\begin{aligned} \hat{F}(\alpha_{2j-1})\hat{F}(\alpha_{2j-1} \oplus a) &= \hat{F}(\alpha_{2j-1})\hat{F}(\alpha_{2j}) \\ &= \begin{cases} 2^{n+1} & j = 1,\ldots,2^{n-3} \\ -2^{n+1} & j = 2^{n-3}+1,\ldots,2^{n-2}. \end{cases} \end{aligned}$$

The n-th bit of α_i is equal to 0 for $i = 1,\ldots,2^{n-2}$ and equal to 1 for $i = 2^{n-2}+1,\ldots,2^{n-1}$. Thus,

$$\begin{aligned} \sum_{\omega \in \{0,1\}^n} \hat{F}(\omega)\hat{F}(\omega \oplus a)(-1)^{\omega_n} &= 2^{n+1}2^{n-2} + (-2^{n+1})(-1)2^{n-2} \\ &= 2^{2n}. \end{aligned}$$

This implies that $f \notin \mathrm{SAC}_n(2)$. □

Example 2.6 We give an example of Boolean functions in B_n that satisfy the PC of degree $n-1$ and that do not satisfy the SAC of order 2.

Let $n = 5$. Let $p \in \mathrm{PC}_4(4)$ such that

$$p(x_1, x_2, x_3, x_4) = x_1x_4 \oplus x_2x_3.$$

Let $\hat{F}(\omega_1,\ldots,\omega_5)$ be defined as in the proof of Theorem 2.12. Table 2.4 shows the $\hat{F}(\omega_1,\ldots,\omega_5)$.

$$f(x_1,\ldots,x_5) = x_1 \oplus x_1x_2 \oplus x_1x_3 \oplus x_1x_4 \oplus x_3x_4 \oplus x_1x_5 \oplus x_2x_5.$$

$f \in \mathrm{PC}_5(4)$ because f satisfies the PC with respect to $\mathrm{V}_5 - \{(1,1,1,1,1)\}$. Let

$$f(0,1,x_3,x_4,x_5) = x_3x_4 \oplus x_5 \stackrel{\mathrm{def}}{=} g(x_3,x_4,x_5).$$

Table 2.4: $\hat{F}$ of Example 2.6

$(\omega_1, \omega_2, \omega_3, \omega_4, \omega_5)$	$\hat{F}$	$(\omega_1, \omega_2, \omega_3, \omega_4, \omega_5)$	$\hat{F}$
$(0,0,0,0,0)$	0	$(0,0,0,0,1)$	8
$(1,0,0,0,0)$	8	$(1,0,0,0,1)$	0
$(0,1,0,0,0)$	8	$(0,1,0,0,1)$	0
$(1,1,0,0,0)$	0	$(1,1,0,0,1)$	-8
$(0,0,1,0,0)$	8	$(0,0,1,0,1)$	0
$(1,0,1,0,0)$	0	$(1,0,1,0,1)$	8
$(0,1,1,0,0)$	0	$(0,1,1,0,1)$	-8
$(1,1,1,0,0)$	8	$(1,1,1,0,1)$	0
$(0,0,0,1,0)$	8	$(0,0,0,1,1)$	0
$(1,0,0,1,0)$	0	$(1,0,0,1,1)$	8
$(0,1,0,1,0)$	0	$(0,1,0,1,1)$	-8
$(1,1,0,1,0)$	8	$(1,1,0,1,1)$	0
$(0,0,1,1,0)$	0	$(0,0,1,1,1)$	-8
$(1,0,1,1,0)$	-8	$(1,0,1,1,1)$	0
$(0,1,1,1,0)$	-8	$(0,1,1,1,1)$	0
$(1,1,1,1,0)$	0	$(1,1,1,1,1)$	8

Then,

$$g(x_3, x_4, x_5) \oplus g(x_3, x_4, x_5 \oplus 1) \equiv 1,$$

and g does not satisfy the SAC. Thus, $f \notin \mathrm{SAC}_5(2)$. □

Lemma 2.12 [PLLGV91] Let $n \geqslant 3$ and $f \in \mathrm{B}_n$. Suppose that the nonlinear order of f is 2. f satisfies the SAC of order m such that $0 \leqslant m \leqslant n-2$ if and only if every variable x_i occurs in at least $(m+1)$ second order terms of the algebraic normal form of f. □

It is obvious that $\mathrm{SAC}_n(n-3) \subsetneq \mathrm{SAC}_n(0) = \mathrm{PC}_n(1)$. It is implicitly described in [PLLGV91] that $\mathrm{SAC}_n(n-2) \subsetneq \mathrm{PC}_n(n-1)$. For $\mathrm{SAC}_n(n-3)$ and $\mathrm{PC}_n(2)$, the following theorem holds.

Theorem 2.13 $\mathrm{SAC}_n(n-3) \not\subseteq \mathrm{PC}_n(2)$ for every $n \geqslant 3$.

(Proof) Let

$$g(x_1, \ldots, x_n) = \bigoplus_{1 \leqslant i < j \leqslant n, i \leqslant n-2} x_i x_j.$$

It is sufficient to show that $g \notin \mathrm{PC}_n(2)$ because $g \in \mathrm{SAC}_n(n-3)$ from Lemma 2.12. Since

$$g(x_1, \ldots, x_n) = \bigoplus_{1 \leqslant i < j \leqslant n-2} x_i x_j \oplus \bigoplus_{1 \leqslant k \leqslant n-2} x_k(x_{n-1} \oplus x_n),$$

for $a = (0, \ldots, 0, 1, 1)$,

$$g(x_1, \ldots, x_n) \oplus g(x_1 \oplus a_1, \ldots, x_n \oplus a_n) \equiv 0.$$

Hence, $g \notin \mathrm{PC}_n(2)$. □

Figure 2.3 and 2.4 show the relationships between the PC and the SAC. In the two figures, if a directed path exists from the set A to the set B, then the set A contains the set B.

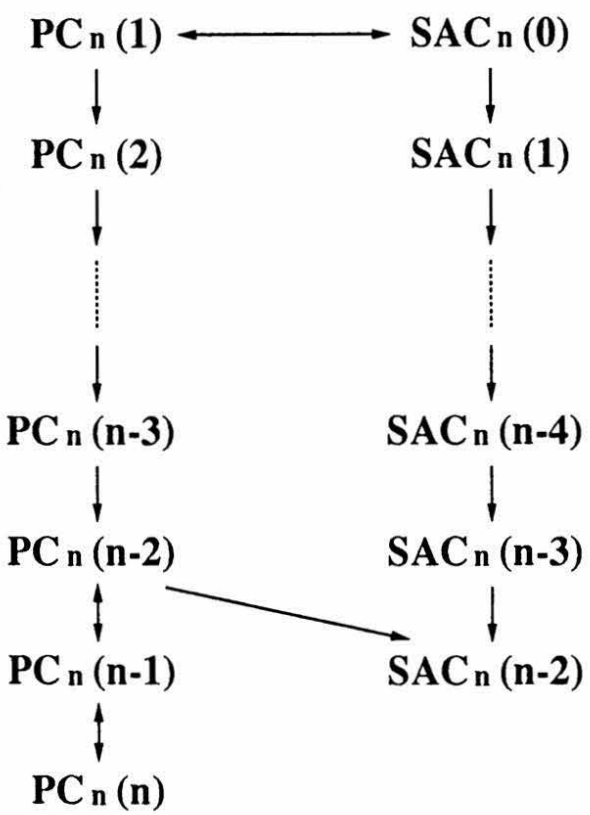

Figure 2.3: Relationships between the PC and the SAC (n is even)

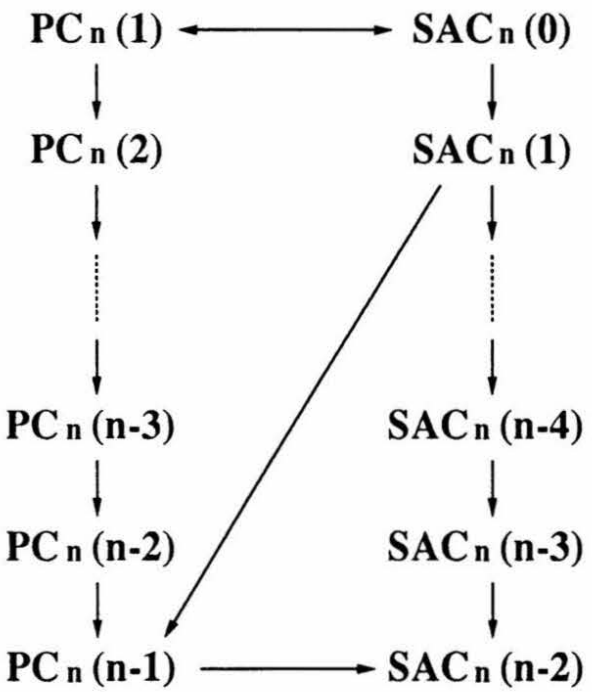

Figure 2.4: Relationships between the PC and the SAC (n is odd)

2.7 Conclusion

This chapter discussed the properties of nonlinearity criteria and the relationships among them. It focused on the PC, the SAC, and the nonlinearity.

First, we discussed Boolean functions with n variables satisfying the PC with respect to all but one elements in V_n, and those satisfying the PC with respect to all but linearly independent elements in V_n. Second, we discussed the construction of Boolean functions satisfying the PC with respect to all but one or all but three elements in V_n. Third, we showed that the Boolean functions with n variables satisfying the PC of degree $n-2$ are perfectly nonlinear for every even $n \geqslant 4$, and that they satisfy the PC with respect to all but one elements in V_n for every odd $n \geqslant 3$. Finally, Some relationships were presented between the PC and the SAC. Different relationships were proved between for Boolean functions with an even number of variables and for those with an odd number of variables.

Chapter 3

Complexity of Boolean Functions Satisfying the PC

3.1 Introduction

In this chapter, complexity of Boolean functions satisfying the PC is discussed on several computation models.

First, some relationships are presented between the unateness and the degree of the PC. It is shown that, for $n \geq 4$, every Boolean function with n variables satisfying the PC of degree 1 is unate in at most two of its variables and that, for $n \geq 4$, there exist Boolean functions with n variables that satisfy the PC of degree 1 and that are unate in two of their variables. The proof of the latter implies the method of construction for such functions. It is also shown that every Boolean function satisfying the PC of degree 2 is not unate in any one of its variables.

Second, inversion complexity of perfectly nonlinear Boolean functions is discussed. The optimal lower bound $\lfloor \log n \rfloor - 1$ is obtained for every perfectly nonlinear Boolean function with n variables constructed by the method of Maiorana[Rue91].

Third, it is mentioned that the formula size of every Boolean function with n variables which satisfies the PC of degree 1 is at least $n^2/4 - 1$. This lower bound is almost optimal for every perfectly nonlinear Boolean function.

Fourth, the area-time-square(AT^2) VLSI complexity[Ull84] of perfectly nonlinear Boolean functions with multiple outputs is discussed. The main result of this topic is that, for every perfectly nonlinear Boolean function with n inputs and $n/2$ outputs, each of whose output functions is constructed by the method of Maiorana, AT^2 complexity of any VLSI implementation is $\Omega(n^2)$.

Finally, the size of ordered binary decision diagrams(OBDDs)[Bry86] is considered. A relationship is presented between a combinatorial problem and the OBDD size of perfectly nonlinear Boolean functions in a subset of those each of whose output functions is constructed by the method of Maiorana. It is also mentioned that, for any variable ordering and for every perfectly nonlinear Boolean function with n inputs and $n/2$ outputs constructed by the method of Nyberg[Nyb91], there exist some output function of the perfectly nonlinear Boolean function such that the OBDD size of the output function is exponential in the number of its inputs.

Computation models considered in this chapter is presented in Section 3.2. Section 3.3 discusses perfectly nonlinear Boolean functions with multiple outputs. The unateness and the inversion complexity are discussed in Section 3.4. Section 3.5 mentions the formula size. The AT^2 complexity of VLSI circuits and the OBDD size are considered in Section 3.6 and Section 3.7, respectively.

3.2 Computation Models

3.2.1 Combinational Circuits and Formulae

A combinational circuit is defined as an acyclic directed graph whose nodes correspond to the gates, the input terminals or the output terminals of the circuit and whose edges correspond to the wires of the circuit. A basis is a set of operations of gates constructing combinational circuits, where a operation of each gate is a Boolean function with a single output.

A combinational circuit on a basis B is an acyclic directed graph with the following properties:

1. Nodes with fan-in(the number of incoming edges) zero are input

nodes. Either an input variable or a constant is assigned to each input node.

2. Nodes with fan-out(the number of leaving edges) zero and fan-in one are output nodes. An output variable is assigned to each output node.

3. The other nodes are computation nodes. A k-input function in B labels a computation node with fan-in k. The incoming edges of a computation node are numbered. If the output of the function labeling a computation node does not depend on the numbering of the inputs, we may withdraw the numbering of the incoming edges of the node.

A combinational circuit on a basis B is called a B-circuit.

The fan-in and the fan-out of a circuit are the maximum fan-in and the maximum fan-out of the computation nodes in the circuit, respectively.

A formula is defined to be a circuit whose fan-out is one.

A Boolean function $f \in \mathrm{B}_{n,m}$ can be computed by a combinational circuit C if C has exactly or more than n input nodes and there is a many-to-one correspondence between $x_1, \ldots, x_n, 0, 1$ and input nodes, C has exactly m output nodes and there is a one-to-one correspondence between $y_1, \ldots, y_m$ and output nodes, and, for every $(x_1, \ldots, x_n) \in \{0,1\}^n$, $(y_1, \ldots, y_m) = f(x_1, \ldots, x_n)$.

A basis B is complete if all the Boolean functions can be computed by circuits on the basis B. For example, $\{\vee, \wedge, \neg\}$ is a complete basis, while $\{\vee, \wedge\}$ is not complete.

3.2.2 VLSI Circuits

The grid model[Ull84] is adopted for VLSI circuits.

In the grid model, a rectangular grid is assumed. Wires run along the grid lines. The spacing of grid lines can be viewed as the minimum repetition rate at which wires on a certain layer can run and it is a fixed constant. There are one or more layers and the number of them is some fixed constant. Each layer has at most one wire on every grid line.

Circuit elements, such as input/output pads, contacts, logic elements and so on are located on the grid points. Two or more inputs or outputs can be fed to one input pad or one output pad, respectively. Wires that meet a grid point occupied by a circuit element are inputs or outputs of the element. If more wires are needed by a circuit element than can connect to a single grid point, the circuit element may be represented by a rectangle covering as many grid points as needed. The fan-in of a circuit element is assumed to be limited by a fixed constant, while the fan-out of a circuit element unbounded. It is also assumed that wires carry signals in only one direction.

All the circuits are assumed to be convex and the area of a circuit is defined to be that of the smallest rectangle whose sides are on grid lines and which covers the circuit.

The unit of time is defined. Each input/output appear on some input/output pad during a unit of time and signals propagate on wires in a unit of time. The computation time of a circuit is defined to be the number of units of time between the first input and the last output.

3.2.3 Ordered Binary Decision Diagrams

A binary decision diagram(BDD) is an acyclic directed graph that has one source node and two sink nodes. Each node except two sink nodes are labeled by an input variable and has two outgoing edges which are labeled by 0 and 1, respectively. Two sink nodes are labeled by 0 and 1, respectively. For each input variable that occurs as a label in a BDD, it appears at most once on each path from a source node to a sink node.

A BDD represents a Boolean function $f(x_1, \ldots, x_n) \in \mathrm{B}_n$ if, for every $(b_1, \ldots, b_n) \in \{0,1\}^n$, the path from the source node tracing each edge outgoing from x_i and labeled by b_i leads to the sink node labeled by $f(b_1, \ldots, b_n)$.

An ordered binary decision diagram(OBDD) is a BDD in which the order of the occurrence of the variables on each path are determined by a total ordering of the variables. If a variable x_i precedes a variable x_j in the total ordering, then x_i appears before x_j on every path that contains x_i and x_j.

3.2.4 Notations

When the computational complexity of Boolean functions is discussed, not each definite function but sequences of functions $\{f_n \mid f_n \in \mathrm{B}_{n,m} \text{ for } n \in \mathbf{N}\}$ are treated. The asymptotic behavior of the complexity of $\{f_n \mid f_n \in \mathrm{B}_{n,m} \text{ for } n \in \mathbf{N}\}$ is considered.

Definition 3.1 Let $p : \mathbf{N} \to \mathbf{R}$ and $q : \mathbf{N} \to \mathbf{R}$ such that $p(n) > 0$ and $q(n) > 0$ for large n.

- If there exist some constant $c > 0$ and $n_c > 0$ such that $p(n)/q(n) \leqslant c$ for every $n > n_c$, then $p = O(q)$.
- If, for any constant $\varepsilon > 0$, there exists some constant $n_\varepsilon > 0$ such that $p(n)/q(n) < \varepsilon$ for every $n > n_\varepsilon$, then $p = o(q)$.
- If there exist some constant $c > 0$ and $n_c > 0$ such that $q(n)/p(n) \leqslant c$ for every $n > n_c$, then $p = \Omega(q)$.
- If, for any constant $\varepsilon > 0$, there exists some constant $n_\varepsilon > 0$ such that $q(n)/p(n) < \varepsilon$ for every $n > n_\varepsilon$, then $p = \omega(q)$.
- If $p = O(q)$ and $q = O(p)$, then $p = \Theta(q)$.

□

3.3 Perfectly Nonlinear Boolean Functions with Multiple Outputs

The definition of perfectly nonlinear Boolean functions can be extended to those with multiple outputs.

Definition 3.2 [Nyb91] $f \in \mathrm{B}_{n,m}$ is perfectly nonlinear if and only if, for every $a \in \mathrm{V}_n$, $f(x) \oplus f(x \oplus a)$ is balanced, that is, for every $b \in \{0,1\}^m$,

$$|\{x \in \{0,1\}^n | f(x) \oplus f(x \oplus a) = b\}| = 2^{n-m}.$$

□

Nyberg [Nyb91] presented a necessary and sufficient condition for perfect nonlinearity.

Proposition 3.1 [Nyb91] $f = (f_1, \ldots, f_m) \in \mathrm{B}_{n,m}$ is perfectly nonlinear if and only if, for every $c = (c_1, \ldots, c_m) \in \mathrm{V}_m$,

$$c \cdot f = c_1 f_1 \oplus \cdots \oplus c_m f_m$$

is perfectly nonlinear. □

It is obvious from Proposition 3.1 that n is even if $f \in \mathrm{B}_{n,m}$ is perfectly nonlinear. The following proposition presents the upper bound of the number of outputs.

Proposition 3.2 [Nyb91] If $f = (f_1, \ldots, f_m) \in \mathrm{B}_{n,m}$ is perfectly nonlinear, then $m \leqslant n/2$. □

Perfectly nonlinear Boolean functions exist in $\mathrm{B}_{n,n/2}$. The upper bound in the above proposition is optimal.

A method of construction of perfectly nonlinear Boolean functions was presented by Maiorana [Rue91]. Let $g \in \mathrm{B}_k$ be any Boolean function and $\pi \in \mathrm{B}_{k,k}$ be any permutation. Then, $f \in \mathrm{B}_{2k}$ represented as

$$f(x, y) = \pi(x) \cdot y \oplus g(x)$$

is perfectly nonlinear, where $x = (x_1, \ldots, x_k)$ and $y = (y_1, \ldots, y_k)$.

Proposition 3.3 Let $n = 2k$. Let $f = (f_1, \ldots, f_m) \in \mathrm{B}_{n,m}$ and, for every i such that $1 \leqslant i \leqslant m$, f_i is represented as

$$f_i(x, y) = \pi_i(x) \cdot y \oplus g_i(x),$$

where $\pi_i \in \mathrm{B}_{k,k}$ is a permutation and $g_i \in \mathrm{B}_k$. Then, f is perfectly nonlinear if and only if, for every $c = (c_1, \ldots, c_m) \in \mathrm{V}_m$,

$$c_1 \pi_1 \oplus \cdots \oplus c_m \pi_m$$

is a permutation.

(Proof) Since

$$\begin{aligned} & c \cdot f(x,y) \\ &= c_1 f_1(x,y) \oplus \cdots \oplus c_m f_m(x,y) \\ &= c_1(\pi_1(x) \cdot y \oplus g_1(x)) \oplus \cdots \oplus c_m(\pi_m(x) \cdot y \oplus g_m(x)) \\ &= (c_1\pi_1(x) \oplus \cdots \oplus c_m\pi_m(x)) \cdot y \oplus (c_1 g_1(x) \oplus \cdots \oplus c_m g_m(x)), \end{aligned}$$

for every $c = (c_1, \ldots, c_m) \in \mathrm{V}_m$, if $c_1\pi_1 \oplus \cdots \oplus c_m\pi_m \in \mathrm{B}_{k,k}$ is a permutation, then $c \cdot f$ is perfectly nonlinear.

If f is perfectly nonlinear, then, for every $c = (c_1, \ldots, c_m) \in \mathrm{V}_m$, $c \cdot f$ is perfectly nonlinear. If $c \cdot f$ is perfectly nonlinear, then

$$\begin{aligned} & c \cdot f(x,y) \oplus c \cdot f(x, y \oplus b) \\ &= \bigoplus_{i=1}^{m} c_i f_i(x,y) \oplus \bigoplus_{i=1}^{m} c_i f_i(x, y \oplus b) \\ &= \bigoplus_{i=1}^{m} (c_i f_i(x,y) \oplus c_i f_i(x, y \oplus b)) \\ &= \bigoplus_{i=1}^{m} (c_i(\pi_i(x) \cdot y \oplus g_i(x)) \oplus c_i(\pi_i(x) \cdot (y \oplus b) \oplus g_i(x))) \\ &= (c_1\pi_1(x) \oplus \cdots \oplus c_m\pi_m(x)) \cdot b \end{aligned}$$

is balanced for every $b \in \mathrm{V}_k$. Thus, $c_1\pi_1 \oplus \cdots \oplus c_m\pi_m$ is a permutation for every $c = (c_1, \ldots, c_m) \in \mathrm{V}_m$. This completes the proof. □

Let $n = 2k$ and $m \leqslant k$. Let

$$\mathrm{P}_{n,m} = \left\{ f \;\middle|\; \begin{array}{l} f = (f_1, \ldots, f_m) \in \mathrm{PC}_{n,m}(n) \text{ and, for each } i \\ \text{such that } 1 \leq i \leq m,\ f_i(x,y) = \pi_i(x) \cdot y \oplus g_i(x), \\ \text{where } \pi_i \in \mathrm{B}_{k,k} \text{ is a permutation and } g_i \in \mathrm{B}_k. \end{array} \right\}.$$

For simplicity, $\mathrm{P}_{n,1} = \mathrm{P}_n$.

3.4 Unateness and Inversion Complexity

3.4.1 Unateness

We begin by defining the unate functions[Koh78].

$x_1\, x_2$	$x_1 \vee x_2$
$\overline{x_1}\, x_2$	$\overline{x_1} \vee x_2$
$x_1\, \overline{x_2}$	$x_1 \vee \overline{x_2}$
$\overline{x_1}\, \overline{x_2}$	$\overline{x_1} \vee \overline{x_2}$

Figure 3.1: Boolean functions in $\mathrm{PC}_2(2)$

Definition 3.3 A Boolean function $f(x_1, \ldots, x_n) \in \mathrm{B}_n$ is said to be positive(negative) in a variable x_i if there exists a disjunctive expression of f in which x_i appears in uncomplemented(complemented) form. If f is positive(negative) in all of its variables, then f is simply said to be positive(negative). □

Definition 3.4 A Boolean function $f(x_1, \ldots, x_n) \in \mathrm{B}_n$ is said to be unate in a variable x_i if f is positive or negative in x_i. If f is unate in all of its variables, then f is simply said to be unate. □

For example, $f(x_1, x_2, x_3) = x_1\, x_2 \vee \overline{x_2}\, \overline{x_3}$ is unate in x_1 and x_3 and not unate in x_2.

Proposition 3.4 Let $f \in \mathrm{B}_n$. $f(x_1, \ldots, x_n)$ is positive in x_i if and only if $f|_{x_i=0} \leqslant f|_{x_i=1}$. □

Proposition 3.5 Let $f \in \mathrm{B}_n$. $f(x_1, \ldots, x_n)$ is negative in x_i if and only if $f|_{x_i=0} \geqslant f|_{x_i=1}$. □

Figure 3.1 and 3.2 give all Boolean functions in $\mathrm{PC}_2(2)$ and $\mathrm{PC}_3(2)$, respectively. From these figures, it is easily observed that both $PC_2(2)$ and $PC_3(2)$ contain unate functions.

Suppose that $f(x_1, \ldots, x_n) \in \mathrm{B}_n$ is unate in x_n. Then, for every $(a_1, \ldots, a_{n-1}) \in \{0,1\}^{n-1}$, $f(a_1, \ldots, a_{n-1}, 0) = 0$ if $f(a_1, \ldots, a_{n-1}, 1) = 0$ or $f(a_1, \ldots, a_{n-1}, 1) = 0$ if $f(a_1, \ldots, a_{n-1}, 0) = 0$. This regularity does not seem compatible with the PC. In the following, this conjecture is shown to be correct for $f \in \mathrm{B}_n$ when $n \geq 4$. Before showing the results, several lemmas are presented.

For $f \in \mathrm{B}_{n,m}$ and $c \in \{0,1\}^m$, let

$$f^{-1}(c) = \{(x_1, \ldots, x_n) \mid f(x_1, \ldots, x_n) = c\}.$$

$x_1x_2 \vee x_1x_3 \vee x_2x_3$	$\overline{x_1}\,\overline{x_2} \vee \overline{x_1}\,\overline{x_3} \vee \overline{x_2}\,\overline{x_3}$
$x_1x_2 \vee x_1\overline{x_3} \vee x_2\overline{x_3}$	$\overline{x_1}\,\overline{x_2} \vee \overline{x_1}x_3 \vee \overline{x_2}x_3$
$x_1\overline{x_2} \vee x_1x_3 \vee \overline{x_2}x_3$	$\overline{x_1}x_2 \vee \overline{x_1}\,\overline{x_3} \vee x_2\overline{x_3}$
$\overline{x_1}x_2 \vee \overline{x_1}x_3 \vee x_2x_3$	$x_1\overline{x_2} \vee x_1\overline{x_3} \vee \overline{x_2}\,\overline{x_3}$
$x_1\overline{x_2}\,\overline{x_3} \vee \overline{x_1}x_2x_3$	$\overline{x_1}\,\overline{x_2} \vee x_1x_3 \vee x_2\overline{x_3}$
$x_1\overline{x_2}x_3 \vee \overline{x_1}x_2\overline{x_3}$	$\overline{x_1}\,\overline{x_2} \vee \overline{x_1}x_3 \vee x_1x_2$
$x_1x_2\overline{x_3} \vee \overline{x_1}\,\overline{x_2}x_3$	$\overline{x_1}x_2 \vee x_1x_3 \vee \overline{x_2}\,\overline{x_3}$
$\overline{x_1}\,\overline{x_2}\,\overline{x_3} \vee x_1x_2x_3$	$x_1\overline{x_2} \vee \overline{x_1}x_3 \vee x_2\overline{x_3}$

Figure 3.2: Boolean functions in $PC_3(2)$

Lemma 3.1 Let $n \geq 2$ and $f \in PC_n(1)$. Then, for every i such that $1 \leq i \leq n$, $f(x_1, \ldots, x_n)$ is positive in x_i if and only if $\hat{F}(0, \ldots, 0, \overset{i}{1}, 0, \ldots, 0) = 2^{n-1}$.

(Proof) Suppose that $i = n$. Since $f \in PC_n(1)$, $f(x_1, \ldots, x_{n-1}, 0) \oplus f(x_1, \ldots, x_{n-1}, 1)$ is balanced,that is,

$$|\{\langle x\rangle_{n-1} \mid f|_{x_n=0} \neq f|_{x_n=1}\}| = 2^{n-2}.$$

Thus,

$$\begin{aligned}
&\hat{F}(0, \ldots, 0, 1) \\
&= \sum_{x \in \{0,1\}^n} \hat{f}(x)(-1)^{x_n} \\
&= \sum_{\langle x\rangle_{n-1} \in \{0,1\}^{n-1}} (\hat{f}|_{x_n=0} - \hat{f}|_{x_n=1}) \\
&= 2|\{\langle x\rangle_{n-1} \mid f|_{x_n=0} = 0, f|_{x_n=1} = 1\}| - \\
&\quad 2|\{\langle x\rangle_{n-1} \mid f|_{x_n=0} = 1, f|_{x_n=1} = 0\}| \\
&= 2^{n-1} - 4|\{\langle x\rangle_{n-1} \mid f|_{x_n=0} = 1, f|_{x_n=1} = 0\}|.
\end{aligned}$$

Hence, $\hat{F}(0, \ldots, 0, 1) = 2^{n-1}$ if and only if $f(x)$ is positive in x_n. The same argument can be applied to the case where $1 \leq i \leq n-1$. □

The following lemma can be proved in the same way as Lemma 3.1.

Lemma 3.2 Let $n \geq 2$ and $f \in \mathrm{PC}_n(1)$. Then, for every i such that $1 \leq i \leq n$, $f(x_1, \ldots, x_n)$ is negative in x_i if and only if $\hat{F}(0, \ldots, 0, \overset{i}{1}, 0, \ldots, 0) = -2^{n-1}$. □

Lemma 3.3 Let $f \in \mathrm{B}_n$. Then, $f \in \mathrm{PC}_n(k)$ if and only if

$$\sum_{a \cdot \omega = 0} \hat{F}^2(\omega) = \sum_{a \cdot \omega = 1} \hat{F}^2(\omega) = 2^{2n-1}$$

for every $a \in \{0,1\}^n$ such that $1 \leq W(a) \leq k$.

(Proof) $f \in \mathrm{PC}_n(k)$ if and only if, for every $a \in \{0,1\}^n$ such that $1 \leq W(a) \leq k$,

$$C_f(a) = \frac{1}{2^n} \sum_{\omega \in \{0,1\}^n} \hat{F}^2(\omega)(-1)^{a \cdot \omega} = 0.$$

Thus,

$$\sum_{a \cdot \omega = 0} \hat{F}^2(\omega) = \sum_{a \cdot \omega = 1} \hat{F}^2(\omega).$$

The lemma holds because $\sum_{\omega \in \{0,1\}^n} \hat{F}^2(\omega) = 2^{2n}$ for every $f \in \mathrm{B}_n$. □

From Lemma 3.1 and 3.2, if $f(x_1, \ldots, x_n) \in \mathrm{PC}_n(1)$ is unate in x_i, then

$$\hat{F}^2(0, \cdots, 0, \overset{i}{1}, 0, \ldots, 0) = 2^{2n-2} = \frac{1}{4} \sum_{\omega \in \{0,1\}^n} \hat{F}^2(\omega).$$

From this fact, it is immediately derived that every Boolean function satisfying the PC of degree 1 is unate in at most 4 of its variables. A more strict result is given in the following.

Theorem 3.1 Let $n \geqslant 4$. If $f \in \mathrm{PC}_n(1)$, then f is unate in at most two of its variables.

(Proof) Without loss of generality, it can be assumed that $f(x_1, \ldots, x_n) \in \mathrm{PC}_n(1)$ is unate in x_1, x_2 and x_3. Then, from Lemma 3.1 and 3.2,

$$\begin{aligned}
&\hat{F}^2(1,0,0,0,\ldots,0) + \hat{F}^2(0,1,0,0,\ldots,0) + \hat{F}^2(0,0,1,0,\ldots,0) \\
&= 2^{2(n-1)} + 2^{2(n-1)} + 2^{2(n-1)} \\
&= 2^{2n-1} + 2^{2n-2}.
\end{aligned}$$

Since $f(x_1, \ldots, x_n) \in \mathrm{PC}_n(1)$, from Lemma 3.3, $\sum_{\omega_4=0} \hat{F}^2(\omega) = 2^{2n-1}$, which causes a contradiction. □

The optimality of the above result can also be proved. The following theorem presents an exact characterization of a Boolean function in $\mathrm{PC}_n(1)$ that is unate in two of its variables.

Theorem 3.2 Let $n \geqslant 4$ and $f \in \mathrm{PC}_n(1)$. $f(x_1, \ldots, x_n)$ is unate in x_i and x_j if and only if f satisfies one of the following two conditions.

- $\left|\hat{F}(\omega)\right| = \begin{cases} 2^{n-1} & \text{if } \omega \in \{2^{i-1}, 2^{j-1}, 2^n - 2^{i-1} - 1, 2^n - 2^{j-1} - 1\} \\ 0 & \text{otherwise} \end{cases}$
  and one or three of nonzero $\hat{F}(\omega)$'s are positive.
- $\left|\hat{F}(\omega)\right| = \begin{cases} 2^{n-1} & \text{if } \omega \in \{2^{i-1}, 2^{j-1}, 2^n - 1, 2^n - 2^{i-1} - 2^{j-1} - 1\} \\ 0 & \text{otherwise} \end{cases}$
  and one or three of nonzero $\hat{F}(\omega)$'s are positive.

(Proof) We prove the theorem only for the first condition. It can be proved in the same way for the second condition.

Suppose that $f \in \mathrm{PC}_n(1)$ is unate in x_1 and x_2. Then,

$$\hat{F}^2(1,0,0,\ldots,0) + \hat{F}^2(0,1,0,\ldots,0) = 2^{2(n-1)} + 2^{2(n-1)} = 2^{2n-1}.$$

Since $\sum_{\omega_i=0} \hat{F}^2(\omega) = 2^{2n-1}$ for every i such that $3 \leqslant i \leqslant n$,

$$\omega \neq \left\{ \begin{array}{ll} (1,0,0,\ldots,0), & (0,1,0,\ldots,0), \\ (1,0,1,\ldots,1), & (0,1,1,\ldots,1), \\ (0,0,1,\ldots,1), & (1,1,1,\ldots,1) \end{array} \right\} \Rightarrow \hat{F}(\omega) = 0.$$

Let

$$\begin{aligned}
&\hat{F}(1,0,0,\ldots,0)=\hat{F}_1, \quad \hat{F}(0,1,0,\ldots,0)=\hat{F}_0,\\
&\hat{F}(1,0,1,\ldots,1)=\hat{F}_{10}, \quad \hat{F}(0,1,1,\ldots,1)=\hat{F}_{01},\\
&\hat{F}(0,0,1,\ldots,1)=\hat{F}_{00}, \quad \hat{F}(1,1,1,\ldots,1)=\hat{F}_{11}.
\end{aligned}$$

Since $\sum_{\omega_i=0} \hat{F}^2(\omega) = \sum_{\omega_i=1} \hat{F}^2(\omega) = 2^{2n-1}$ for $i = 1, 2$,

$$\hat{F}_{00}^2 + \hat{F}_{01}^2 = \hat{F}_{10}^2 + \hat{F}_{11}^2 = \hat{F}_{00}^2 + \hat{F}_{10}^2 = \hat{F}_{01}^2 + \hat{F}_{11}^2 = 2^{2(n-1)}.$$

From Lemma 2.1, there are following two cases:

C-1. $\left|\hat{F}_{10}\right| = \left|\hat{F}_{01}\right| = 2^{n-1}$, $\left|\hat{F}_{00}\right| = \left|\hat{F}_{11}\right| = 0$,

C-2. $\left|\hat{F}_{00}\right| = \left|\hat{F}_{11}\right| = 2^{n-1}$, $\left|\hat{F}_{10}\right| = \left|\hat{F}_{01}\right| = 0$.

For C-1, let

$$\begin{aligned}
&b_1=(1,0,0,\ldots,0), \quad b_2=(0,1,0,\ldots,0)\\
&b_3=(1,0,1,\ldots,1), \quad b_4=(0,1,1,\ldots,1).
\end{aligned}$$

Then, $\left[\hat{f}\right]$ can be represented as

$$\begin{aligned}
\left[\hat{f}\right] &= \frac{1}{2^n}\left[\hat{F}\right] H_n\\
&= \frac{1}{2^n}\left(\hat{F}_1\left[\hat{l}_{b_1}\right] + \hat{F}_0\left[\hat{l}_{b_2}\right] + \hat{F}_{10}\left[\hat{l}_{b_3}\right] + \hat{F}_{01}\left[\hat{l}_{b_4}\right]\right).
\end{aligned}$$

Since $b_1 \oplus b_2 \oplus b_3 \oplus b_4 = (0,\ldots,0)$, for every $x \in \{0,1\}^n$, an even number of $\hat{l}_{b_1}(x)$, $\hat{l}_{b_2}(x)$, $\hat{l}_{b_3}(x)$ and $\hat{l}_{b_4}(x)$ are equal to 1, and the others are equal to -1. Thus, an odd number of $\hat{F}_0$, $\hat{F}_1$, $\hat{F}_{10}$ and $\hat{F}_{01}$ must be equal to 2^{n-1} and the others must be equal to -2^{n-1} since $f \in \mathrm{B}_n$.

Conversely, if an odd number of $\hat{F}_0$, $\hat{F}_1$, $\hat{F}_{10}$ and $\hat{F}_{01}$ are equal to 2^{n-1} and the others are equal to -2^{n-1}, then $f \in \mathrm{PC}_n(1)$ and f is unate in x_1 and x_2.

The same argument as the above one can be applicable to C-2. □

Corollary 3.1 Let $n \geqslant 4$. For every i, j such that $1 \leqslant i < j \leqslant n$, there are 16 $f(x_1,\ldots,x_n)$'s in $\mathrm{PC}_n(1)$ that are unate in x_i and x_j. □

The proof of Theorem 3.2 gives a method of construction for Boolean functions satisfying the PC of degree 1 and are unate in two of their variables. This method can construct every such functions.

Example 3.1 Four Boolean functions are given that are in $PC_4(1)$ and that are positive in x_1 and x_2. Let

$$\begin{aligned}\left[\hat{F}_0\right] &= [0,8,8,0,0,0,0,0,0,0,0,0,0,-8,8,0],\\ \left[\hat{F}_1\right] &= [0,8,8,0,0,0,0,0,0,0,0,0,0,8,-8,0].\end{aligned}$$

Then,

$$\begin{aligned}\left[\hat{f}_0\right] &= \frac{1}{2^4}\left[\hat{F}_0\right] H_4\\ &= [1,1,-1,-1,1,-1,1,-1,1,-1,1,-1,1,1,-1,-1],\\ \left[\hat{f}_1\right] &= \frac{1}{2^4}\left[\hat{F}_1\right] H_4\\ &= [1,-1,1,-1,1,1,-1,-1,1,1,-1,-1,1,-1,1,-1].\end{aligned}$$

Thus,

$$\begin{aligned}f_0(x_1,x_2,x_3,x_4) &= x_1\overline{x_3}x_4 \vee x_1x_3\overline{x_4} \vee x_2x_3x_4 \vee x_2\overline{x_3}\,\overline{x_4}\\ &= x_1(x_3 \oplus x_4) \vee x_2(x_3 \oplus x_4 \oplus 1),\\ f_1(x_1,x_2,x_3,x_4) &= x_1x_3x_4 \vee x_1\overline{x_3}\,\overline{x_4} \vee x_2\overline{x_3}x_4 \vee x_2x_3\overline{x_4}\\ &= x_1(x_3 \oplus x_4 \oplus 1) \vee x_2(x_3 \oplus x_4).\end{aligned}$$

The other two functions can be constructed in the same way as the above. They are

$$\begin{aligned}g_0(x_1,x_2,x_3,x_4) &= x_1x_2 \vee x_1\overline{x_3}x_4 \vee x_1x_3\overline{x_4} \vee x_2\overline{x_3}x_4 \vee x_2x_3\overline{x_4}\\ &= x_1x_2 \vee (x_1 \vee x_2)(x_3 \oplus x_4),\\ g_1(x_1,x_2,x_3,x_4) &= x_1x_2 \vee x_1x_3x_4 \vee x_1\overline{x_3}\,\overline{x_4} \vee x_2x_3x_4 \vee x_2\overline{x_3}\,\overline{x_4}\\ &= x_1x_2 \vee (x_1 \vee x_2)(x_3 \oplus x_4 \oplus 1).\end{aligned}$$

The truth tables of the above functions are presented in Figure 3.3. □

The following theorem is on non-unateness of the Boolean functions satisfying the PC of degree 2.

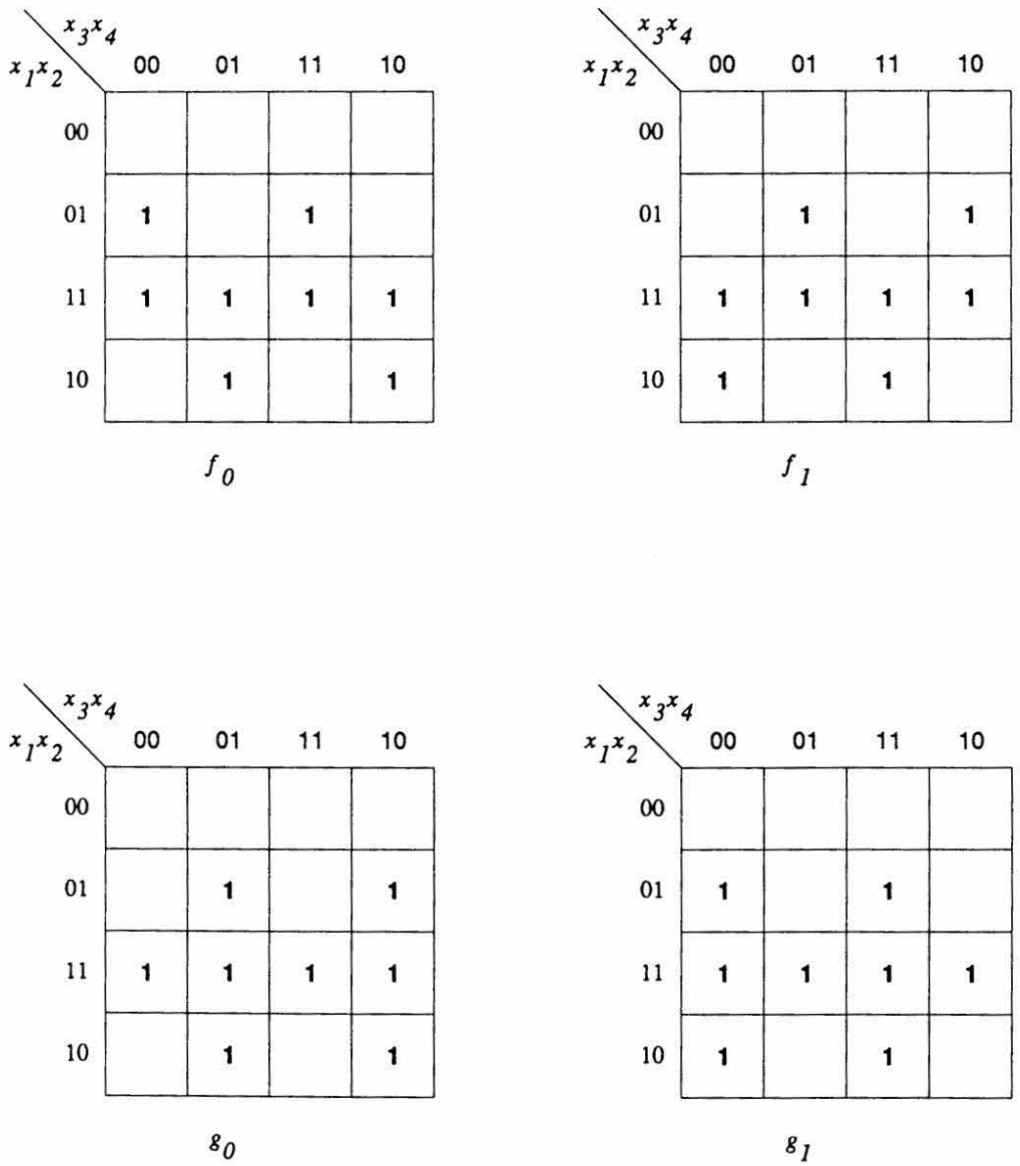

Figure 3.3: Functions in $\mathrm{PC}_4(1)$ and positive in x_1 and x_2

Theorem 3.3 Let $n \geqslant 4$. If $f \in \mathrm{PC}_n(2)$, then f is not unate in any one of its variables.

(Proof) Suppose that $f \in \mathrm{PC}_n(2)$ is unate in x_1. Then,

$$\hat{F}^2(1,0,\ldots,0) = 2^{2(n-1)}.$$

Let i, j be any integers such that $2 \leqslant i < j \leqslant n$. Since $f \in \mathrm{PC}_n(2)$,

$$\begin{aligned}
\sum_{\omega_i=0} \hat{F}^2(\omega) &= 2^{2n-1}, \\
\sum_{\omega_j=0} \hat{F}^2(\omega) &= 2^{2n-1}, \\
\sum_{\omega_i \oplus \omega_j = 1} \hat{F}^2(\omega) &= 2^{2n-1}.
\end{aligned}$$

Thus,

$$\left(\sum_{\omega_i=0} \hat{F}^2(\omega) + \sum_{\omega_j=0} \hat{F}^2(\omega) \right) - 2\hat{F}^2(1,0,\ldots,0) - \sum_{\omega_i \oplus \omega_j = 1} \hat{F}^2(\omega) = 0.$$

From this equation, if $\hat{F}(\omega) \neq 0$, then at most one of $\omega_2, \ldots, \omega_n$ is 0 or $\omega = (1,0,\ldots,0)$.

If $n = 4$, then, for every $f \in \mathrm{PC}_4(2)$, f is perfectly nonlinear and $\hat{F}^2(\omega) = 16$ for every $\omega \in \{0,1\}^4$, which is a contradiction.

For $n \geqslant 5$,

$$\begin{aligned}
&\sum_{\omega_i \oplus \omega_j = 1} \hat{F}^2(\omega) \\
&= \hat{F}^2(0,1,\ldots,1,\overset{i}{\check{0}},1,\ldots,1) + \hat{F}^2(0,1,\ldots,1,\overset{j}{\check{0}},1,\ldots,1) + \\
&\quad \hat{F}^2(1,1,\ldots,1,\overset{i}{\check{0}},1,\ldots,1) + \hat{F}^2(1,1,\ldots,1,\overset{j}{\check{0}},1,\ldots,1) \\
&= 2^{2n-1}.
\end{aligned}$$

Thus, from Lemma 2.1, 2.5, 2.6, for every $\omega \in \{0,1\}^n$ such that only one of $\omega_2, \ldots, \omega_n$ is 0 and $\hat{F}(\omega) \neq 0$, $\hat{F}^2(\omega) = 2^{2n-2}$. For every $\omega \in \{0,1\}^n$ such that only one of $\omega_2, \ldots, \omega_n$ is 0, since

$$\sum_{\omega_1=0} \hat{F}^2(\omega) = \sum_{\omega_1=1} \hat{F}^2(\omega) = 2^{2n-1}$$

and

$$\hat{F}^2(1,0,\ldots,0) = 2^{2(n-1)},$$

at most three of $\hat{F}^2(\omega)$'s are 2^{2n-2}, while, since

$$\sum_{\omega_2 \oplus \omega_3 = 1} \hat{F}^2(\omega) = \sum_{\omega_4 \oplus \omega_5 = 1} \hat{F}^2(\omega) = 2^{2n-1},$$

at least four of $\hat{F}^2(\omega)$'s are 2^{2n-2}. This causes a contradiction. Thus, the theorem has been proved. □

3.4.2 Inversion Complexity

This section shows that $\{\wedge, \vee, \neg\}$-circuits that compute a perfectly nonlinear Boolean function in P_n requires many $\neg$-gates.

It is easy to see that the fact that f is not unate in any one of its variables does not necessarily imply that f has high inversion complexity. For example, $f(x_1,\ldots,x_n) = x_1 \cdots x_n \vee \overline{x_1} \cdots \overline{x_n}$ can be computed with only one negation although f is not unate in any one of $x_1,\ldots,x_n$.

Definition 3.5 The inversion complexity[Mar58] of a Boolean function f, $I(f)$, is the smallest number of $\neg$-gates necessary to compute f by $\{\wedge, \vee, \neg\}$-circuits. □

Let $a = (a_1,\ldots,a_n)$, $b = (b_1,\ldots,b_n)$ and $a, b \in \{0,1\}^n$. $a \leqslant b$ if and only if $a_i \leqslant b_i$ for every i such that $1 \leqslant i \leqslant n$. $a < b$ if and only if $a \leqslant b$ and $a_i < b_i$ for some i such that $1 \leqslant i \leqslant n$.

Definition 3.6 [Mar58] Let $C = (\alpha_1,\ldots,\alpha_k)$ be a sequence such that $\alpha_i \in \{0,1\}^n$ for $1 \leq i \leq k$ and $\alpha_j < \alpha_{j+1}$ for $1 \leq j \leq k-1$. C is called sign-variable chain of length k of $f \in B_n$ if and only if

$$f(\alpha_i) = \begin{cases} 1 & \text{if } i \text{ is odd} \\ 0 & \text{if } i \text{ is even} \end{cases}$$

for every i such that $1 \leq i \leq k$. □

Definition 3.7 $alt(f)$ is the length of the longest sign-variable chain of f. □

Inversion complexity $I(f)$ is completely characterized by $alt(f)$.

Lemma 3.4 [Mar58] For every $f \in \mathrm{B}_n$,

$$I(f) = \begin{cases} 0 & \text{if } alt(f) = 0 \\ \lfloor \log_2 alt(f) \rfloor & \text{otherwise.} \end{cases}$$

□

The following proposition is trivial from the definition of alt.

Proposition 3.6 For every $f \in \mathrm{B}_n$, $I(f) \leqslant \lfloor \log_2(n+1) \rfloor$. □

Lemma 3.5 For every $f \in \mathrm{P}_n$, $I(f) \geq \lfloor \log_2 n \rfloor - 1$.

(Proof) Let $n = 2k$. $f(x,y) = \pi(x) \cdot y \oplus g(x)$, where $\pi \in \mathrm{B}_{k,k}$ be a permutation and $g \in \mathrm{B}_k$.

Since π is a permutation, for some $v \in \{0,1\}^k$, $\pi(v) = (1,\ldots,1)$ and

$$f(v,y) = y_1 \oplus \cdots \oplus y_k \oplus g(v).$$

Let $C = (\alpha^1,\ldots,\alpha^k)$ such that $\alpha^i = (v, \underbrace{1,\ldots,1}_{i}, 0,\ldots,0) \in \{0,1\}^{2k}$ for every i such that $1 \leq i \leq k$. If $g(v) = 0$, then

$$f(\alpha^i) = \begin{cases} 1 & \text{if } i \text{ is odd,} \\ 0 & \text{if } i \text{ is even.} \end{cases}$$

and C is a sign-variable chain of f. It also can be shown that there exists a sign-variable chain of length $k+1$ of f if $g(v) = 1$. This completes the proof. □

Theorem 3.4 For every $f \in \mathrm{P}_n$, $\lfloor \log_2 n \rfloor - 1 \leq I(f) \leq \lfloor \log_2(n+1) \rfloor$. □

The following two examples show that the bounds of Theorem 3.4 are optimal.

Example 3.2 For $f(x,y) = x_1y_1 \oplus \cdots \oplus x_ky_k$, $f \in \mathrm{P}_{2k}$ and $I(f) = \lfloor \log_2 k \rfloor$. □

Example 3.3 Let $n = 2k$ and $f \in \mathrm{B}_n$ such that

$$f(x,y) = 1 \oplus (x_1 \oplus \cdots \oplus x_k) \oplus (x_1y_1 \oplus \cdots \oplus x_ky_k).$$

For $1 \leqslant i \leqslant n+1$, let

$$\alpha_i = (\underbrace{1,\ldots,1}_{i-1},0,\ldots,0) \in \{0,1\}^n.$$

Then, $\alpha_j < \alpha_{j+1}$ for every j such that $1 \leqslant j \leqslant n$, and

$$f(\alpha_i) = \begin{cases} 1 & \text{if } i \text{ is odd} \\ 0 & \text{if } i \text{ is even.} \end{cases}$$

Thus, $alt(f) = n+1$ and every circuit that computes f requires $\lfloor \log_2(n+1) \rfloor$ negations. □

3.5 Formula Size

In this section, a lower bound on the formula size of any Boolean function in $\mathrm{PC}_n(1)$ is obtained with the use of the method of Krapchenko[Kra71].

Some notations are defined in the following definition.

Definition 3.8 [Weg87] Let $\mathrm{Q}, \mathrm{S} \subseteq \{0,1\}^n$ and $f \in \mathrm{B}_n$. Let $H(\mathrm{Q},\mathrm{S})$ be the set of neighbors in (Q,S), i.e.,

$$H(\mathrm{Q},\mathrm{S}) = \left\{ (q,s) \;\middle|\; \begin{array}{l} (q,s) \in (\mathrm{Q},\mathrm{S}) \text{ and there exists some } i \\ \text{such that } 1 \leq i \leq n,\ q_i \neq s_i \text{ and } q_j = s_j \\ \text{for every } j \text{ such that } 1 \leq j \leq n \text{ and } j \neq i \end{array} \right\}.$$

Let

$$K_{\mathrm{Q},\mathrm{S}} = \frac{|H(\mathrm{Q},\mathrm{S})|^2}{|\mathrm{Q}||\mathrm{S}|},$$
$$K(f) = \max\{K_{\mathrm{Q},\mathrm{S}} \mid \mathrm{Q} \subseteq f^{-1}(1), \mathrm{S} \subseteq f^{-1}(0)\}.$$

□

Lemma 3.6 [Weg87] For every Boolean function f, $L_{\mathrm{U}}(f) \geq K(f) - 1$, where $\mathrm{U} = \mathrm{B}_2 - \{\oplus, \equiv\}$. □

Theorem 3.5 For every $f \in \mathrm{PC}_n(1)$, $L_{\mathrm{U}}(f) \geq n^2/4 - 1$.

(Proof) Suppose that $f \in \mathrm{PC}_n(1)$. Then,

$$K(f) \geq K_{f^{-1}(1),f^{-1}(0)} = \frac{(n2^{n-2})^2}{|f^{-1}(1)|(2^n - |f^{-1}(1)|)} \geq n^2/4.$$

This completes the proof. □

The lower bound in Theorem 3.5 is almost optimal even for perfectly nonlinear Boolean functions. For the perfectly nonlinear Boolean function f in Example 3.2, $L_{\mathrm{U}}(f) \leq n^2/2 - 1$ when n is a power of 2.

For the same f, $L_{\mathrm{B}_2}(f) = n - 1$. On the formula model, $\oplus$ and $\equiv$ is essential for the efficient computation of the Boolean functions satisfying the PC.

3.6 VLSI Complexity

This section gives some results on AT^2 complexity of VLSI circuits computing $f \in \mathrm{P}_{n,m}$.

Lemma 3.7 [Ull84] Let I be any set of input variables and C a VLSI chip. If no more than one third of the inputs in I are fed into an input pad of C, then a line with a single jog can be drawn that divides C into two parts each of which includes between one third and two thirds of the inputs in I. □

The proofs of the results make use of the information flow argument[Ull84]. Let C be a VLSI chip of area A and time T computing some function, and let L be a line on C satisfying the condition of the above lemma. If some of the outputs on a side of L depend on some of the inputs on the other side, then some amount of information must be transferred across L during the computation. If the amount of the information is proved to be at least I, then $\lambda^{-1}(\sqrt{A} + \lambda)T \geq I$. Thus, $AT^2 = \Omega(I^2)$.

Theorem 3.6 Let $n = 2k$ and $f \in \mathrm{P}_{2k,k}$. For every VLSI circuit that computes f,

$$AT^2 = \Omega(n^2).$$

(Proof) Let $f = (f_1, \ldots, f_k) \in \mathrm{P}_{n,k}$ such that, for $1 \leqslant i \leqslant k$,

$$f_i(x, y) = \pi_i(x) \cdot y \oplus g_i(x),$$

where $\pi_i = (\pi_{i,1}, \ldots, \pi_{i,k}) \in \mathrm{B}_{k,k}$ is a permutation and $g_i \in \mathrm{B}_k$.

If some input pad accepts $k/3 = n/6$ or more inputs, then $T \geqslant n/6$ and $AT^2 = \Omega(n^2)$.

Suppose that each input pad accept less than $k/3$ inputs. Then, Lemma 3.7 implies that the chip can be split in two parts by a line of length at most $\sqrt{A} + \lambda$ so that each of the sides has between $1/3$ and $2/3$ of the inputs $y = (y_1, \ldots, y_k)$, where λ is the spacing of grid lines. Without loss of generality, it can be assumed that the left side of the chip contains at least half of the outputs. Let $r = \lceil k/3 \rceil$. Choose r of the inputs y on the right side of the chip and r of the outputs on the left side of the chip. Without loss of generality, we can assume that they are $y_1, \ldots, y_r$ and $f_1, \ldots, f_r$.

For $c \in \mathrm{V}_r$, let

$$\pi_0^c = (\pi_{0,1}^c, \ldots, \pi_{0,k}^c) = c_1\pi_1 \oplus \cdots \oplus c_r\pi_r.$$

Since π_0^c is a permutation for every $c \in \mathrm{V}_r$,

$$\left|\{x \mid \pi_{0,1}^c(x) = \cdots = \pi_{0,r}^c(x) = 0\}\right| = 2^{k-r}.$$

Thus,

$$\left|\{x \mid \exists c(\pi_{0,1}^c(x) = \cdots = \pi_{0,r}^c(x) = 0)\}\right| \leqslant 2^{k-r}(2^r - 1) = 2^k - 2^{k-r}.$$

There exists $a \in \{0,1\}^k$ such that, for any $c \in \mathrm{V}_r$, there exists some j such that $1 \leqslant j \leqslant r$ and $\pi_{0,j}^c(a) \neq 0$. Hence,

$$(\pi_{1,1}(a), \ldots, \pi_{1,r}(a)), \ldots, (\pi_{r,1}(a), \ldots, \pi_{r,r}(a))$$

are linearly independent.

Thus, for every pair of $\langle y \rangle_r, \langle y' \rangle_r \in \{0,1\}^r$ such that $\langle y \rangle_r \neq \langle y' \rangle_r$,

$$(f_1(a, \langle y \rangle_r, b), \ldots, f_r(a, \langle y \rangle_r, b)) \neq (f_1(a, \langle y' \rangle_r, b), \ldots, f_r(a, \langle y' \rangle_r, b)),$$

where $b \in \{0,1\}^{k-r}$.

From the above discussions, during the computation, at least r bits of information must be transferred across the line splitting the chip. Hence, $\lambda^{-1}(\sqrt{A} + \lambda)T \geqslant r$ and $AT^2 = \Omega(n^2)$. □

The proof of Theorem 3.6 can be easily extended to prove the following corollary.

Corollary 3.2 Let ϵ be a constant such that $0 < \epsilon \leqslant 1/2$. Let $f \in \mathrm{P}_{n,\lfloor \epsilon n \rfloor}$. For every VLSI circuit that computes f,

$$AT^2 = \Omega((\epsilon n)^2).$$

□

It is considered to be more realistic that input/output pads are located on the boundary of a VLSI chip. The following theorem can also be proved in the same way as Theorem 3.6.

Theorem 3.7 Any VLSI circuit that receives inputs on the boundary of the chip and outputs $f \in \mathrm{P}_{n,m}$ requires

$$AT^2 = \Omega(nm).$$

□

The above result has some implication for the VLSI implementation of cryptographic transformations. For nonlinear elements in secure cryptographic transformations, their area-time-square VLSI complexity is expected to grow at least in proportion to the number of inputs and that of the outputs.

3.7 OBDD Size

In this section, we consider the OBDD size of perfectly nonlinear Boolean functions in a subset of $\mathrm{P}_{n,m}$.

For every perfectly nonlinear Boolean function, its outputs change independently of each other if any change of inputs occurs. This independence induces a conjecture that, for every perfectly nonlinear Boolean function, there exist no variable ordering such that all of the output functions can be represented by OBDD's of small size.

We consider $f = (f_1, \ldots, f_k) \in \mathrm{P}_{2k,k}$ such that, for every i such that $1 \leq i \leq k$,

$$f_i(x, y) = (P_i x^{\mathrm{T}}) \cdot y \oplus g_i(x),$$

where P_i is a $k \times k$ $\{0,1\}$-matrix and $g_i \in \mathrm{B}_k$. For every $(c_1, \ldots, c_k) \in \mathrm{V}_k$, $c_1 P_1 \oplus \cdots \oplus c_k P_k$ is non-singular. Let $\mathrm{P}^M_{2k,k}$ be the set of such functions.

Let A be a matrix. Let $rank(A)$ denote the rank of A and let $A[i_1, \ldots, i_a][j_1, \ldots, j_b]$ denote an $a \times b$ matrix whose (u, v)-element is (i_u, j_v)-element of A, where $i_p \neq i_q$ and $j_s \neq j_t$ for every p, q and s, t such that $1 \leq p < q \leq a$ and $1 \leq s < t \leq b$, respectively.

The following theorem gives a relationship between the OBDD size of a perfectly nonlinear Boolean function in $\mathrm{P}^M_{2k,k}$ and a combinatorial problem.

Theorem 3.8 Let $f = (f_1, \ldots, f_k) \in \mathrm{P}^M_{2k,k}$ and, for each i such that $1 \leq i \leq k$,

$$f_i(x, y) = (P_i x^{\mathrm{T}}) \cdot y \oplus g_i(x),$$

where P_i is a $k \times k$ $\{0,1\}$-matrix and $g_i \in \mathrm{B}_k$. Let

$$r(f) \stackrel{\mathrm{def}}{=} \min_{\substack{1 \leq i_1, \ldots, i_{\lceil k/2 \rceil} \leq k \\ 1 \leq j_1, \ldots, j_{\lceil k/2 \rceil} \leq k}} \max_{1 \leq l \leq k} rank(P_l[i_1, \ldots, i_{\lceil k/2 \rceil}][j_1, \ldots, j_{\lceil k/2 \rceil}]).$$

Then, for any variable ordering, there exists some i such that $1 \leq i \leq k$ and the OBDD size of f_i is $\Omega(2^{r(f)})$.

(Proof) For each i such that $1 \leq i \leq k$, $f_i(x, y) = y P_i x^{\mathrm{T}} \oplus g_i(x)$.

In a variable ordering of (x, y), suppose that x_{u_p} precedes x_{u_q} and y_{v_p} precedes y_{v_q} if $1 \leq p < q \leq k$.

Suppose that $x_{u_{\lceil k/2 \rceil}}$ precedes $y_{v_{\lceil k/2 \rceil}}$. Then, for some d such that $1 \leq d \leq k$, $rank(P'_d) \geq r(f)$, where $P'_d = P_d[v_{\lfloor k/2 \rfloor+1}, \ldots, v_k][u_1, \ldots, u_{\lceil k/2 \rceil}]$. Thus,

$$\left|\{a \mid a = P'_d b^{\mathrm{T}}, b \in \{0,1\}^{\lceil k/2 \rceil}\}\right| \geq 2^{r(f)}.$$

Let $x' = (x_{u_1}, \ldots, x_{u_{\lceil k/2 \rceil}})$ and $y' = (y_{v_{\lfloor k/2 \rfloor+1}}, \ldots, y_{v_k})$. Let $g' = g|_{x_{u_{\lceil k/2 \rceil+1}}=\cdots=x_{u_k}=0}$. Then,

$$f'_d = y' P'_d (x')^{\mathrm{T}} \oplus g'(x')$$

is a sub-function of f_d constructed by substituting 0's for $x_{u_{\lceil k/2 \rceil+1}}, \ldots, x_{u_k}$ and $y_{v_1}, \ldots, y_{v_{\lfloor k/2 \rfloor}}$. For the variable ordering $(x_{u_1}, \ldots, x_{u_{\lceil k/2 \rceil}}, y_{v_{\lceil k/2 \rceil+1}}, \ldots, y_{v_k})$, the OBDD size of f'_d is $\Omega(2^{r(f)})$. Thus, the OBDD size of f_d is $\Omega(2^{r(f)})$ if $x_{u_{\lceil k/2 \rceil}}$ precedes $y_{v_{\lceil k/2 \rceil}}$.

For the case where $y_{v_{\lceil k/2 \rceil}}$ precedes $x_{u_{\lceil k/2 \rceil}}$, the theorem can be proved in the same way. □

We conjecture that $r(f) \geq \lfloor k/4 \rfloor$ for every $f \in \mathrm{P}^M_{2k,k}$.

A method of construction was proposed by Nyberg[Nyb91] for perfectly nonlinear Boolean functions in $\mathrm{P}^M_{2k,k}$. Let R be a $k \times k$ matrix which express a state transition function of a linear feedback shift register(LFSR) of length k with a primitive feedback polynomial. Let $f = (f_1, \ldots, f_k) \in \mathrm{B}_{2k,k}$ such that, for each i such that $1 \leqslant i \leqslant k$,

$$f_i(x, y) = (R^{i-1} x^{\mathrm{T}}) \cdot y \oplus g_i(x),$$

where $g_i \in \mathrm{B}_k$. Then, $f \in \mathrm{P}^M_{2k,k}$.

Example 3.4 Let p be a polynomial over $GF(2)$ such that

$$p(x) = x^4 + x + 1.$$

$p(x)$ is a primitive polynomial. The LFSR of p is in Figure 3.4. The state transition function of the LFSR is described by R such that

$$R = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

□

The following theorem can be immediately derived from the proof of the OBDD size of integer multiplication in [Bry86].

Theorem 3.9 Suppose that $f = (f_1, \ldots, f_k) \in \mathrm{P}^M_{2k,k}$ is constructed with the method of Nyberg. Then, for any variable ordering, there exists some i such that $1 \leq i \leq k$ and the OBDD size of f_i is $\Omega(2^{k/16})$. □

3.8 Conclusion

This section have discussed the complexity of nonlinear Boolean functions on several computation models.

First, unateness and inversion complexity have been discussed. It has been shown that there exist Boolean functions which satisfy the PC of degree 1 and unate in two of their variables and that Boolean functions satisfying the PC of degree 2 is not unate in any one of its variables. Inversion complexity of perfectly nonlinear Boolean functions has been proved to be almost maximum.

Second, we have mentioned that the formula size of the Boolean functions satisfying the PC of degree 1 is at least $n^2/4 - 1$ and that the lower bound is almost optimal even for perfectly nonlinear Boolean functions.

Third, the area-time tradeoff of VLSI has been discussed. For every Boolean function in a subset of perfectly nonlinear ones with multiple outputs, AT^2 of VLSI computing the function has been proved to be $\Omega(n^2)$.

Finally, the size of OBDD has been considered. For any variable ordering, every perfectly nonlinear Boolean functions constructed by

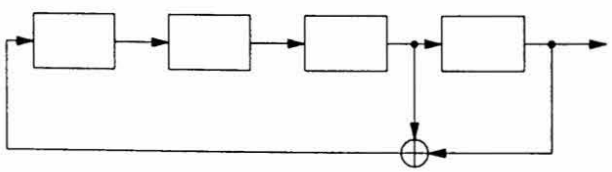

Figure 3.4: The linear feedback shift register of p

the method of Nyberg has some output function such that the size of OBDD representing the function is exponential in the number of its inputs.

Chapter 4

Circuit Complexity of Homogeneous Boolean Functions and Their Slices

4.1 Introduction

In this chapter, we consider the circuit complexity of slice Boolean functions and homogeneous Boolean functions.

It is known that for any k-homogeneous Boolean function, its $(k+1)$-th slice is not much more difficult to compute than its k-th slice[Dun86]. On the other hand, it has been proved that there exist k-homogeneous Boolean functions such that the monotone complexity of their k-th slices is much larger than that of their $u(> k)$-th slices [Weg86]. One topic is an improvement of the latter result. The optimal lower bound is obtained on the monotone circuit size complexity of the k-th slices for constant k.

The other topic is the homogeneous Boolean functions whose circuit size complexity and monotone circuit size complexity are almost equal. For these homogeneous Boolean functions with n variables, their monotone circuit size complexity is larger than their circuit size complexity at most by a constant factor and an additive term of $O(n(\log n)^2)$. Hence, a lower bound of $\omega(n(\log n)^2)$ on the monotone circuit size complexity implies the same lower bound on the circuit size complexity.

In the next section, we define complexity measures of Boolean functions on the circuit model. Section 4.3 presents the definitions of slice Boolean functions and homogeneous Boolean functions, and show their basic properties. In Section 4.4, we present the k-homogeneous Boolean functions whose k-th slice is much more difficult to compute than their other slices. Section 4.5 presents a class of k-homogeneous Boolean functions such that negation is powerless for computing them. Section 4.6 is the conclusion of this chapter.

4.2 Preliminaries

4.2.1 Complexity Measures for Combinational Circuits

The complexity of combinational circuits is measured by their size, depth, and so on. In this chapter, we discuss only about the size of the circuits. The size of a circuit C, $Size(C)$, is the number of the gates in the circuit C.

From the practical point of view, it may be necessary to bound the fan-in and the fan-out of gates by some constants. Bounding the fan-in of the gates in a circuit by some constant r means that all the Boolean functions in the basis of the circuit have exactly or less than r inputs and that the basis is finite.

For any finite and complete bases B and B′, each Boolean function in B′ can be computed by a constant size circuit on B. Thus, for any Boolean function f, each gate in a B′-circuit computing f can be replaced by a constant size B-circuit.

Proposition 4.1 Let B and B′ be finite and complete bases. For any Boolean function f, if a B-circuit C computes f, then there exists a B′-circuit C' computing f such that

$$Size(C') \leqslant c \cdot Size(C),$$

where c is a constant depending on B and B′. □

For any f, the size of a circuit computing f is never getting larger as the fan-out of the circuit is getting larger. The following proposition

says that if a Boolean function f can be computed by an unbounded fan-out combinational circuit C on a finite basis B, f can also be computed by a combinational circuit C' with fan-out t (≥ 2) on B whose size is larger than those of the circuit C at most by a constant factor.

Proposition 4.2 [HKP84] Let f be a Boolean function and B be a finite basis. For any unbounded fan-out B-circuit C computing f, a B-circuit C' with fan-out t (≥ 2) computing f can be constructed such that

$$SIZE(C') \leqslant c \cdot SIZE(C) + \frac{q-1}{t-1},$$

where q is the number of output nodes and c is a constant which depend on B and t. □

4.2.2 Circuit Complexity

The computational complexity of a Boolean function is measured by the complexity of circuits computing the Boolean function. A Boolean function can be computed by infinitely many circuits varying in complexity. The computational complexity of a Boolean function is measured with the complexity of optimal circuits computing the Boolean function.

Definition 4.1 The circuit size complexity of a Boolean function f on a basis B, $C_{\mathrm{B}}(f)$, is the smallest size of B-circuits computing f. □

In this chapter, we discuss the circuit complexity of single-output Boolean functions on finite bases with unbounded fan-out.

Proposition 4.3 [Sha49] Let $\mathrm{E}_n \subseteq \mathrm{B}_n$ and $|\mathrm{E}_n| = 2^{e(n)}$ for some $e : \mathbf{N} \to \mathbf{N}$. Then, for almost all Boolean functions, g's, in E_n,

$$C(g) = \Omega\left(\frac{e(n)}{\log e(n)}\right).$$

□

In the above proposition, "almost all" means that

$$\lim_{n\to\infty} |\{g \in \mathrm{E}_n \mid C(g) = \Omega(e(n)/\log e(n))\}|/|\mathrm{E}_n| = 1.$$

This proposition means that, for almost all sequences $\{g_n \mid g_n \in \mathrm{E}_n \text{ for } n \in \mathrm{N}\}$, the circuit size complexity of each g_n in the sequence is $\Omega(e(n)/\log e(n))$.

4.3 Slice Boolean Functions and Homogeneous Boolean Functions

4.3.1 Monotone Boolean Functions and Monotone Circuit Complexity

Homogeneous Boolean functions and slice Boolean functions are in the class of monotone Boolean functions. We first define monotone Boolean functions and show their properties.

Definition 4.2 Let $a, b \in \{0,1\}^n$. A Boolean function $f \in \mathrm{B}_n$ is monotone if and only if $a \leqslant b$ implies $f(a) \leqslant f(b)$. □

M_n denotes the set of all n-input monotone Boolean functions. M_n coincides with the set of positive Boolean functions.

Monotone Boolean functions are also defined as Boolean functions which can be computed by circuits on the basis $\mathrm{M}_2 = \{\vee, \wedge, 0, 1\}$. The M_2-circuits are called monotone circuits. The circuit size complexity of monotone Boolean functions on the basis M_2 is called monotone circuit size complexity. For any monotone Boolean function f, $C_{\mathrm{M}_2}(f)$ is denoted as $C_m(f)$.

We define pseudo-complements of monotone Boolean functions. First, we define standard circuits.

Definition 4.3 A standard circuit is a monotone circuit with negated inputs permitted. □

If $f(x_1, \ldots, x_n)$ is monotone, for each variable x_i, some monotone Boolean function pc_i replaces $\overline{x_i}$ of standard circuits computing f.

Definition 4.4 Let $f \in \mathrm{M}_n$. A Boolean function $pc_i(x_1, \ldots, x_n) \in \mathrm{M}_n$ is a *pseudo-complement* for x_i with respect to $f(x_1, \ldots, x_n)$ if a circuit obtained by replacing the input $\overline{x_i}$ by $pc_i(x_1, \ldots, x_n)$ of any standard circuit computing f still computes f. □

For $f \in \mathrm{M}_n$, consider a size optimal B_2-circuit computing f. The size of the circuit is $C(f)$. From Proposition 4.1, for this circuit, a circuit S on the basis $\{\vee, \wedge, \neg\}$ computing f can be constructed whose size is $O(C(f))$. A standard circuit computing f can be constructed by applying De Morgan's Laws to this circuit S. The size of this standard circuit is at most twice larger than that of the circuit S, and it is $O(C(f))$. Since the size of size optimal standard circuits computing $f \in \mathrm{M}_n$ is $O(C(f))$, the following proposition follows.

Proposition 4.4 For $f \in \mathrm{M}_n$, let $pc_i \in \mathrm{M}_n$ be a pseudo-complement for x_i with respect to f. Then,

$$C(f) \leqslant C_m(f) \leqslant O(C(f)) + C_m(pc_1, \ldots, pc_n).$$

$C_m(pc_1, \ldots, pc_n) = C_m(pc)$, where pc is an n-input n-output monotone Boolean function such that $pc(x_1, \ldots, x_n) = (y_1, \ldots, y_n)$ and $y_i = pc_i(x_1, \ldots, x_n)$ for every i such that $1 \leqslant i \leqslant n$. □

For some monotone Boolean functions, their monotone circuit complexity is much larger than their circuit complexity. For a bipartite perfect matching Boolean function, its circuit size complexity is polynomial in the number of the input variables, while its monotone circuit size complexity is super-polynomial, that is, no polynomial size monotone circuit computes the bipartite perfect matching Boolean function[Raz85]. In general, the lower bound on $C_m(f)$ does not imply any lower bound on $C(f)$.

From Proposition 4.4, for any monotone Boolean function f, if f has pseudo-complements easy to compute, then its monotone circuit complexity is not much larger than its circuit complexity. In this case, if $\omega(C_m(pc_1, \ldots, pc_n))$ lower bound on $C_m(f)$ can be proved, it implies the same lower bound on $C(f)$.

Slice Boolean functions have pseudo-complements that are easy to compute.

4.3.2 Slice Boolean Functions

Definition 4.5 $f \in \mathrm{B}_n$ is a k-slice Boolean function if and only if

$$f(x_1, \ldots, x_n) = \begin{cases} 0 & \text{if } x_1 + \cdots + x_n < k \\ 1 & \text{if } x_1 + \cdots + x_n > k. \end{cases}$$

□

S_n^k denotes the set of k-slice Boolean functions with n variables.

T_n^k is a k-threshold function with n variables such that $T_n^k(x_1, \ldots, x_n) = 1$ if and only if k or more than k of $x_1, \ldots, x_n$ are equal to 1.

Definition 4.6 Let $f \in \mathrm{B}_n$. $f^k = f \wedge T_n^k \vee T_n^{k+1}$ is the k-th slice of f.

□

From the above definition, for any $f \in \mathrm{B}_n$,

$$f^k(x_1, \ldots, x_n) = \begin{cases} 0 & \text{if } x_1 + \cdots + x_n < k \\ f(x_1, \ldots, x_n) & \text{if } x_1 + \cdots + x_n = k \\ 1 & \text{if } x_1 + \cdots + x_n > k. \end{cases}$$

f^k is a k-slice Boolean function. For any k-slice Boolean function $g \in \mathrm{B}_n$, there exists some $f \in \mathrm{B}_n$ such that $f^k = g$. S_n^k is equal to the set of k-th slices of Boolean functions in B_n.

The following proposition is on the pseudo-complements of k-slice Boolean functions.

Proposition 4.5 [Weg87] Let $f \in \mathrm{S}_n^k$ and $1 \leqslant k \leqslant n$. A pseudo-complement for x_i of the k-th slice of $f(x_1, \ldots, x_n)$ is

$$T_{n-1}^k(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n).$$

□

The following proposition is immediate from Proposition 4.4 and Proposition 4.5.

Proposition 4.6 For every $f \in \mathrm{B}_n$ and every k such that $1 \leqslant k \leqslant n$,

$$C_m(f^k) \leqslant O(C(f^k)) + C_m(T_{n-1}^k(X_1), \ldots, T_{n-1}^k(X_n)),$$

where $X_i = (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$ for every i such that $1 \leqslant i \leqslant n$.

□

Proposition 4.7 [Weg85, Val86] For every k such that $1 \leqslant k \leqslant n$,

$$C_m(T^k_{n-1}(X_1), \ldots, T^k_{n-1}(X_n)) = O(n \min\{k, n-k, (\log n)^2\}).$$

□

It is shown in Proposition 4.5 and 4.7 that slice Boolean functions have pseudo-complements easy to compute. For any $f \in \mathrm{B}_n$, if $C_m(f^k) = \omega(n(\log n)^2)$, then $C(f^k) = \Theta(C_m(f^k))$. Thus, if $\omega(n(\log n)^2)$ lower bound on $C_m(f^k)$ is proved, then it implies the same lower bound on $C(f^k)$.

We show some relationships between computational complexity of a Boolean function and that of its slices. First, we show the upper bounds on the complexity of threshold functions.

Proposition 4.8 Let $1 \leqslant k \leqslant n$.

1. $C(T^k_n) = O(n)$ for every k.
2. If k or $n-k$ are constant, then $C_m(T^k_n) = O(n)$, otherwise $C_m(T^k_n) = O(n \log n)$.
3. $C(T^1_n, \ldots, T^n_n) = O(n)$.
4. [AKS83] $C_m(T^1_n, \ldots, T^n_n) = O(n \log n)$.

□

Proposition 4.9 For every $f \in \mathrm{B}_n$,

1. $C(f) \leqslant C(f^0, \ldots, f^n) + O(n)$,
2. $C(f^0, \ldots, f^n) \leqslant C(f) + O(n)$,
3. if f is monotone, $C_m(f^k) \leqslant C_m(f) + O(n \cdot \min\{k, n-k, \log n\})$,
4. $C(f^k) \leqslant C(f) + O(n)$.

□

From the part 1 and 2 of Proposition 4.9, if $f \in \mathrm{B}_n$ depends on all of its n variables, then $C(f) = \Theta(C(f^0, \ldots, f^n))$. If f is difficult to compute, then some of its slices are difficult, and vice versa.

4.3.3 Homogeneous Boolean Functions

Definition 4.7 $f \in \mathrm{M}_n$ is a k-homogeneous Boolean function if and only if all the prime implicants of f contain exactly k variables. □

H_n^k denotes the set of k-homogeneous Boolean functions with n variables. $\left|\mathrm{H}_n^k\right| = 2^{{}_n\mathrm{C}_k} - 1$ since the number of the products of k positive literals is ${}_n\mathrm{C}_k$ and each k-homogeneous Boolean function is a nonempty disjunction of such products. From Proposition 4.3, the following proposition can be proved.

Proposition 4.10 For almost all Boolean functions f's in H_n^k,

$$\begin{aligned} C(f) &= \Omega({}_n\mathrm{C}_k / \log {}_n\mathrm{C}_k), \\ C_m(f) &= \Omega({}_n\mathrm{C}_k / \log {}_n\mathrm{C}_k). \end{aligned}$$

□

Proposition 4.11 Let $\mathrm{H}_{n,m}^k$ be a set of n-input m-output Boolean functions such that, for each $f = (f_1, \ldots, f_m) \in \mathrm{H}_{n,m}^k$, $f_i \in \mathrm{H}_n^k$ for every i such that $1 \leqslant i \leqslant m$. Let $C_m(\mathrm{H}_{n,m}^k) = \max\{C_m(f) \mid f \in \mathrm{H}_{n,m}^k\}$. Then,

1. [Sav76] $C_m(\mathrm{H}_{n,m}^1) = O(n^2/\log n)$,
2. $C_m(\mathrm{H}_{n,n}^k) \leqslant n \cdot C_m(\mathrm{H}_n^k)$,
3. $C_m(\mathrm{H}_n^k) \leqslant C_m(\mathrm{H}_{n,n}^{k-1}) + 2n - 1$,
4. [Weg87] for $k \geq 2$, $C_m(\mathrm{H}_n^k) = O(n^k/\log n)$.

□

From the above proposition, the lower bound in Proposition 4.10 is optimal for constant $k \geq 2$.

4.4 Circuit Complexity of Slices of Homogeneous Boolean Functions

In this section, we compare the computational complexity of the k-th slice with that of the u-th ($u > k$) slice of some f in H_n^k. The following two results have been obtained before.

Proposition 4.12 [Dun86] For every $f \in \mathrm{H}_n^k$ and every c such that $1 \leqslant c \leqslant n-k$,

$$C_m(f^{k+c}) \leqslant n \cdot C_m(f^{k+c-1}) + O(n^2).$$

□

Proposition 4.13 [Weg86] Let $2 \leqslant k < n$. There exists some $f \in \mathrm{H}_n^k$ such that

$$C_m(f^k) = \Omega\left(\frac{{}_{n-1}\mathrm{C}_{k-1}}{\log {}_{n-1}\mathrm{C}_{k-1}}\right),$$

and, for every u such that $k < u \leqslant n$,

$$C_m(f^u) = O(n \log n).$$

□

Proposition 4.12 shows that, for every $f \in \mathrm{H}_n^k$, the monotone circuit size complexity of f^u is not much larger than that of f^{u-1} for every u such that $k < u \leqslant n$. Proposition 4.13 claims that there exists some $f \in \mathrm{H}_n^k$ such that the monotone circuit size complexity of f^u is much smaller than that of f^k for $k < u \leqslant n$.

Proposition 4.13 can be improved.

Theorem 4.1 For every k such that $2 \leqslant k < n$, there exists some $f \in \mathrm{H}_n^k$ such that

$$C_m(f^k) = \Omega\left(\frac{{}_n\mathrm{C}_k}{\log {}_n\mathrm{C}_k}\right),$$

and for every u such that $k < u \leqslant n$, $C_m(f^u) = O(n \log n)$. □

The lower bound in Theorem 4.1 is optimal if k is constant from the part 4 of Proposition 4.11 and $C_m(f^k) \leqslant C_m(f) + O(n \log n)$.

For every k-slice and k-homogeneous Boolean function $f \in \mathrm{H}_n^k \cap \mathrm{S}_n^k$, $C_m(f^u) = O(n \log n)$ since

$$f^u = f \wedge T_n^u \vee T_n^{u+1} = T_n^u$$

for every u such that $k < u \leqslant n$. And $f = f^k$. Wegener[Weg86] obtained Proposition 4.13 from Proposition 4.3 by proving $\left|\mathrm{H}_n^k \cap \mathrm{S}_n^k\right| = 2^{n-1}\mathrm{C}_{k-1}$. $\left|\mathrm{H}_n^k \cap \mathrm{S}_n^k\right| = 2^n\mathrm{C}_k$ is proved in this section.

We first prove two lemmas of the theorem.

Definition 4.8 Let $\mathrm{N} = \{1, 2, \ldots, n\}$. For every k such that $1 \leqslant k \leqslant n$,

$$\mathrm{Q}_k \stackrel{\mathrm{def}}{=} \{\{i_1, \ldots, i_k\} \mid \{i_1, \ldots, i_k\} \subseteq \mathrm{N}\},$$

and, for every l such that $1 \leqslant l \leqslant n-1$ and $\mathrm{P} \subseteq \mathrm{Q}_l$,

$$\begin{aligned} Aug(\mathrm{P}) \stackrel{\mathrm{def}}{=} \{\{j_1, \ldots, j_{l+1}\} \mid & \{j_1, \ldots, j_{l+1}\} \subseteq \mathrm{N}, \\ & \text{and for some } \{i_1, \ldots, i_l\} \in \mathrm{P}, \\ & |\{j_1, \ldots, j_{l+1}\} - \{i_1, \ldots, i_l\}| = 1\}. \end{aligned}$$

□

The next lemma shows an upper bound on the minimum number of the elements in $\mathrm{P} \subseteq \mathrm{Q}_k$ such that $Aug(\mathrm{P}) = \mathrm{Q}_{k+1}$, that is, for every element I in Q_{k+1}, there exists some element I′ in P such that $|\mathrm{I}-\mathrm{I}'| = 1$. We define I′ conceals I if $|\mathrm{I} - \mathrm{I}'| = 1$.

We divide N into two sets N_1 and N_2 such that $\mathrm{N}_1 \cap \mathrm{N}_2 = \phi$ and $\mathrm{N}_1 \cup \mathrm{N}_2 = \mathrm{N}$. For every element in Q_k, there exists some i such that the element has i elements in N_1 and $(k-i)$ elements in N_2. Any element $\mathrm{I}' \in \mathrm{Q}_{k+1}$ such that $|\mathrm{I}' \cap \mathrm{N}_1| = i+1$ and $|\mathrm{I}' \cap \mathrm{N}_2| = k-i$ is concealed by some element $\mathrm{I} \in \mathrm{Q}_k$ such that $|\mathrm{I} \cap \mathrm{N}_1| = i$ and $|\mathrm{I} \cap \mathrm{N}_2| = k-i$. And any element $\mathrm{I}'' \in Q_{k+1}$ such that $|\mathrm{I}'' \cap \mathrm{N}_1| = i$ and $|\mathrm{I}'' \cap \mathrm{N}_2| = k-i+1$ is concealed by some such element in Q_k.

Lemma 4.1 Let n and k be integers such that $2 \leqslant k < n$. Then,

$$\min\{|\mathrm{P}| \mid \mathrm{P} \subseteq \mathrm{Q}_k \text{ and } Aug(\mathrm{P}) = Q_{k+1}\}$$
$$\leqslant \begin{cases} \frac{1}{2}\left({}_n\mathrm{C}_k + (-1)^{k/2}{}_{n/2}\mathrm{C}_{k/2}\right) & \text{if } n \text{ and } k \text{ are even} \\ \frac{1}{2}{}_n\mathrm{C}_k + {}_{n/2}\mathrm{C}_k & \text{if } n \text{ is even and } k \text{ is odd} \\ \frac{1}{2}\left({}_n\mathrm{C}_k + (-1)^{k/2}{}_{\lfloor n/2 \rfloor}\mathrm{C}_{k/2}\right) & \text{if } n \text{ is odd and } k \text{ is even} \\ \frac{1}{2}\left({}_n\mathrm{C}_k + (-1)^{\lfloor k/2 \rfloor}{}_{\lfloor n/2 \rfloor}\mathrm{C}_{\lfloor k/2 \rfloor}\right) & \\ \quad + {}_{\lfloor n/2 \rfloor}\mathrm{C}_k & \text{if } n \text{ and } k \text{ are odd.} \end{cases}$$

(Proof) Let $\mathrm{N}_1 = \{1, \ldots, \lfloor n/2 \rfloor\}$, and $\mathrm{N}_2 = \{\lfloor n/2 \rfloor + 1, \ldots, n\}$. Suppose that $2 \leqslant k \leqslant \lfloor n/2 \rfloor$. Let P be constructed in the following way;

1. when k is even, $\mathrm{P} = \mathrm{P}_0 \cup \cdots \cup \mathrm{P}_{k/2}$, where

$$P_j = \{\{i_1, \ldots, i_k\} \subseteq \mathrm{N} \mid |\{i_1, \ldots, i_k\} \cap \mathrm{N}_1| = 2j\}$$

for every j such that $0 \leqslant j \leqslant k/2$,

2. when k is odd, $\mathrm{P} = \mathrm{P}_0 \cup \cdots \cup \mathrm{P}_{\lceil k/2 \rceil}$, where

$$P_j = \{\{i_1, \ldots, i_k\} \subseteq \mathrm{N} \mid |\{i_1, \ldots, i_k\} \cap \mathrm{N}_1| = 2j\}$$

for every j such that $0 \leqslant j \leqslant \lfloor k/2 \rfloor$, and

$$P_{\lceil k/2 \rceil} = \{\{i_1, \ldots, i_k\} \subseteq \mathrm{N} \mid |\{i_1, \ldots, i_k\} \cap \mathrm{N}_1| = k\}.$$

We show that $Aug(\mathrm{P}) = \mathrm{Q}_{k+1}$. For $m = 0, 1, \ldots, k+1$, let

$$\mathrm{T}_m = \{\{i_1, \ldots, i_{k+1}\} \subseteq \mathrm{N} \mid |\{i_1, \ldots, i_{k+1}\} \cap \mathrm{N}_1| = m\}.$$

Obviously, $\mathrm{Q}_{k+1} = \mathrm{T}_0 \cup \cdots \cup \mathrm{T}_{k+1}$. $Aug(\mathrm{P}_j) = \mathrm{T}_{2j} \cup \mathrm{T}_{2j+1}$ for every j such that $0 \leqslant j \leqslant \lfloor k/2 \rfloor$, and $\mathrm{T}_{k+1} \subseteq Aug(\mathrm{P}_{\lceil k/2 \rceil})$ when k is odd. Thus,

$$\begin{aligned} Aug(\mathrm{P}) &= Aug(\mathrm{P}_0) \cup \cdots \cup Aug(\mathrm{P}_{\lceil k/2 \rceil}) \\ &= \mathrm{T}_0 \cup \cdots \cup \mathrm{T}_{k+1} \\ &= \mathrm{Q}_{k+1}. \end{aligned}$$

Next, we evaluate the number of the elements in P. When k is even, $|\mathrm{P}_j| = {}_{\lfloor n/2 \rfloor}\mathrm{C}_{2j} \cdot {}_{\lceil n/2 \rceil}\mathrm{C}_{k-2j}$ for every j such that $0 \leqslant j \leqslant k/2$ and $\mathrm{P}_i \cap \mathrm{P}_j = \phi$ if $i \neq j$. Thus,

$$|\mathrm{P}| = \sum_{j=0}^{k/2} {}_{\lfloor n/2 \rfloor}\mathrm{C}_{2j} \cdot {}_{\lceil n/2 \rceil}\mathrm{C}_{k-2j}.$$

When k is odd, $|\mathrm{P}_j| = {}_{\lfloor n/2 \rfloor}\mathrm{C}_{2j} \cdot {}_{\lceil n/2 \rceil}\mathrm{C}_{k-2j}$ for every j such that $0 \leqslant j \leqslant \lfloor k/2 \rfloor$ and $\left|\mathrm{P}_{\lceil k/2 \rceil}\right| = {}_{\lfloor n/2 \rfloor}\mathrm{C}_k$. $\mathrm{P}_i \cap \mathrm{P}_j = \phi$ if $i \neq j$. Thus,

$$|\mathrm{P}| = \sum_{j=0}^{\lfloor k/2 \rfloor} {}_{\lfloor n/2 \rfloor}\mathrm{C}_{2j} \cdot {}_{\lceil n/2 \rceil}\mathrm{C}_{k-2j} + {}_{\lfloor n/2 \rfloor}\mathrm{C}_k.$$

For the calculation of $|\mathrm{P}|$, see Appendix.

For $k > \lfloor n/2 \rfloor$, it is clear that we obtain the same results from the construction of P. □

The following lemma shows an upper bound on the minimum number of prime implicants of k-slice and k-homogeneous Boolean functions. For every $f \in \mathrm{B}_n$, $PI(f)$ denotes the number of prime implicants of f.

Lemma 4.2 For every k such that $2 \leqslant k < n$, there exists some $f \in \mathrm{H}_n^k$ such that

$$\begin{aligned} f \wedge T_n^{k+1} &= T_n^{k+1}, \\ |PI(f)| &\leqslant \left(\frac{1}{2} + \frac{1}{2^k}\right) {}_n\mathrm{C}_k + o({}_n\mathrm{C}_k). \end{aligned}$$

(Proof) From Lemma 4.1, there exists $\mathrm{P} \subseteq \mathrm{Q}_k$ such that

$$\begin{aligned} Aug(\mathrm{P}) &= \mathrm{Q}_{k+1}, \\ |\mathrm{P}| &\leqslant \left(\frac{1}{2} + \frac{1}{2^k}\right) {}_n\mathrm{C}_k + o({}_n\mathrm{C}_k). \end{aligned}$$

For such P, consider the following $f \in \mathrm{H}_n^k$.

$$f(x_1, \ldots, x_n) = \bigvee_{\{i_1, \ldots, i_k\} \in \mathrm{P}} x_{i_1} \cdots x_{i_k}.$$

Then, it is clear that $|PI(f)| = |\mathrm{P}|$. And since $Aug(\mathrm{P}) = \mathrm{Q}_{k+1}$, for any product $x_{j_1} \cdots x_{j_k} x_{j_{k+1}}$, there exists some prime implicant $x_{i_1} \cdots x_{i_k}$ of f such that $x_{j_1} \cdots x_{j_{k+1}} \leqslant x_{i_1} \cdots x_{i_k}$, which implies

$$f(x_1, \ldots, x_n) \wedge T_n^{k+1}(x_1, \ldots, x_n) = T_n^{k+1}(x_1, \ldots, x_n).$$

□

Now, we prove Theorem 4.1.

Proof of Theorem 4.1: From Lemma 4.2, there exists some $h \in \mathrm{H}_n^k$ such that

$$\begin{aligned} h \wedge T_n^{k+1} &= T_n^{k+1}, \\ |PI(h)| &\leqslant \left(\frac{1}{2} + \frac{1}{2^k}\right) {}_n\mathrm{C}_k + o({}_n\mathrm{C}_k). \end{aligned}$$

Let $\mathrm{G}_n = \{f \mid f \in \mathrm{H}_n^k \text{ and } h \leqslant f\}$. Then, since $k \geqslant 2$,

$$|\mathrm{G}_n| \geqslant 2^{(\frac{1}{2} - \frac{1}{2^k}) {}_n\mathrm{C}_k - o({}_n\mathrm{C}_k)} \geqslant 2^{\frac{1}{4} {}_n\mathrm{C}_k - o({}_n\mathrm{C}_k)}.$$

Thus, from Proposition 4.3, for almost all $f \in \mathrm{G}_n$,

$$C_m(f^k) = \Omega\left(\frac{{}_n\mathrm{C}_k}{\log {}_n\mathrm{C}_k}\right).$$

For every $f \in \mathrm{G}_n$ and $k < u \leqslant n$, since $h \leqslant f$,

$$f^u = f \wedge T_n^u \vee T_n^{u+1} = T_n^u,$$

which implies

$$C_m(f^u) = C_m(T_n^u) = O(n \log n).$$

4.5 Homogeneous Boolean Functions for Which Negation Is Powerless

In this section, we present a set of homogeneous Boolean functions whose monotone circuit complexity is almost equal to their circuit complexity.

For every $f \in \mathrm{H}_n^k$, there exists some $J \in \mathrm{M}_{n+1}$ such that

$$f(x_1, \ldots, x_n) = J(f^k, x_1, \ldots, x_n).$$

It implies $C_m(f) \leqslant C_m(f^k) + C_m(J)$. Since $C(f) \leqslant C_m(f)$ and $C_m(f^k) \leqslant O(C(f)) + O(n(\log n)^2)$,

$$C(f) \leqslant C_m(f) \leqslant O(C(f)) + C_m(J) + O(n(\log n)^2).$$

Hence, if J is easy to compute by some monotone circuit, we can conclude that the circuit size complexity and the monotone circuit size complexity of f are almost equal. All Boolean functions in $\mathrm{R}_n^{k,c} \subseteq \mathrm{H}_n^k$ defined in the following definition have J which is easy to be computed by monotone circuits.

Definition 4.9 Let $X_1 = (x_1, \ldots, x_{\lceil n/2 \rceil})$, $X_2 = (x_{\lceil n/2 \rceil + 1}, \ldots, x_n)$. Let $n \geqslant 6$, $3 \leqslant k \leqslant \lfloor n/2 \rfloor$, and c be a constant such that $1 \leqslant c \leqslant k/3$. $f \in \mathrm{R}_n^{k,c}$ if and only if $f \in \mathrm{H}_n^k$ and there exist $2c$ integers $p_1, \ldots, p_c$ and $q_1, \ldots, q_c$ such that

1. $p_1 \geqslant 2$,
2. $q_i \geqslant 0$ for every i such that $1 \leqslant i \leqslant c$,
3. $p_j + q_j + 3 \leqslant p_{j+1}$ for every j such that $1 \leqslant j \leqslant c-1$,
4. $p_c + q_c \leqslant k - 1$,
5. for every j such that $j \in \{0, 1, \ldots, k+1\} - \bigcup_{i=1}^{c} \{p_i, p_i+1, \ldots, p_i+q_i\}$

$$T^j_{\lceil n/2 \rceil}(X_1) \wedge T^{k+1-j}_{\lfloor n/2 \rfloor}(X_2) \leqslant f(x_1, \ldots, x_n),$$

6. for every i such that $1 \leqslant i \leqslant c$ and for every l such that $-1 \leqslant l \leqslant q_i$,

$$PI(f) \cap PI(T^{p_i+l}_{\lceil n/2 \rceil}(X_1) \wedge T^{k-(p_i+l)}_{\lfloor n/2 \rfloor}(X_2)) = \emptyset.$$

□

Let $T^{k_1}_{X_1}$ and $T^{k_2}_{X_2}$ denote $T^{k_1}_{\lceil n/2 \rceil}(X_1)$ and $T^{k_2}_{\lfloor n/2 \rfloor}(X_2)$, respectively. The following theorem guarantees that $\mathrm{R}^{k,c}_n \neq \emptyset$.

Theorem 4.2 $\mathrm{R}^{k,c}_n \neq \emptyset$.

(Proof) Let

$$f(x_1, \ldots, x_n) = \bigvee_{u=0}^{c} \left(T^{3u}_{X_1} \wedge T^{k-3u}_{X_2} \right) \vee \bigvee_{v=3c+1}^{k} \left(T^{v}_{X_1} \wedge T^{k-v}_{X_2} \right).$$

It is shown that $f \in \mathrm{R}^{k,c}_n$.

For the above f, let $p_i = 3i - 1$, $q_i = 0$ for every i such that $i = 1, \ldots, c$. Then, it is clear that $p_1, \ldots, p_c$ and $q_1, \ldots, q_c$ satisfy 1 through 4 in Definition 4.9.

For 5 in Definition 4.9, since $j \in \{0, \ldots, k+1\} - \{p_1, \ldots, p_c\}$, the following three cases can be considered;

C-1. $j = 3i$ for every i such that $i = 0, 1, \ldots, c-1$,

C-2. $j = 3i + 1$ for every i such that $i = 0, 1, \ldots, c-1$,

C-3. $3c \leqslant j \leqslant k+1$.

For C-1. For every i such that $0 \leqslant i \leqslant c-1$, since $f(x) \vee T^{3i}_{X_1} \wedge T^{k-3i}_{X_2} = f(x)$,

$$\begin{aligned} & f(x) \wedge \left(T^{3i}_{X_1} \wedge T^{k+1-3i}_{X_2} \right) \\ &= f(x) \wedge \left(T^{3i}_{X_1} \wedge T^{k+1-3i}_{X_2} \right) \vee \left(T^{3i}_{X_1} \wedge T^{k-3i}_{X_2} \right) \wedge \left(T^{3i}_{X_1} \wedge T^{k+1-3i}_{X_2} \right) \\ &= f(x) \wedge \left(T^{3i}_{X_1} \wedge T^{k+1-3i}_{X_2} \right) \vee T^{3i}_{X_1} \wedge T^{k+1-3i}_{X_2} \\ &= T^{3i}_{X_1} \wedge T^{k+1-3i}_{X_2}. \end{aligned}$$

This implies that $T_{X_1}^{3i} \wedge T_{X_2}^{k+1-3i} \leqslant f(x)$. C-2 and C-3 can be proved in the same way.

For 6 in Definition 4.9, since

$$PI(f) = \bigcup_{u=0}^{c} PI\left(T_{X_1}^{3u} \wedge T_{X_2}^{k-3u}\right) \cup \bigcup_{v=3c+1}^{k} PI\left(T_{X_1}^{v} \wedge T_{X_2}^{k-v}\right),$$

for every i such that $1 \leqslant i \leqslant c$,

$$\begin{aligned} PI(f) \cap PI\left(T_{X_1}^{3i-2} \wedge T_{X_2}^{k-(3i-2)}\right) &= \emptyset, \\ PI(f) \cap PI\left(T_{X_1}^{3i-1} \wedge T_{X_2}^{k-(3i-1)}\right) &= \emptyset. \end{aligned}$$

Hence $f \in \mathrm{R}_n^{k,c}$. □

Monotone circuits computing $f \in \mathrm{R}_n^{k,c}$ can be constructed from monotone circuits computing f^k and additional $O(n \log n)$ gates computing Boolean functions in M_2. The monotone circuit size complexity of $f \in \mathrm{R}_n^{k,c}$ is larger than that of its k-th slice at most by the additive term of $O(n \log n)$.

For the proof of the following theorem, we received a hint from [Dun89].

Theorem 4.3 For every $f \in \mathrm{R}_n^{k,c}$, $C_m(f) \leqslant C_m(f^k) + O(n \log n)$.

(Proof) Let $\mathrm{D} = \{0, 1, \ldots, k+1\} - \bigcup_{i=1}^{c} \{p_i, \ldots, p_i + q_i\}$.

$$\begin{aligned} f^k(x) &= f(x) \wedge T_n^k(x) \vee T_n^{k+1}(x) \\ &= f(x) \vee T_n^{k+1}(x) \\ &= f(x) \vee \bigvee_{u=0}^{k+1} T_{X_1}^{u} \wedge T_{X_2}^{k+1-u} \\ &= f(x) \vee \bigvee_{v \in \mathrm{D}} T_{X_1}^{v} \wedge T_{X_2}^{k+1-v} \vee \bigvee_{i=1}^{c} \bigvee_{j=0}^{q_i} T_{X_1}^{p_i+j} \wedge T_{X_2}^{k+1-(p_i+j)}. \end{aligned}$$

From the fifth condition in Definition 4.9, since $f(x) = 1$ if $T_{X_1}^{v} \wedge T_{X_2}^{k+1-v} = 1$ for every $v \in \mathrm{D}$,

$$f^k(x) = f(x) \vee \bigvee_{i=1}^{c} \bigvee_{j=0}^{q_i} T_{X_1}^{p_i+j} \wedge T_{X_2}^{k+1-(p_i+j)}.$$

Thus,

$$f^k(x) \wedge \left(T_{X_1}^{p_l+q_l+1} \vee T_{X_2}^{k+2-p_l}\right) =$$
$$f(x) \wedge \left(T_{X_1}^{p_l+q_l+1} \vee T_{X_2}^{k+2-p_l}\right) \vee$$
$$\bigvee_{i=1}^{c} \bigvee_{j=0}^{q_i} T_{X_1}^{p_i+j} \wedge T_{X_2}^{k+1-(p_i+j)} \wedge \left(T_{X_1}^{p_l+q_l+1} \vee T_{X_2}^{k+2-p_l}\right).$$

For the right-hand side of the above equation, the next three equations hold.

1. $f(x) \wedge \left(T_{X_1}^{p_l+q_l+1} \vee T_{X_2}^{k+2-p_l}\right) = f(x)$.

2. For every $p_m \neq p_l$,

$$\left(\bigvee_{j=0}^{q_m} T_{X_1}^{p_m+j} \wedge T_{X_2}^{k+1-(p_m+j)}\right) \wedge \left(T_{X_1}^{p_l+q_l+1} \vee T_{X_2}^{k+2-p_l}\right)$$
$$= \bigvee_{j=0}^{q_m} T_{X_1}^{p_m+j} \wedge T_{X_2}^{k+1-(p_m+j)} \wedge \left(T_{X_1}^{p_l+q_l+1} \vee T_{X_2}^{k+2-p_l}\right)$$
$$= \bigvee_{j=0}^{q_m} T_{X_1}^{p_m+j} \wedge T_{X_2}^{k+1-(p_m+j)}.$$

3. $$\bigvee_{j=0}^{q_l} T_{X_1}^{p_l+j} \wedge T_{X_2}^{k+1-(p_l+j)} \wedge \left(T_{X_1}^{p_l+q_l+1} \vee T_{X_2}^{k+2-p_l}\right)$$
$$= \bigvee_{j=0}^{q_l} T_{X_1}^{p_l+q_l+1} \wedge T_{X_2}^{k+1-(p_l+j)} \vee \bigvee_{j=0}^{q_l} T_{X_1}^{p_l+j} \wedge T_{X_2}^{k+2-p_l}.$$

First, the equation 1 is proved. For every i such that $1 \leqslant i \leqslant c$ and for every r such that $-1 \leqslant r \leqslant q_i$,

$$PI(f) \cap PI(T_{X_1}^{p_i+r} \wedge T_{X_2}^{k-(p_i+r)}) = \emptyset.$$

Thus, for each prime implicant t of f, there exists some s such that $s \in \{0, 1, \ldots, k\} - \bigcup_{i=1}^{c} \{p_i - 1, \ldots, p_i + q_i\}$ and $t = x_{1,1} \cdots x_{1,s} x_{2,1} \cdots x_{2,(k-s)}$, where

$$\{x_{1,1}, \ldots, x_{1,s}\} \subseteq \{x_1, \ldots, x_{\lceil n/2 \rceil}\},$$
$$\{x_{2,1}, \ldots, x_{2,(k-s)}\} \subseteq \{x_{\lceil n/2 \rceil+1}, \ldots, x_n\}.$$

If $0 \leqslant s \leqslant p_l - 2$, then

$$x_{2,1} \cdots x_{2,(k-s)} \wedge T_{X_2}^{k+2-p_l} = x_{2,1} \cdots x_{2,(k-s)}.$$

Thus,

$$t \wedge \left(T_{X_1}^{p_l+q_l+1} \vee T_{X_2}^{k+2-p_l}\right) = t \wedge T_{X_1}^{p_l+q_l+1} \vee t = t.$$

If $p_l + q_l + 1 \leqslant s \leqslant k$, since

$$x_{1,1} \cdots x_{1,s} \wedge T_{X_1}^{p_l+q_l+1} = x_{1,1} \cdots x_{1,s},$$

$$t \wedge (T_{X_1}^{p_l+q_l+1} \vee T_{X_2}^{k+2-p_l}) = t \vee t \wedge T_{X_2}^{k+2-p_l} = t.$$

The equation 1 has been proved.

The equation 2 holds because if $p_m < p_l$, then $p_m + q_m < p_l$ and

$$T_{X_2}^{k+1-(p_m+j)} \wedge T_{X_2}^{k+2-p_l} = T_{X_2}^{k+1-(p_m+j)},$$

and if $p_m > p_l$, then $p_m > p_l + q_l + 1$ and

$$T_{X_1}^{p_m+j} \wedge T_{X_1}^{p_l+q_l+1} = T_{X_1}^{p_m+j}.$$

For the equation 3, it is apparent that it holds.

From 1, 2 and 3,

$$\begin{aligned} & f^k(x) \wedge \left(T_{X_1}^{p_l+q_l+1} \vee T_{X_2}^{k+2-p_l}\right) \\ &= f(x) \vee \bigvee_{j=0}^{q_l} T_{X_1}^{p_l+q_l+1} \wedge T_{X_2}^{k+1-(p_l+j)} \vee \bigvee_{j=0}^{q_l} T_{X_1}^{p_l+j} \wedge T_{X_2}^{k+2-p_l} \\ &\quad \vee \bigvee_{m \neq l} \bigvee_{j=0}^{q_m} T_{X_1}^{p_m+j} \wedge T_{X_2}^{k+1-(p_m+j)} \\ &= f(x) \vee \bigvee_{m \neq l} \bigvee_{j=0}^{q_m} T_{X_1}^{p_m+j} \wedge T_{X_2}^{k+1-(p_m+j)} \end{aligned}$$

because, for every j such that $0 \leqslant j \leqslant q_l$,

$$T_{X_1}^{p_l+q_l+1} \wedge T_{X_2}^{k+1-(p_l+j)} \leqslant T_{X_1}^{p_l+q_l+1} \wedge T_{X_2}^{k+1-(p_l+q_l+1)} \leqslant f(x),$$

$$T_{X_1}^{p_l+j} \wedge T_{X_2}^{k+2-p_l} \leqslant T_{X_1}^{p_l-1} \wedge T_{X_2}^{k+1-(p_l-1)} \leqslant f(x).$$

Hence,

$$
\begin{aligned}
& f^k(x) \wedge \bigwedge_{l=1}^{c} \left(T_{X_1}^{p_l+q_l+1} \vee T_{X_2}^{k+2-p_l} \right) \\
&= \bigwedge_{l=1}^{c} f^k(x) \wedge \left(T_{X_1}^{p_l+q_l+1} \vee T_{X_2}^{k+2-p_l} \right) \\
&= \bigwedge_{l=1}^{c} \left(f(x) \vee \bigvee_{j \neq l} \bigvee_{i=0}^{q_j} T_{X_1}^{p_j+i} \wedge T_{X_2}^{k+1-(p_j+i)} \right) \\
&= f(x) \vee \bigwedge_{l=1}^{c} \left(\bigvee_{j \neq l} \bigvee_{i=0}^{q_j} T_{X_1}^{p_j+i} \wedge T_{X_2}^{k+1-(p_j+i)} \right) \\
&= f(x) \vee \\
&\quad \bigvee_{1 \leqslant u < v \leqslant c} \left(\bigvee_{i=0}^{q_u} T_{X_1}^{p_u+i} \wedge T_{X_2}^{k+1-(p_u+i)} \right) \left(\bigvee_{j=0}^{q_v} T_{X_1}^{p_v+j} \wedge T_{X_2}^{k+1-(p_v+j)} \right) \\
&= f(x) \vee \bigvee_{1 \leqslant u < v \leqslant c} T_{X_1}^{p_v} \wedge T_{X_2}^{k+1-(p_u+q_u)}.
\end{aligned}
$$

Since

$$
T_{X_1}^{p_v} \wedge T_{X_2}^{k+1-(p_u+q_u)} \leqslant T_{X_1}^{p_v-1} \wedge T_{X_2}^{k+1-(p_v-1)} \leqslant f(x),
$$

$$
f^k(x) \wedge \bigwedge_{l=1}^{c} \left(T_{X_1}^{p_l+q_l+1} \vee T_{X_2}^{k+2-p_l} \right) = f(x).
$$

Thus,

$$
C_m(f) \leqslant C_m(f^k) + C_m \left(\bigwedge_{l=1}^{c} \left(T_{X_1}^{p_l+q_l+1} \vee T_{X_2}^{k+2-p_l} \right) \right) + 1.
$$

Because $C_m(T_{X_1}^{p_l+q_l+1}) = O(n \log n)$, $C_m(T_{X_2}^{k+2-p_l}) = O(n \log n)$ and c is a constant,

$$
C_m(f) \leqslant C_m(f^k) + O(n \log n).
$$

The subsequent two corollaries follow from Theorem 4.3. Corollary 4.1 shows that, for every $f \in \mathrm{R}_n^{k,c}$ whose monotone circuit size complexity is $\omega(n \log n)$, it is different from the circuit size complexity of f^k at most by a multiplicative constant. If $\omega(n \log n)$ lower bound on the monotone circuit size complexity of f (f^k) is proved, it implies the same lower bound on that of f^k (f).

Corollary 4.1 For every $f \in \mathrm{R}_n^{k,c}$, if $C_m(f^k) = \omega(n \log n)$ or $C_m(f) = \omega(n \log n)$, then $C_m(f) = \Theta(C_m(f^k))$.

(Proof) This corollary is immediate from Theorem 4.3 and the fact that $C_m(f^k) \leqslant C_m(f) + O(n \log n)$. □

Corollary 4.2 For every $f \in \mathrm{R}_n^{k,c}$, if $C_m(f) = \omega(n(\log n)^2)$, then $C(f) = \Theta(C_m(f))$.

(Proof) For every $f \in \mathrm{R}_n^{k,c}$, from Theorem 4.3, $C_m(f) \leqslant C_m(f^k) + O(n \log n)$. Since $C_m(f^k) \leqslant O(C(f^k)) + O(n(\log n)^2)$ and $C(f^k) \leqslant C(f) + O(n)$,

$$C(f) \leqslant C_m(f) \leqslant O(C(f)) + O(n(\log n)^2).$$

Hence, if $C_m(f) = \omega(n(\log n)^2)$, then $C(f) = \Theta(C_m(f))$. □

The above corollary claims that, for every Boolean function in $\mathrm{R}_n^{k,c}$ whose monotone circuit size complexity is $\omega(n(\log n)^2)$, it is larger than the circuit size complexity of f at most by a constant factor. If $C_m(f)$ is proved to be $\omega(n(\log n)^2)$, then we obtain the same lower bound on $C(f)$.

4.6 Conclusion

The circuit size complexity of slice Boolean functions and homogeneous Boolean functions has been discussed.

In Section 4.4, the circuit size complexity of homogeneous Boolean functions and their slices has been discussed. It is known that there exist k-homogeneous Boolean functions whose k-th slices are much harder to compute than their other slices. We have improved the lower bound

on the complexity of the k-th slices of such homogeneous Boolean functions. The improved lower bound is optimal for constant k.

In Section 4.5, homogeneous Boolean functions whose circuit size complexity and monotone circuit size complexity is almost equal have been presented. For any of these homogeneous Boolean functions with n variables, if its monotone circuit size complexity is $\omega(n(\log n)^2)$, then it differs from the circuit size complexity of the Boolean function at most by a multiplicative constant.

For slice Boolean functions with n variables and homogeneous Boolean functions with n variables presented in Section 4.5, a lower bound of $\omega(n(\log n)^2)$ on their monotone circuit size complexity implies the same lower bound on their circuit size complexity. This approach gives us chances to prove a nonlinear lower bound on the circuit size complexity of some explicitly defined Boolean functions.

Appendix

Supplement of the Proof of Lemma 4.1

(1) When n is even,

$$\sum_{i=1}^{k} {}_{n/2}\mathrm{C}_i \, {}_{n/2}\mathrm{C}_{k-i} = {}_n\mathrm{C}_k$$

and

$$\sum_{i=1}^{k} (-1)^i \, {}_{n/2}\mathrm{C}_i \, {}_{n/2}\mathrm{C}_{k-i} = \begin{cases} (-1)^{k/2} \, {}_{n/2}\mathrm{C}_{k/2} & \text{if } k \text{ is even} \\ 0 & \text{if } k \text{ is odd.} \end{cases}$$

If k is even,

$$\begin{aligned} |\mathrm{P}| &= \sum_{j=0}^{k/2} {}_{n/2}\mathrm{C}_{2j} \, {}_{n/2}\mathrm{C}_{k-2j} \\ &= \frac{1}{2}\left(\sum_{i=0}^{k} {}_{n/2}\mathrm{C}_i \, {}_{n/2}\mathrm{C}_{k-i} + \sum_{i=0}^{k} (-1)^i \, {}_{n/2}\mathrm{C}_i \, {}_{n/2}\mathrm{C}_{k-i} \right) \\ &= \frac{1}{2}\left({}_n\mathrm{C}_2 + (-1)^{k/2} \, {}_{n/2}\mathrm{C}_{k/2} \right). \end{aligned}$$

If k is odd,

$$\begin{aligned}
|\mathrm{P}| &= \sum_{j=0}^{\lfloor k/2 \rfloor} {}_{\lfloor n/2 \rfloor}\mathrm{C}_{2j}\ {}_{\lceil n/2 \rceil}\mathrm{C}_{k-2j} + {}_{n/2}\mathrm{C}_k \\
&= \frac{1}{2}\left(\sum_{i=0}^{k} {}_{n/2}\mathrm{C}_i\ {}_{n/2}\mathrm{C}_{k-i} + \sum_{i=0}^{k}(-1)^i\ {}_{n/2}\mathrm{C}_i\ {}_{n/2}\mathrm{C}_{k-i}\right) + {}_{n/2}\mathrm{C}_k \\
&= \frac{1}{2}{}_n\mathrm{C}_2 + {}_{n/2}\mathrm{C}_k.
\end{aligned}$$

(2) When n is odd,

$$\sum_{i=0}^{k} {}_{\lfloor n/2 \rfloor}\mathrm{C}_i\ {}_{\lceil n/2 \rceil}\mathrm{C}_{k-i} = {}_n\mathrm{C}_k$$

and

$$\sum_{i=1}^{k}(-1)^i\ {}_{\lfloor n/2 \rfloor}\mathrm{C}_i\ {}_{\lceil n/2 \rceil}\mathrm{C}_{k-i} = \begin{cases} (-1)^{k/2}\ {}_{\lfloor n/2 \rfloor}\mathrm{C}_{k/2} & \text{if } k \text{ is even} \\ (-1)^{\lfloor k/2 \rfloor}\ {}_{\lfloor n/2 \rfloor}\mathrm{C}_{\lfloor k/2 \rfloor} & \text{if } k \text{ is odd.} \end{cases}$$

If k is even,

$$\begin{aligned}
|\mathrm{P}| &= \sum_{j=0}^{k/2} {}_{\lfloor n/2 \rfloor}\mathrm{C}_{2j}\ {}_{\lceil n/2 \rceil}\mathrm{C}_{k-2j} \\
&= \frac{1}{2}\left(\sum_{i=0}^{k} {}_{n/2}\mathrm{C}_i\ {}_{n/2}\mathrm{C}_{k-i} + \sum_{i=0}^{k}(-1)^i\ {}_{n/2}\mathrm{C}_i\ {}_{n/2}\mathrm{C}_{k-i}\right) \\
&= \frac{1}{2}\left({}_n\mathrm{C}_2 + (-1)^{k/2}\ {}_{n/2}\mathrm{C}_{k/2}\right).
\end{aligned}$$

If k is odd,

$$\begin{aligned}
|\mathrm{P}| &= \sum_{j=0}^{\lfloor k/2 \rfloor} {}_{\lfloor n/2 \rfloor}\mathrm{C}_{2j}\ {}_{\lceil n/2 \rceil}\mathrm{C}_{k-2j} + {}_{\lfloor n/2 \rfloor}\mathrm{C}_k \\
&= \frac{1}{2}\left(\sum_{i=0}^{k} {}_{\lfloor n/2 \rfloor}\mathrm{C}_i\ {}_{\lceil n/2 \rceil}\mathrm{C}_{k-i} + \sum_{i=0}^{k}(-1)^i\ {}_{\lfloor n/2 \rfloor}\mathrm{C}_i\ {}_{\lceil n/2 \rceil}\mathrm{C}_{k-i}\right) + \\
&\quad {}_{\lfloor n/2 \rfloor}\mathrm{C}_k \\
&= \frac{1}{2}\left({}_n\mathrm{C}_2 + (-1)^{\lfloor k/2 \rfloor}\ {}_{\lfloor n/2 \rfloor}\mathrm{C}_{\lfloor k/2 \rfloor}\right) + {}_{\lfloor n/2 \rfloor}\mathrm{C}_k.
\end{aligned}$$

Chapter 5

Conclusion

In this thesis, Boolean functions related to cryptography, that is, nonlinear Boolean functions, homogeneous Boolean functions and slice Boolean functions have been discussed.

In Chapter 2, properties of nonlinear Boolean functions and relationships among nonlinearity criteria were discussed. The PC, the SAC and the nonlinearity were mainly investigated. Exact characterizations of Boolean functions satisfying the PC of degree $n-1$ and $n-2$ were presented. In particular, Boolean functions with n variables satisfying the PC of degree $n-2$ are perfectly nonlinear for even $n \geqslant 4$. A necessary and sufficient condition was presented for a Boolean function to satisfy the PC with respect to all but linearly independent elements. The methods of construction were shown for Boolean functions satisfying the PC with respect to all but one or three elements in $\{0,1\}^n - \{(0,\ldots,0)\}$. The methods can generate all such functions from perfectly nonlinear Boolean functions. Relationships between the degree of the PC and the order of the SAC were also discussed. These results are expected to be a foundation of the design of private key cryptosystems.

It is an open question whether there exist balanced Boolean functions satisfying the PC of degree $n-3$ for even n. Future work is to investigate properties of Boolean functions satisfying other nonlinearity criteria.

In Chapter 3, complexity of Boolean functions satisfying the PC was discussed. First, it was shown that every Boolean function satisfy-

ing the PC of degree 1 is unate in at most two of its variables and that every Boolean function satisfying the PC of degree 2 is not unate in any one of its variables. Second, the optimal lower bound of $\lfloor \log n \rfloor - 1$ was obtained for the inversion complexity of perfectly nonlinear Boolean functions constructed by the method of Maiorana. Third, the nearly optimal lower bound of $n^2/4 - 1$ was presented for the formula size of every Boolean function which satisfies the PC of degree 1. Fourth, a lower bound of $\Omega(n^2)$ was obtained for the AT^2 VLSI complexity of perfectly nonlinear Boolean functions with multiple outputs. Finally, an exponential lower bound was obtained for the OBDD size of perfectly nonlinear Boolean functions with multiple outputs. Some of the above results were obtained with the use of novel techniques previously proposed.

Open questions are the lower bounds on the inversion complexity, the AT^2 VLSI complexity and the OBDD size of every perfectly nonlinear Boolean function.

In Chapter 4, the maximal complexity gap was obtained for the monotone circuit size complexity of the slices of homogeneous Boolean functions. There exist a k-homogeneous Boolean function with the property that the monotone circuit size complexity of its k-th slice is $\Omega({}_nC_k/\log {}_nC_k)$ and that of its $u(> k)$-th slice is $O(n \log n)$. A set of homogeneous Boolean functions with circuit size complexity and monotone circuit size complexity almost equal were presented. For every Boolean function in this set, a lower bound of $\omega(n(\log n)^2)$ on the monotone circuit size complexity implies the same lower bound on the circuit size complexity.

It is left as a future work to get a super-linear lower bound on the circuit complexity of some explicitly defined Boolean function.

Bibliography

[AT90] Adams, C. M. and Tavares, S. E., "Generating and counting binary bent sequences," IEEE Trans. Info. Theo., vol. IT-36, no. 5, 1990, pp. 1170–1173.

[AKS83] Ajtai, M., Komlos, J. and Szemeredi, E., "An $O(n \log n)$ sorting network," Proc. 15th ACM STOC, 1983, pp. 1–9.

[AB86] Alon, N. and Boppana, R., "The monotone circuit complexity of Boolean functions," Combinatorica, vol. 7, 1986, pp. 1–22.

[And85] Andreev, A. E., "A method of proving lower bounds on the complexity of monotone Boolean functions," Dokl. Akad. Nauk, 282, 1985, pp. 1033–1037.

[BL90] Benaloh, J. and Leichter, J., "Generalized secret sharing and monotone functions," Proc. CRYPTO'88, LNCS no. 403, 1990, pp. 27–35.

[Ber82] Berkowitz, S., "On some relationships between monotone and non-monotone circuit complexity," Tech. Rep. Univ. of Tronto, 1982.

[BS93] Biham, E. and Shamir, A., Differential cryptanalysis of the Data Encryption Standard, Springer-Verlag, 1993.

[Bla79] Blakley, G. R., "Safeguarding cryptographic keys," Proc. AFIPS 1979 Natl. Comp. Conf., vol. 48, 1979, pp. 313–317.

[Blu84] Blum, N., "A Boolean function requiring 3n network size," Theoretical Computer Science, vol. 28, 1984, pp. 337–345.

[Bry86] Bryant, R., "Graph-based algorithms for Boolean function manipulation," IEEE Trans. Comput., vol. C-35, no. 8, 1986, pp. 677–691.

[Dun86] Dunne, P. E., "The complexity of central slice functions," Theoretical Computer Science, vol. 44, 1986, pp. 247–257.

[Dun89] Dunne, P. E., "On monotone simulations of nonmonotone networks," Theoretical Computer Science, vol. 66, 1989, pp. 15–25.

[For90] Forré, R., "The strict avalanche criterion: Spectral properties of Boolean functions and an extended definition," Proc. CRYPTO'88, LNCS no. 403, 1990, pp. 450–468.

[HKP84] Hoover, H. J., Klawe, M. M. and Pippenger, N. J., "Bounding fan-out in logical networks," JACM, vol. 31, 1984, pp. 13–18.

[Koh78] Kohavi, Z., Switching and finite automata theory, 2nd edition, Tata McGraw-Hill, 1978.

[Kra71] Krapchenko, V., "A method of determining lower bounds for the complexity of π schemes," Math. Notes Acad. Sci. USSR, vol. 11, 1971, pp. 474–479.

[Llo91] Lloyd, S., "Properties of binary functions," Proc. EUROCRYPT'90, LNCS no. 473, 1991, pp. 124–139.

[Mar58] Markov, A. A., "On the inversion complexity of a system of functions," JACM, vol. 5, no. 4, 1958, pp. 331–334.

[Mat94] Matsui, M., "Linear cryptanalysis of DES cipher," Proc. 1994 Sympo. Cryptography and Information Security, 1994 (In-Japanese).

[MS90] Meier, W. and Staffelbach, O., "Nonlinearity criteria for cryptographic functions," Proc. EUROCRYPT'89, LNCS no. 434, 1990, pp. 549–562.

[Nyb91] Nyberg, K., "Perfect nonlinear S-boxes," Proc. EUROCRYPT'91, LNCS no. 547, 1991, pp. 378–386.

[PLLGV91] Preneel, B., Leekwijk, W. V., Linden, L. V., Govaerts, R. and Vandewalle, J., "Propagation characteristics of Boolean functions," Proc. EUROCRYPT'90, LNCS no. 473, 1991, pp. 161–173.

[PGV92] Preneel, B., Govaerts, R. and Vandewalle, J., "Boolean functions satisfying higher order propagation criteria," Proc. EUROCRYPT'91, LNCS no. 547, 1992, pp. 141–152.

[Raz85] Razborov, A. A., "A lower bound on the monotone complexity of the logical permanent," Mat. Zametki, vol. 37, 1985, pp. 887–901.

[Rot76] Rothaus, O. S., "On 'bent' functions," J. Combinatorial Theo. (A), vol. 20, 1976, pp. 300–305.

[Rue91] Rueppel, R. A., "Stream ciphers," in Contemporary cryptology: The science of information integrity, G. Simmons, ed., IEEE Press, 1991, pp. 65–134.

[Sav76] Savage, J. E., The complexity of computing, John Wiley, 1976.

[SZZ93] Seberry, J., Zhang, X. M. and Zheng, Y., "Highly nonlinear balanced Boolean functions satisfying high degree propagation criterion," Tech. Rep. The Univ. Wollongong, tr-93-1, 1993.

[Sha79] Shamir, A., "How to share a secret," Commun. ACM, vol. 22, no. 11, 1979, pp. 612–613.

[Sha49] Shannon, C. E., "The synthesis of two-terminal switching circuits," Bell System Tech. Journal, vol. 28, 1949, pp. 59–98.

[Ull84] Ullman, J. D., Computational aspects of VLSI, Computer Science Press, 1984.

[Val86] Valiant, L. G., "Negation is powerless for Boolean slice functions," SIAM Journal on Computing, vol. 15, 1986, pp. 531–535.

[WT86] Webster, A. F. and Tavares, S. E., "On the design of S-boxes," Proc. CRYPTO'85, LNCS no. 218, 1986, pp. 523–534.

[Weg85] Wegener, I., "On the complexity of slice functions," Theoretical Computer Science, vol. 38, 1985, pp. 55–68.

[Weg86] Wegener, I., "More on the complexity of slice functions," Theoretical Computer Science, vol. 43, 1986, pp. 201–211.

[Weg87] Wegener, I., The complexity of Boolean functions, John Wiley & Sons, 1987.

Acknowledgements

I would like to express sincere gratitude to Professor IKEDA Katsuo of Kyoto University. He gave me the opportunity of this research, and has been giving me continuous guidance, interesting suggestions, intensive critisism and encouragements during this research.

I would also like to appreciate Professor YAJIMA Shuzo of Kyoto University who introduced me the research field of computartional complexity theory and has been giving me invaluable suggestions and encouragements.

I am indebt to Associate Professor MINOH Michihiko of Kyoto University for his helpful suggestions, comments and valuable insights.

I also acknowledge the interesting comments that I have received from Professor HIRAISHI Hiromi of Kyoto Sangyo University, Associate Professor TAKAGI Naofumi of Nagoya University, Lecturer ISHIURA Nagisa of Osaka University and Dr. MINATO Shin-ichi of NTT LSI Laboratory.

Thanks are also due to all members of Professor Ikeda's Laboratory for their discussions and supports throughout this research.

List of Publications by The Author

Major Publications

1. Hirose, S. and Yajima, S., "On the circuit complexity of slice functions and homogeneous functions," IEICE Trans. D-I, vol. J75-D-I, no. 6, 1992, pp. 331–338 (in Japanese).

2. Hirose, S. and Ikeda, K., "Propagation characteristics of Boolean functions and their balancedness," IEICE Trans. on Fundamentals, vol. E78-A, no. 1, 1995, pp. 11–18.

3. Hirose, S. and Ikeda, K., "Relationships among nonlinearity criteria of Boolean functions," To appear in IEICE Trans. on Fundamentals, vol. E78-A, no. 2, 1995.

4. Hirose, S. and Ikeda, K., "Complexity of Boolean functions satisfying the propagation criterion," To appear in IEICE Trans. on Fundamentals, vol. E78-A, no. 4, 1995.

Technical Reports

1. Hirose, S. and Yajima, S., "On the circuit complexity of slice functions and homogeneous functions," Technical Report of IEICE, COMP89-117, 1990, pp. 33–38 (in Japanese).

2. Hirose, S. and Ikeda, K., "On a group digital signature scheme," The Proc. of the 15th Symposium on Information Theory and Its

Applications, W42-3, 1992, pp. 203–206 (in Japanese).

3. Hirose, S. and Ikeda, K., "Relationships among nonlinearity criteria of Boolean functions," Technical Report of IEICE, COMP93-66, 1993, pp. 37–46.

4. Hirose, S. and Ikeda, K., "A note on the propagation characteristics and the strict avalanche criteria," The Proc. of the 1994 Symposium on Cryptography and Information Security, SCIS94-8B, 1994.

5. Hirose, S. and Ikeda, K., "Complexity of Boolean functions satisfying the propagation criterion," The Proc. of the 1995 Symposium on Cryptography and Information Security, SCIS95-B3.3, 1995.

Convention Records

1. Hirose, S. and Yajima, S., "The circuit complexity of homogeneous functions and their slices," IEICE National Convention Record of Information and System, 6-4, 1990 (in Japanese).

2. Kato, T., Hirose, S., Minoh, M. and Ikeda K., "A protocol for group oriented signature scheme applying ElGamal's public key cryptosystem," IEICE National Convention Record of Fundamentals, A-184, 1992 (in Japanese).

3. Hirose, S. and Ikeda, K., "Computational complexity of perfectly nonlinear Boolean functions," IEICE National Convention Record of Information and System, D-5, 1994.

4. Hirose, S. and Ikeda, K., "Unateness, symmetry and self-duality of Boolean functions satisfying the propagation criterion," IEICE National Convention Record of Information and System, D, 1995.