ulm university universität
uulm

On Canalizing Boolean Functions

DISSERTATION

zur Erlangung des akademischen Grades eines

DOKTOR-INGENIEURS
(DR.-ING.)

der Fakultät für Ingenieurwissenschaften
und Informatik der Universität Ulm

von

Johannes Georg Klotz

aus Ichenhausen

Gutachter:	Prof. Dr.-Ing. Martin Bossert
	Prof. Dr. Ivo Große
Amtierende Dekanin:	Prof. Dr. Tina Seufert

Ulm, 11. Juni 2014

Meinen Großeltern

Preface

This dissertation presents recent results of my research activities at the Institute for Communications Engineering at Ulm University. My work has been supported by the German research council "Deutsche Forschungsgemeinschaft" (DFG) under Grants Bo 867/25-1 and Bo 867/25-2. Parts of the results have been published in the following articles in conference proceedings and scientific journals:

[1] Johannes Georg Klotz, Steffen Schober, and Martin Bossert. On the predecessor problem in Boolean network models of regulatory networks. In *Proc. of the 3rd International Conference on Bioinformatics and Computational Biology (BICoB)*, pages 67–73, New Orleans, Louisiana, USA, March 2011.

[2] Johannes Georg Klotz, Ronny Feuer, Katrin Gottlieb, Oliver Sawodny, Georg Sprenger, Martin Bossert, Michael Ederer, and Steffen Schober. Properties of a Boolean network model of *Escherichia coli*. In *Proc. of the 8th International Workshop on Computational Systems Biology (WCSB)*, pages 108–111, Zuerich, Switzerland, June 2011.

[3] Johannes Georg Klotz, Steffen Schober, and Martin Bossert. On the predecessor problem in Boolean network models of regulatory networks. *International Journal of Computers and Their Applications*, 19(2):93–100, June 2012.

[4] Johannes Georg Klotz, Ronny Feuer, Oliver Sawodny, Martin Bossert, Michael Ederer, and Steffen Schober. Properties of Boolean networks and methods for their tests. *EURASIP Journal on Bioinformatics and Systems Biology*, 2013(1):1, 2013.

[5] Johannes Georg Klotz, David Kracht, Martin Bossert, and Steffen Schober. Canalizing Boolean functions maximize the mutual information. In *9th International ITG Conference on Systems, Communications and Coding (SCC)*, Munich, Germany,, January 2013.

[6] Johannes Georg Klotz, Reinhard Heckel, and Steffen Schober. Bounds on the average sensitivity of nested canalizing functions. *PLoS ONE*, 8 (5), 2013.

[7] Johannes Georg Klotz, Martin Bossert, and Steffen Schober. Computing preimages of boolean networks. *BMC Bioinformatics*, 14(Suppl 10):S4, August 2013.

[8] Johannes Georg Klotz, Martin Bossert, and Steffen Schober. On the noise sensitivity and mutual information of (nested-) canalizing Boolean functions. In *IEEE Information Theory Workshop, ITW 2013*, 2013.

[9] Johannes Georg Klotz, David Kracht, Martin Bossert, and Steffen. Schober. Canalizing Boolean functions maximize mutual information. *Information Theory, IEEE Transactions on*, 60(4):2139–2147, April 2014.

Acknowledgments

This thesis is the result of my years as a research assistant at the Institute of Communications Engineering at Ulm University. Many people have contributed to its development and I would like to express my gratitude to them.

First of all, I would like to thank my supervisor, Prof. Dr.-Ing. Martin Bossert, very much for giving me the opportunity to work in his research group. His constant efforts to provide a perfect working environment, not only scientifically but also socially, made my years at his institute very fruitful and enjoyable.

Secondly, I am grateful to Prof. Dr. Ivo Große from the Martin-Luther-University Halle-Wittenberg for acting as my co-supervisor and showing so much interest in my research. Furthermore, I would like to thank Prof. Ph. D. Carl E. Krill and Prof. Dr.-Ing. Jian Xie for being part of my commission.

My research would have not been possible without my "bio" research group, consisting of Katharina Mir, David Kracht and our head, Steffen Schober. Thank you for your support, inspiration, motivation and discussions! Furthermore, I would like to thank the members of my former research group, Carolin Huppert and Frederic Knabe, and all current and former colleagues at TAIT and NT, in particular Ulrike Stier and Ilse Walter, the "heart and soul" of our institute. Further, I thank Patrizia Di Campli San Vito, Katharina Mir and Steffen Schober for proofreading the manuscript.

Finally, I would like to thank my family and friends for all their love and support.

Johannes Georg Klotz
Ulm, June 2014

Abstract

Boolean networks are an important model of gene regulatory networks in systems and computational biology. Such networks have been widely studied with respect to their stability and error tolerance. It has turned out that canalizing Boolean functions and their subclass, the nested canalizing functions, appear frequently in such networks. These classes have been shown to have a stabilizing effect on the network dynamics. One measure for the stability is the average sensitivity of Boolean functions. Using Fourier analysis, we provide upper bounds on the average sensitivity for canalizing and nested canalizing functions. The latter bound proves an open conjecture in the literature. Further, we state upper and lower bounds on the noise sensitivity, based on an inductive proof of the spectra of the functions. The noise sensitivity gives the error tolerance of functions with noisy inputs. We show that canalizing functions maximize the mutual information between an input variable and the outcome of the function. We provide relationships between the noise sensitivity and the mutual information of functions with noisy inputs. Using these relationships we prove upper and lower bounds on the mutual information for canalizing and nested canalizing functions. To prove our results we derive spectral properties of canalizing and nested canalizing functions, which can also be used to test membership of functions to these classes.

Finally, we present two algorithms solving the preimage problem of Boolean networks. The first approach depends on the properties of canalizing functions, the second algorithm is based on the well-known sum-product algorithm (belief propagation).

Contents

Chapter 1

Introduction

BOOLEAN MODELING plays an important role in systems and computational biology. It turned out that canalizing and nested canalizing functions often appear in Boolean networks [Kau74]. In this thesis we will investigate these functions with respect to error tolerance, namely noise sensitivity and average sensitivity, and mutual information, a measure for the information processing capability. Further, we will address the preimage problem of Boolean networks.

In this chapter let us first briefly give the biological background and motivation. After that the topics of this thesis are introduced, and finally the will state the outline of this work.

1.1. Biological Background

The smallest structural and functional entity of a living organism is the cell [Mun00]. All actions and tasks of cells are performed by chemical reactions. These reactions are often consecutive and form so-called reaction chains or cycles [KLW+09]. One famous example for such a cycle is the *citric acid cycle*, which occurs in all organisms [AJL+04]. It metabolizes nutrients into biochemical energy, mainly by oxidation [AJL+04]. If in such a chain or cycle one reaction has a higher throughput than the other, the cell would have a lack of some intermediate products and a overrun of others. Hence, all reactions need to be somehow *synchronized* [KLW+09].

The process of such reactions is described in systems biology by so-called *flux models*. A flux is a normalized measure describing the rate with which a certain reaction takes place, assuming that the intermediate products are in a steady state [KLW+09]. The so-called flux balance analysis is used to predict the behavior of a cell using this model. More specific, one can perform *in silico* experiments to analyze how cells behave in certain environments and under a certain nutrition. Hereby one assumes that the cell works in the most efficient way [KLW+09, FET+07].

But those reactions do not happen uncontrolled in actual cells. To take place, most reactions would need a much higher temperature than is usually the case. Therefore, cells use enzymes, which are mostly proteins, as catalysts, where each enzyme controls only one reaction [AJL+04]. Proteins are encoded as genes on the *Deoxyribonucleic Acid* (DNA). The production of proteins is called gene expression and consists mainly of two parts (see Figure 1.1). The first part is the transcription, which is the reading and copying of the DNA and results in the *mRNA* (messenger ribonucleic acid), for which a special molecule, the *polymerase*, docks and slides along the DNA performing the copying process. Then, during

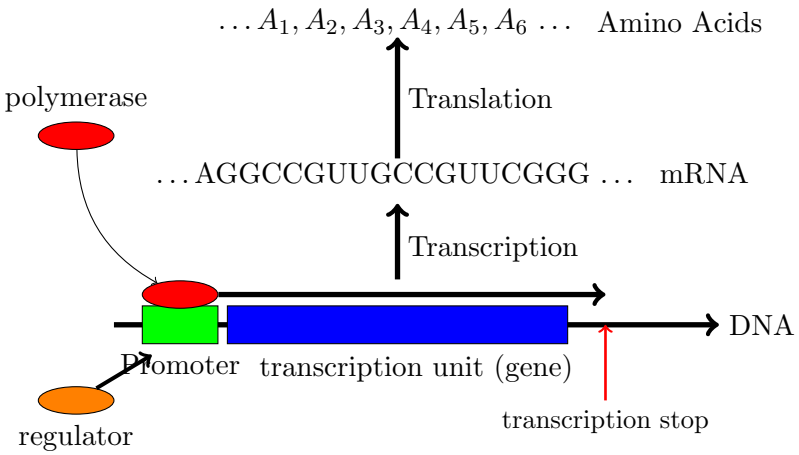


Figure 1.1.: [Sch13] Gene Expression

the translation, the mRNA is used to produce the protein out of amino acids [AJL+04]. Not all proteins are acting as enzymes, they also perform a variety of tasks in a cell, e.g., they serve as structure and motion elements [AJL+04].

The amount of a certain protein present in a cell directly controls the rate of the corresponding chemical reaction. Hence, there is a need to control the rate with which a gene is expressed. This is done through regulation. Therefore, other proteins can bind within the so-called *promoter region* on the DNA, which is mostly located directly in front of the gene [AJL+04]. Those bound proteins either act as *repressors*, i.e., they reduce the probability of the polymerase to attach to the DNA at this point and, hence, reduce the expression rate of the gene, or they act as *activators* magnifying the docking probability of the polymerase and, hence, leading to an increase in the expression rate [JK08]. Let us for example examine regulation of the gene coding for protein *β-galactosidase*, an enzyme controlling the decomposition of *lactose* in the well-studied gut bacteria *Escherichia coli* (*E. coli*). Another protein called *LacI* binds at the promoter region of β-galactosidase and therefore prohibits this gene to be transcribed. However, if lactose is present in the cell, the LacI can not bind anymore, hence, β-galactosidase can be produced, and finally the lactose will be consumed [JK08]. This example also shows, how, by regulation, a cell can adapt to different environmental conditions.

As stated by Stuart Kauffman [Kau93] many cellular and chemical processes react with an S-shaped response to a change in the molecular input. Hence, small changes to the input do not alter the reaction a lot, however, if the input exceeds a certain threshold, a drastic change in the output occurs. Thus, one can consider genes to be either *on* or *off* and use Boolean, i.e., binary, values to model the state (see Figure 1.2).

1.2. Boolean Networks

If more than one protein regulates the expression (*on*/*off*) of a gene, the overall effect on the expression of a particular gene can be modeled using a *Boolean function* [Kau93]. Consequently, all these genes, functions, and interactions form a *Boolean network*, modeling the regulative relations within a cell [Kau93].

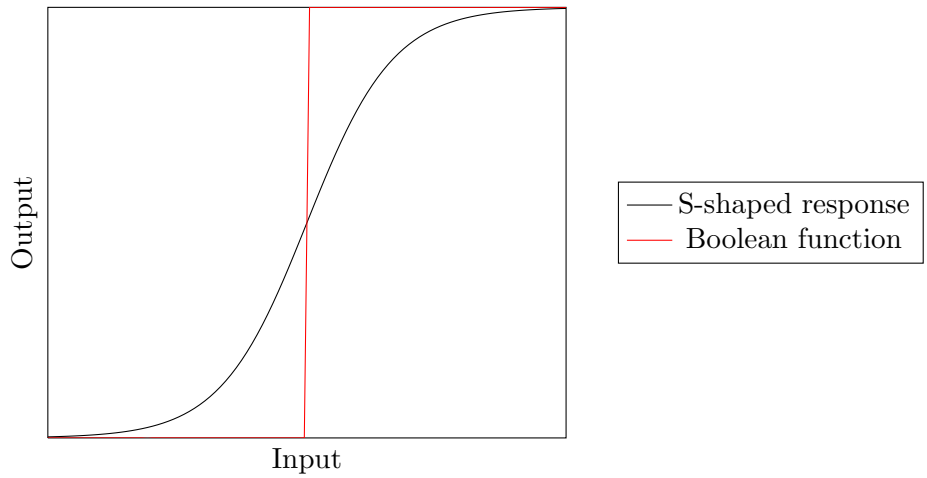


Figure 1.2.: [Kau93] S-shaped Response and a Boolean function approximating it.

These networks, including random Boolean Networks, have been widely studied with respect to their *stability* [Kau69b, Kau74]. A network is considered stable, if the effect on the network caused by the perturbation of a node asymptotically vanishes after a certain number of iterations. Kauffman investigated these networks numerically. Later, Lynch [Lyn07] showed that stable networks can exist. Such stable networks are also called *ordered* or *non-chaotic*. Further, it has been shown that a network only consisting of *nested canalizing functions* is always stable [KPST04]. *Canalizing Boolean functions* and their subclass nested canalizing functions have been classified by Kauffman [Kau93, KPST03] and he pointed out that many important regulatory Boolean functions are in fact canalizing. A function is canalizing, if one state of a certain input determines one output state alone.

In this thesis, we consider feed-forward networks, i.e., networks without feedback loops, for example the regulatory network of *E. coli* as introduced by Covert et al. [CKR+04]. However, instead of investigating the network itself, we are focusing on the properties of the Boolean functions attached to the nodes.

1.3. Topics of this Thesis

Many results indicate that it is of importance for regulative Boolean functions to have a high error tolerance [SK04], i.e., are somehow more resistant to small changes on the inputs, which may be caused by a mutation. Further, it is well known, that a low **average sensitivity**, a measure for the error tolerance indicating the impact of inverting an input, is a prerequisite for non-chaotic behavior of random Boolean networks [SB07]. A generalization of the average sensitivity is the **noise sensitivity**, measuring the impact of noise added to the function's inputs.

It turns out that Boolean functions in such regulatory networks do not only need to be robust against errors, they also need to pass on information. Therefore, we will also investigate the **mutual information**, a measure to determine the information processing capabilities, which is closely related to the noise sensitivity.

We will also address the so-called ***preimage* problem** of Boolean network. If we are

observing feed-forward networks, i.e., networks without feedback loops, these networks provide a direct mapping of the input states (environmental conditions) to the output states (gene activities). The preimage problem inverts this mapping, i.e., finds the set of environmental conditions leading to a certain gene activity.

1.4. Outline of this Thesis

This thesis is organized as follows. In Chapter 2 we will formally introduce Boolean functions, canalizing and nested canalizing functions. Further, we will briefly recall the Fourier analysis of Boolean functions, which is our main tool in this thesis. Next, we will discuss the concept of restricted Boolean functions and prove the impact of this restriction to the Fourier spectra. We call a function restricted, if one input is set to a certain value. Using the properties of restricted functions, we prove the conditions a function's spectra must fulfill in order to be canalizing.

Chapter 3 starts with the definition of important concepts, including the average sensitivity and the noise sensitivity. We will show the effect of restrictions on these sensitivities. Further, we will give bounds on the noise sensitivity for canalizing functions, nested canalizing functions and some specific nested canalizing functions, which indicate that the noise sensitivity of those classes is remarkably low. Further we prove that the average sensitivity of nested canalizing functions is always smaller than $4/3$ as conjectured in literature.

In Chapter 4 the information processing capabilities of Boolean functions, namely the mutual information, is addressed. We will first examine the mutual information between one or more inputs and the function's output and will show that it is maximized by canalizing functions. After that, we will investigate the mutual information between the noisy input and the output of a function. Here, the relationship between mutual information and noise sensitivity is derived. Based on this we will then give upper and lower bounds on the mutual information, which again indicate a high information throughput for nested canalizing functions. Further, we will show that a subclass of nested canalizing functions asymptotically reaches an upper bound conjectured in literature.

After addressing Boolean functions we will focus on Boolean networks in Chapter 5. In particular we will present two algorithms to solve the preimage problem. The first approach uses the fact that canalizing functions can be inverted under concrete circumstances. The second algorithm only delivers parts of the set of preimages by using an adaption of the well-known sum-product algorithm. We will evaluate our approaches for the regulatory network of *Escherichia coli* and some randomly generated networks.

Finally, the results of this theses are summarized in Chapter 6.

Chapter 2

Boolean Functions and their Spectral Properties

A mathematical description of logic was first given by George Boole in 1847, when he published his fundamental work *The Mathematical Analysis of Logic* [Boo47]. Other early important works[1] establishing Boolean algebra were published by Poretski (e.g., [Por84]), Schröder ([Sch90]), Löwenheim (e.g. [Lö08]) and Stone ([Sto35, Sto36]). In 1949 Claude Shannon used Boolean logic for analysis and synthesis of switching circuits [Sha49]. Since then Boolean algebra and subsequently Boolean functions have become a fundamental element of electrical engineering and computer science.

In a biological context the use of binary systems to describe gene expression has been suggested –among others– by Jacob and Monod [JM62] and Kauffman [Kau67]. The latter also proposed to use networks of Boolean functions to describe regulatory relations among genes [Kau69b]. We will address Boolean networks in Chapter 5.

Spectral methods have first been applied on Boolean functions by Golomb [Gol59]. His PhD student Titsworth was the first to propose a *Fourier-type* transform [Tit62]. Their investigations, however, did not take the distribution of the input variables into account. This was done much later by the authors of [FJS91], where the Fourier transform was extended by using different basis functions, which takes the probability distributions of the input variables into account and was proposed by Bahadur in 1961 [Bah61]. In this work we will use the Fourier transform as our main tool.

Before Kauffman introduced canalizing functions (CF), sometimes also called *forcible* or *forcing* functions, in the context of regulatory Boolean networks [Kau69b, Kau74], it has long been known [Wad42] that canalizing effects play an important role in cellular biology. It was conjectured that these functions have a stabilizing effect on the network dynamics [Kau74, KPST04] and were therefore widely investigated [HV86, Sta87]. It has further been shown that many regulatory network models contain mainly CFs, e.g., yeast [HSWK02, LLL$^+$04, DB08].

CFs are also significant in the field of signal processing when designing nonlinear filters, namely *stack filters* [WCG86, YC90, GYC92]. For example, it turned out that if these filters are defined on CFs, they have optimal convergence behavior [GYC92]. Finally, the number of CFs and the probability of a random BF to be canalizing have been investigated in [JSK04].

In [SLE04] the so-called forcing transform was introduced to test the membership of a function to the class of CFs. Furthermore, some spectral properties of canalizing functions

[1] See Preface of [Rud74]

with respect to network stability have been investigated in [KRYH05].

This chapter is organized as follows: We start with the definition of Boolean random variables and some other basic notations in Section 2.1. In Section 2.2 we define Boolean functions and the Fourier transform. Further, we introduce the concept of restrictions and prove some spectral properties of restricted functions. Several classes of Boolean functions are addressed in Section 2.3. The main focus is on canalizing and nested canalizing functions, which are analyzed in the Fourier domain.

2.1. Basic Notations

Traditionally, Boolean variables are used to describe binary states and hence their possible values are often referred to as $\{TRUE, FALSE\}$ or $\{ON, OFF\}$. In computer science, where a Boolean variable is denoted by a *bit*, their states are represented by $\{0, 1\}$. For our purposes, however, we define a Boolean variable x to be either -1 or $+1$, i.e.,

$$x \in \Omega = \{-1, +1\},$$

where -1 represents $TRUE$ and, respectively, $+1$ means $FALSE$. This choice has no influence on the general validity of our results, however, it turns out that this representation is advantageous as it simplifies the notation.

We denote random Boolean variables with capital letters, i.e., $X \in \Omega$, and their probability mass functions as

$$P_X(x) = \Pr[X = x].$$

Given two jointly distributed random Boolean variables $X, Y \in \Omega$, the conditional probability mass function for Y given X is written as:

$$P_{Y|X}(y|x) = \Pr[Y = y|X = x].$$

Further, by $\mathbf{E}[X]$ we denote the expectation of X and by $\mathbf{Var}[X]$ its variance.

Random Boolean vectors are denoted by capital bold letters, e.g., $\mathbf{X} = (X_1, \ldots, X_n) \in \Omega^n$, where n is a natural number. In general we assume that random vectors are product distributed, i.e.,

$$P_{\mathbf{X}}(x_1, \ldots, x_n) = P_{\mathbf{X}}(\mathbf{x}) = \Pr[\mathbf{X} = \mathbf{x}] = \prod_{i=1}^{n} P_{X_i}(x_i),$$

and, hence, that

$$P_{X_i|X_j}(x_i|x_j) = P_{X_i}(x_i) \quad \text{for all } i, j \in [n], i \neq j,$$

where $[n]$ denotes the set of natural numbers from 1 to n:

$$[n] := \{1, \ldots, n\}.$$

In this context, we abbreviate the expectation of the i-th component of $\mathbf{X}$ by $\mu_i = \mathbf{E}[X_i]$ and the variance by $\sigma_i^2 = \mathbf{Var}[X_i] = 1 - \mu_i^2$. It can been seen that

$$P_{X_i}(x_i) = \frac{1 + x_i \mu_i}{2}. \tag{2.1}$$

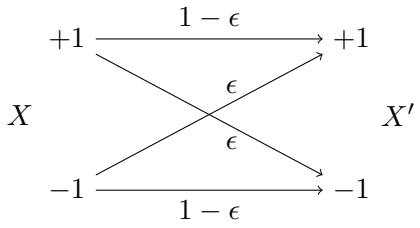


Figure 2.1.: Binary symmetric channel

For the special case of a uniform distribution, $P_{\mathbf{X}}(\mathbf{x}) = 2^{-n}$.

Let us now look at a binary symmetric channel (BSC) with error probability ϵ as depicted in the Figure 2.1. One can see that with probability ϵ the input is flipped, i.e., $x' = -x$. The transition probability of this channel is hence given as:

$$\Pr\left[X' = x' | X = x\right] = \begin{cases} 1-\epsilon & \text{if } \; x' = x \\ \epsilon & \text{else} \end{cases}$$

If we are now considering a random vector $\mathbf{X}$, which is transmitted over n parallel BSCs, i.e., its elements are flipped with probability ϵ, we can give the transition probability as:

$$\Pr\left[\mathbf{X}' = \mathbf{x}' | \mathbf{X} = \mathbf{x}\right] = \prod_{i=1}^{n} \Pr\left[X_i' = x_i' | X_i = x_i\right], \tag{2.2}$$

with

$$\Pr\left[X_i' = x_i' | X_i = x_i\right] = \begin{cases} 1-\epsilon & \text{if } \; x_i' = x_i \\ \epsilon & \text{else.} \end{cases}$$

Finally, let us define a noise-operator, which is equivalent to transmitting a vector over n parallel BSCs:

Definition 2.1 (*[O'D03, Definition 2.1.3]*)

Given a random vector $\mathbf{X} \in \Omega^n$ and $0 \leq \epsilon \leq \frac{1}{2}$, we define the noise-operator $N_\epsilon(\mathbf{X})$ to be the random variable given by flipping each element of $\mathbf{X}$ independently with probability ϵ. Further, the operator, which only denoted the flip of one element i is denoted by $N_{\epsilon,i}(X_i)$.

2.2. Boolean Functions

2.2.1. Basic Definitions and Examples

Let us now define a Boolean function (BF) as a mapping of an n-dimensional binary vector to a binary output:

$$f : \Omega^n \to \Omega, \quad f(\mathbf{x}) := f(x_1, \ldots, x_n) = y,$$

where n is the number of input variables, also called *in-degree*. In general, however, not all input variables have an impact on the output, i.e., are relevant.

Definition 2.2 (*[LAM+13]*)

A variable i is relevant to a BF f, if there exists an $\mathbf{x} \in \Omega^n$ such that

$$f(\mathbf{x}) \neq f(\mathbf{x} \oplus e_i),$$

where $x \oplus e_i$ is the vector obtained from $\mathbf{x}$ by flipping its i-th entry.
Further we define $rel(f) \subseteq [n]$ as the set containing all relevant variables of f. We will denote the number of relevant variables by $k = |rel(f)|$.

If the inputs of a BF are random, e.g., can be described by a random Boolean vector $\mathbf{X}$, we can calculate the expectation of the function's output, often also called *bias* as:

$$\mathbf{E}[f(\mathbf{X})] = \sum_{\mathbf{x} \in \Omega^n} P_{\mathbf{X}}(\mathbf{x}) f(\mathbf{x}).$$

We call a function *balanced*, if its expectation is zero, i.e., if the probabilities for the function's output being either $+1$ or -1, are equal. The following example gives the truth-table representation and the bias for three well-known BFs, namely the AND, OR, and XOR functions, assuming uniformly distributed input vectors.

Example 2.1 (*Uniform distribution*)

x_1	x_2	*AND*	*OR*	*XOR*
$+1$	$+1$	$+1$	$+1$	$+1$
$+1$	-1	$+1$	-1	-1
-1	$+1$	$+1$	-1	-1
-1	-1	-1	-1	$+1$
$\mathbf{E}[f(\mathbf{X})]$		$+\frac{1}{2}$	$-\frac{1}{2}$	0

It can be seen that the XOR function is balanced, while AND, OR are unbalanced. However, by adjusting the input distributions one can make, e.g., the OR function balanced. As displayed in the next example, we can achieve this by setting the probabilities of the input variables to: $P_{X_1}(+1) = P_{X_2}(+1) = \frac{\sqrt{2}}{2}$.

Example 2.2 ($P_{X_1}(+1) = P_{X_2}(+1) = \frac{\sqrt{2}}{2}$)

	AND	*OR*	*XOR*
$\mathbf{E}[f(\mathbf{X})]$	$\approx 0,83$	0	$\approx 0,17$

2.2.2. Fourier Analysis of Boolean Functions

Let us consider a wider set of functions $\mathcal{F}_n = \{f : \Omega^n \to \mathbb{R}\}$, i.e., all functions mapping n-ary input tuples to real numbers. Note that the BFs are a subset of $\mathcal{F}_n$. As seen in the examples, the input distribution plays an important role when investigating and characterizing BFs. Hence, according to [FJS91] we will define a Fourier-like transform for functions in $\mathcal{F}_n$. We

start by defining an inner product of two functions $f, g \in \mathcal{F}_n$ as

$$\langle f, g \rangle_{P_{\mathbf{X}}} = \sum_{\mathbf{x} \in \Omega^n} f(\mathbf{x}) g(\mathbf{x}) P_{\mathbf{X}}(\mathbf{x}).$$

This allows us to define the orthogonal functions $\phi_U(x)$ for any subset $U \subseteq [n]$:

$$\phi_U(\mathbf{x}) = \begin{cases} 1 & U = \emptyset \\ \prod_{i \in U} \frac{x_i - \mu_i}{\sigma_i} & \text{else} \end{cases}. \tag{2.3}$$

The orthogonality follows from the fact that for $U, S \subseteq [n]$

$$\langle \phi_U, \phi_S \rangle_{P_{\mathbf{X}}} = \begin{cases} 1 & U = S \\ 0 & \text{else} \end{cases}.$$

Another important property of these functions follows directly from their definition:
Let $A \subseteq U$, then

$$\phi_U(\mathbf{x}) = \phi_A(\mathbf{x}) \cdot \phi_{U \setminus A}(\mathbf{x}). \tag{2.4}$$

Using $\phi_U(\mathbf{x})$ as a basis we can express any function $f \in \mathcal{F}_n$ by its Fourier-expansion [Tit62, FJS91]:

$$f(\mathbf{x}) = \sum_{U \subseteq [n]} \hat{f}(U) \cdot \phi_U(\mathbf{x}).$$

The *Fourier coefficients* $\hat{f}(U)$ can be recovered by

$$\begin{aligned} \hat{f}(U) &= \sum_{\mathbf{x} \in \Omega^n} P_{\mathbf{X}}(\mathbf{x}) \cdot f(\mathbf{x}) \cdot \phi_U(\mathbf{x}) \\ &= \mathbf{E}\left[f(\mathbf{X}) \cdot \phi_U(\mathbf{X}) \right]. \end{aligned} \tag{2.5}$$

The cardinality of U, i.e., $|U|$ is also called the *order* of a coefficient. From now on we will again only consider BFs, i.e. functions mapping $\Omega^n \to \Omega$.

Then, the square of a coefficient, i.e., $\hat{f}^2(U)$ is often referred to as its *energy*. And, due to Parseval's theorem, we know that the sum of all squares equals one, i.e.,

$$\sum_{U \subseteq [n]} \hat{f}^2(U) = 1.$$

For the special case that the input variables X_i are uniformly distributed, and hence, $\mu_i = 0$ and $\sigma_i = 1$, Eq. (2.3) reduces to

$$\chi_U(\mathbf{x}) \equiv \phi_U(\mathbf{x}) = \prod_{i \in U} x_i,$$

and consequently Eq. (2.5), becomes

$$\hat{f}(U) = 2^{-n} \sum_{\mathbf{x}} f(\mathbf{x}) \cdot \chi_U(\mathbf{x}).$$

In Examples 2.1 and 2.2 we gave the truth-table representation and the expected value of a BF under uniformly and product distributed inputs. Now let's look at their Fourier representations:

Example 2.3 (*Uniform distribution*)

The Fourier coefficients of the AND, OR, and XOR functions with uniformly distributed inputs are given by:

U	AND	OR	XOR
$\emptyset$	$1/2$	$-1/2$	0
$\{1\}$	$1/2$	$1/2$	0
$\{2\}$	$1/2$	$1/2$	0
$\{1,2\}$	$-1/2$	$1/2$	1

Equivalently written in polynomial form:

AND	OR	XOR
$1/2 + 1/2x_1 + 1/2x_2 - 1/2x_1x_2$	$-1/2 + 1/2x_1 + 1/2x_2 + 1/2x_1x_2$	x_1x_2

Example 2.4 ($P_{X_1}(+1) = P_{X_2}(+1) = \frac{\sqrt{2}}{2}$)

Assuming an input distribution of $P_{X_1}(+1) = P_{X_2}(+1) = \frac{\sqrt{2}}{2}$, the Fourier coefficients of the AND, OR, and XOR functions are given as:

U	AND	OR	XOR
$\emptyset$	≈ 0.83	0	≈ 0.17
$\{1\}$	≈ 0.27	≈ 0.64	≈ 0.38
$\{2\}$	≈ 0.27	≈ 0.64	≈ 0.38
$\{1,2\}$	≈ -0.41	≈ 0.41	≈ 0.83

Comparing the zero order coefficient $\hat{f}(\emptyset)$ in Examples 2.1 and 2.2 with the values of their corresponding expectation a direct relation between these entities becomes visible. In fact, from the definition of the expectation and Eq. (2.5) it follows that:

$$\mathbf{E}\left[f(\mathbf{X})\right] = \hat{f}(\emptyset) = \sum_{\mathbf{x}\in\Omega^n} P_{\mathbf{X}}(\mathbf{x}) \cdot f(\mathbf{x}) \cdot \underbrace{\phi_\emptyset(\mathbf{x})}_{=1}, \tag{2.6}$$

hence, the bias of a BF is directly given by its zero order coefficient $\hat{f}(\emptyset)$. Let us now look at the variance of $f(\mathbf{X})$, which is defined as

$$\mathbf{Var}\left[f(\mathbf{X})\right] = \mathbf{E}\left[f^2(\mathbf{x})\right] - \mathbf{E}\left[(f(\mathbf{x}))\right]^2.$$

Since $f^2(\mathbf{x}) = 1$ for all $\mathbf{x}$, we get

$$\mathbf{Var}\left[f(\mathbf{X})\right] = 1 - \mathbf{E}\left[(f(\mathbf{x}))\right]^2 = 1 - \hat{f}^2(\emptyset).$$

A simple way to calculate the Fourier coefficients in the uniform case is the (Fast) Walsh Hadamard transform [Sha69]. Let $\hat{\mathbf{f}}$ be a vector containing the Fourier coefficients, i.e.,

$$\hat{\mathbf{f}} = \left(\hat{f}(\emptyset), \hat{f}(\{1\}), \hat{f}(\{2\}), \hat{f}(\{1,2\}), \ldots, \hat{f}(\{[n]\})\right)^T,$$

and $\mathbf{f}$ be a vector containing the function values for all inputs, i.e.,

$$\mathbf{f} = \left(f(+1,\ldots,+1,+1), f(+1,\ldots,+1,-1), f(+1,\ldots,-1,+1), \ldots, f(-1,\ldots,-1,-1)\right)^T,$$

then $\hat{\mathbf{f}}$ can be obtained by multiplying $\mathbf{f}$ with the corresponding Walsh Hadamard matrix:

$$\hat{\mathbf{f}} = \frac{1}{2^n} H_n \cdot \mathbf{f}.$$

One prerequisite for the fast computation of this equation is the recursive defined Walsh Hadamard matrix H_n:

$$H_0 = [1],$$
$$H_m = \begin{bmatrix} H_{m-1} & H_{m-1} \\ H_{m-1} & -H_{m-1} \end{bmatrix}, \text{ for } m > 0.$$

This can be generalized for the product distributed case [Hec10]. Therefore, we define a matrix W_n as:

$$W_0 = [1],$$
$$W_m = \begin{bmatrix} P_{X_m}(+1) \cdot W_{m-1} & P_{X_m}(-1) \cdot W_{m-1} \\ \sqrt{P_{X_m}(+1) \cdot P_{X_m}(-1)} \cdot W_{m-1} & -\sqrt{P_{X_m}(+1) \cdot P_{X_m}(-1)} \cdot W_{m-1} \end{bmatrix}, \text{ for } m > 0.$$

Then, the Fourier coefficients can be obtained by

$$\hat{\mathbf{f}} = W_n \cdot \mathbf{f}.$$

2.2.3. Restrictions of Boolean Functions

In this subsection we will discuss the concept of restricted functions and the implications for the Fourier coefficients. In particular, we will give relations between the Fourier coefficients of the original function and its restrictions and vice versa. A function is called restricted, if an input variable is fixed. Thus, the number of relevant variables is reduced by one. We call a BF $f^{(i,a_i)}$ a restriction of f, if it is obtained by setting the i-th input variable of f to some constant $a_i \in \Omega$. Every BF can be decomposed into two unique restricted functions for each relevant variable, as stated by the following proposition:

Proposition 2.1

For any BF f and each $i \in [n]$ there exist unique functions $f^{(i,+)}, f^{(i,-)} \in \mathcal{F}_n$, with $i \notin rel(f^{(i,+)})$ and $i \notin rel(f^{(i,-)})$, such that[2]

$$f(\mathbf{x}) = g^{(i,+)}(\mathbf{x}) f^{(i,+)}(\mathbf{x}) + g^{(i,-)}(\mathbf{x}) f^{(i,-)}(\mathbf{x}),$$

where the functions $g^{(i,+)}, g^{(i,-)} \in \mathcal{F}_n$ are given by

$$g^{(i,+)}(\mathbf{x}) = \begin{cases} 1 & \text{if } x_i = 1 \\ 0 & \text{else} \end{cases} \quad \text{and} \quad g^{(i,-)}(\mathbf{x}) = \begin{cases} 1 & \text{if } x_i = -1 \\ 0 & \text{else} \end{cases}.$$

Proof: The proof follows directly from the definitions of $g^{(i,+)}$ and $g^{(i,-)}$. ■

Let us now characterize the Fourier coefficients of these restricted functions. The following theorem gives a relation between the coefficients of the original function and its restriction.

[2] for the sake of readability we abbreviate $f^{(i,-1)}$ with $f^{(i,-)}$, etc.

Theorem 2.1

Let f be a BF with n variables and $f^{(i,a_i)}$ the restricted function obtained by setting $x_i = a_i$, $f^{(i,a_i)}$, then

$$\hat{f}^{(i,a_i)}(U) = \hat{f}(U) + \phi_{\{i\}}(a_i) \cdot \hat{f}(U \cup \{i\}), \tag{2.7}$$

where $U \subseteq [n] \setminus \{i\}$ and $\phi_{\{i\}}(a_i) = \frac{a_i - \mu_i}{\sigma_i}$.

Proof: Using the definition of the Fourier coefficients (Eq. (2.5)), we can rewrite Eq. (2.7) as:

$$\hat{f}^{(i,a_i)}(U) = \sum_{\mathbf{x}} P_{\mathbf{X}}(\mathbf{x}) \cdot f(\mathbf{x}) \cdot \phi_U(\mathbf{x}) + \phi_{\{i\}}(a_i) \cdot \sum_{\mathbf{x}} P_{\mathbf{X}}(\mathbf{x}) \cdot f(\mathbf{x}) \cdot \phi_{U \cup \{i\}}(\mathbf{x}). \tag{2.8}$$

By applying Eq. (2.4) and Eq. (2.3) we get

$$\begin{aligned} \phi_{U \cup \{i\}}(\mathbf{x}) &= \phi_U(\mathbf{x}) \cdot \phi_{\{i\}}(\mathbf{x}) \\ &= \phi_U(\mathbf{x}) \cdot \frac{x_i - \mu_i}{\sigma_i}. \end{aligned}$$

Hence, we can combine the two sums in Eq. (2.8) and obtain:

$$\begin{aligned} \hat{f}^{(i,a_i)}(U) &= \sum_{\mathbf{x}} P_{\mathbf{X}}(\mathbf{x}) \cdot f(\mathbf{x}) \cdot \phi_U(\mathbf{x}) \cdot \left(1 + \phi_{\{i\}}(a_i) \cdot \frac{x_i - \mu_i}{\sigma_i}\right) \\ &= \sum_{\mathbf{x}} (P_{\mathbf{X}}(\mathbf{x}) \cdot f(\mathbf{x}) \cdot \phi_U(\mathbf{x}) \cdot \Xi_i), \end{aligned} \tag{2.9}$$

where

$$\Xi_i = 1 + \phi_{\{i\}}(a_i) \cdot \frac{x_i - \mu_i}{\sigma_i}.$$

Since,

$$\phi_{\{i\}}(a_i) = \frac{a_i - \mu_i}{\sigma_i} = \frac{a_i - \mu_i}{\sigma_i} \cdot \frac{a_i + \mu_i}{a_i + \mu_i} = \frac{\overbrace{a_i^2}^{=1} - \mu_i^2}{\sigma_i \cdot (a_i + \mu_i)} = \frac{\sigma_i}{a_i + \mu_i},$$

we get

$$\Xi_i = \left(1 + \frac{x_i - \mu_i}{a_i + \mu_i}\right) = \frac{a_i + x_i}{a_i + \mu_i}.$$

Let's now investigate Ξ_i. Since $a_i \in \Omega$ we rewrite $a_i = \frac{1}{a_i}$ and together with Eq. (2.1), we get:

$$\Xi_i = \begin{cases} \frac{2}{1 + a_i \cdot \mu_i} = \frac{1}{P_{X_i}(a_i)} & \text{, if } x_i = a_i \\ 0 & \text{, if } x_i = -a_i \end{cases}.$$

Thus, the sum in Eq. (2.9) can be simplified to

$$\hat{f}^{(i,a_i)}(U) = \sum_{\mathbf{x} | x_i = a_i} \frac{P_{\mathbf{X}}(\mathbf{x})}{P_{X_i}(a_i)} \cdot f(\mathbf{x}) \cdot \phi_U(\mathbf{x})$$

and finally to

$$\hat{f}^{(i,a_i)}(U) = \sum_{\mathbf{x}|x_i=a_i} P_{\mathbf{X}|x_i}(\mathbf{x}|a_i) \cdot f(\mathbf{x}) \cdot \phi_U(\mathbf{x}),$$

which is the definition of the Fourier coefficients from Eq. (2.5) and this concludes the proof. ■

This theorem enables us to calculate the Fourier coefficients of the restricted functions given the coefficients of the original function. The reverse approach is shown in the following proposition. Please note that this result for uniformly distributed input variables can also be retrieved using [Ber98, Lemma 2.17].

Proposition 2.2

Let $i \in [n]$ be fixed. For any n-ary Boolean function f,

$$\hat{f}(U) = \sum_{a_i \in \Omega} P_{X_i}(a_i) \cdot \left(\phi_{\{i\}}(a_i)\right)^{|U \cap \{i\}|} \cdot \hat{f}^{(i,a_i)}(U \setminus \{i\}),$$

or

$$\hat{f}(U) = \mathop{\mathbf{E}}_{x_i}\left[\left(\phi_{\{i\}}(X_i)\right)^{|U \cap \{i\}|} \cdot \hat{f}^{(i,X_i)}(U \setminus \{i\})\right].$$

Proof: Starting from the definition of the Fourier coefficients (Eq. (2.5)), we obtain

$$\begin{aligned}\hat{f}(U) =& \mathop{\mathbf{E}}_{\mathbf{x}}[f(\mathbf{X}) \cdot \phi_U(\mathbf{X})] \\ =& P_{X_i}(+1) \cdot \mathop{\mathbf{E}}_{\mathbf{x}|x_i=+1}[f(\mathbf{X}) \cdot \phi_U(\mathbf{X})|X_i = +1] \\ &+ P_{X_i}(-1) \cdot \mathop{\mathbf{E}}_{\mathbf{x}|x_i=-1}[f(\mathbf{X}) \cdot \phi_U(\mathbf{X})|X_i = -1].\end{aligned}$$

If $i \in U$, we can write

$$\mathop{\mathbf{E}}_{\mathbf{x}|x_i=a_i}[f(\mathbf{X}) \cdot \phi_U(\mathbf{X})|X_i = a_i] = \phi_{\{i\}}(a_i) \mathop{\mathbf{E}}_{\mathbf{x}|x_i=a_i}\left[f^{(i,a_i)}(\mathbf{X}) \cdot \phi_{U\setminus\{i\}}(\mathbf{X})\right],$$

and if $i \notin U$

$$\mathop{\mathbf{E}}_{\mathbf{x}|x_i=a_i}[f(\mathbf{X}) \cdot \phi_U(\mathbf{X})|X_i = a_i] = \mathop{\mathbf{E}}_{\mathbf{x}|x_i=a_i}\left[f^{(i,a_i)}(\mathbf{X}) \cdot \phi_{U\setminus\{i\}}(\mathbf{X})\right].$$

Hence,

$$\begin{aligned}\hat{f}(U) =& P_{X_i}(+1) \cdot \phi_{\{i\}}(+1)^{|U\cap\{i\}|} \mathop{\mathbf{E}}_{\mathbf{x}|x_i=a_i}\left[f^{(i,+1)}(\mathbf{X}) \cdot \phi_{U\setminus\{i\}}(\mathbf{X})\right] \\ &+ P_{X_i}(-1) \cdot \phi_{\{i\}}(-1)^{|U\cap\{i\}|} \mathop{\mathbf{E}}_{\mathbf{x}|x_i=a_i}\left[f^{(i,-1)}(\mathbf{X}) \cdot \phi_{U\setminus\{i\}}(\mathbf{X})\right].\end{aligned}$$

Note that for $a_i = +1$ or $a_i = -1$

$$\mathop{\mathbf{E}}_{\mathbf{x}|x_i=-1}\left[f^{(i,a_i)}(\mathbf{X}) \cdot \phi_{U\setminus\{i\}}(\mathbf{X})\right] = \hat{f}^{(i,a_i)}(U \setminus \{i\})$$

by definition, which concludes the proof. ■

An immediate corollary of Proposition 2.2 shows that the zero coefficient of a function only depends on the zero coefficients of the restricted functions:

Corollary 2.1

The zero Fourier coefficient of any BF f can be written as:

$$\hat{f}(\emptyset) = P_{X_i}(+1) \cdot \hat{f}^{(i,+1)}(\emptyset) + P_{X_i}(-1) \cdot \hat{f}^{(i,-1)}(\emptyset),$$

where $i \in [n]$ is the index of some variable.

In the general case, if we restrict a function to more than one variable, namely to a set of variables $K \subset rel(f)$, we denote the restricted function with $f^{(K,A)}$, where A is a set containing the values to which the function is restricted. Consequently, the number of relevant variables of $f^{(K,A)}$ is given by $|rel(f)| - |K|$. The Fourier coefficients of $f^{(K,A)}$ are given by the following proposition:

Proposition 2.3

Let f be a BF and $\hat{f}(U)$ its Fourier coefficients. Furthermore, let K be a set containing the indices i of the input variables x_i, which are fixed to certain values a_i. The Fourier coefficients of the restricted function $f^{(K,A)}$ are then given as:

$$\hat{f}^{(K,A)}(U) = \sum_{S \subseteq K} \left(\Phi_S(A) \cdot \hat{f}(U \cup S) \right),$$

where A is the set containing all $a_i, i \in K$.

Proof: The claim follows from recursive application of Theorem 2.1. ■

2.3. Classes of Boolean Functions

In this section we will introduce and discuss different classes of Boolean functions, namely *canalizing* functions (CFs), their subclass, the *nested canalizing* functions (NCFs), and *unate* functions. The section is organized as follows: In Subsection 2.3.1, we will first introduce canalizing functions and show important spectral properties, then we will introduce the notion of *jointly canalizing functions*. In Subsection 2.3.2 we will define the nested canalizing functions, show their spectral properties and discuss some important subclasses. Finally we will shortly introduce unate functions (Subsection 2.3.3).

2.3.1. Canalizing Functions

As an example let us first look at the function $f(\mathbf{x}) = (\mathrm{NOT}x_1)\mathrm{AND}(x_2\mathrm{XOR}x_3)$, whose truth-table is given as follows:

x_1	x_2	x_3	(NOTx_1)AND(x_2XORx_3)
$+1$	$+1$	$+1$	$+1$
$+1$	$+1$	-1	-1
$+1$	-1	$+1$	-1
$+1$	-1	-1	$+1$
-1	$+1$	$+1$	$+1$
-1	$+1$	-1	$+1$
-1	-1	$+1$	$+1$
-1	-1	-1	$+1$

One can see that the function's output is always $+1 = b_1$, if x_1 equals $-1 = a_1$, i.e., if the restricted function $f^{(1,a_1)}$ is a constant. We say that $f(\mathbf{x})$ is canalizing in x_1 with canalizing value a_1 and canalized value b_i, respectively. More formally:

Definition 2.3 (*[LAM+13]*)

A BF is called $< i : a_i : b_i >$ canalizing, if there exists a canalizing variable x_i and constant Boolean values $a_i, b_i \in \Omega$ such that

$$f^{(i,a_i)}(\mathbf{x}) = b_i,$$

for all $x_1, ...x_{i-1}, x_{i+1}....x_n$.

It is quite obvious that all functions with $n = 2$ variables are canalizing, except for XOR and its negation. However, this changes for larger n. It has been shown in [ACK03] that there are at most $4n \cdot 2^{2^{(n-1)}}$ canalizing functions. As there exist 2^{2^n} BF in total, it becomes clear that the fraction of CFs is at most $\frac{4n}{2^{2^{n-1}}}$ and hence double exponentially decreasing with n. In [JSK04] it has been shown that the exact number of CFs is given as

$$\#\mathrm{CF} = 2\left((-1)^n - n\right) - \sum_{j=1}^{n} (-1)^{j+1} \binom{n}{j} 2^{j+1} 2^{2^{n-j}}.$$

Thus, as the fraction of CFs is double exponential decreasing, it is quite unlikely to actually get a CF for larger n, when drawing BFs randomly.

2.3.1.1. Spectral Properties of Canalizing Functions

As mentioned before, the authors of [SLE04] proposed a test of the membership of a BF in the class of CFs using the so-called forcing transform. We will now generalize this to the Fourier transform, which is a more intuitive and natural approach and gives a relation between the Fourier coefficients and the canalizing property:

Theorem 2.2

A BF f is $< i : a_i : b_i >$ canalizing in variable x_i, with $a_i, b_i \in \Omega$, if and only if the Fourier coefficients $\hat{f}(U)$ fulfill the following condition:

$$\hat{f}(\emptyset) + \phi_{\{i\}}(a_i) \cdot \hat{f}(\{i\}) = b_i.$$

Proof: Using Definition 2.3 we can state that a function is canalizing, if and only if

$$\mathbf{E}\left[f^{(i,a_i)}(\mathbf{X})\right] = b_i.$$

By applying the relation described in Eq. (2.6), we get

$$\hat{f}^{(i,a_i)}(\emptyset) = b_i.$$

From Theorem 2.1 we know that

$$\hat{f}^{(i,a_i)}(\emptyset) = \hat{f}(\emptyset) + \phi_{\{i\}}(a_i) \cdot \hat{f}(\{i\}),$$

which concludes the proof. ■

Further, it can be seen that in the uniformly distributed case

$$b_i = \operatorname{sgn}\left(\hat{f}(\emptyset)\right), \tag{2.10}$$

where $\operatorname{sgn}(\cdot)$ gives the sign.

A similar result, namely the calculation of the Fourier coefficients of a canalizing BF from the coefficients of the restricted functions $\hat{f}^{(i,a_i)}(U)$, is addressed in [KRYH05]. These results can also be achieved using Proposition 2.2.

Theorem 2.2 shows that the canalizing property can be tested by inspecting all Fourier coefficients of order one. Using the Fast Walsh Transform (FWT) [Sha69], this test has the same computational complexity as the one presented in [SLE04].

2.3.1.2. Jointly Canalizing Functions

If a BF is not canalizing in a single variable, but two or more variables act together as canalizing influence, we call this function *jointly canalizing*.

Definition 2.4

A BF f with n variables is called jointly canalizing in a set of variables $T \subset [n]$, if there exists a constant $b_T \in \Omega$ and a set of Boolean restrictive values $A = \{a_i \in \Omega : i \in T\}$ such that

$$f^{(T,A)}(\mathbf{x}) = b_T,$$

for all x_i, $i \in [n] \setminus T$, where $f^{(T,A)}(\mathbf{x})$ is a BF, whose inputs $x_j, j \in T$ are fixed to $x_j = a_j$.

Similar to CFs, we can prove the conditions jointly canalizing functions have to fulfill in the Fourier domain.

Proposition 2.4

A BF f is jointly canalizing in $T \subset [n]$, if there exists a set $A = \{a_i \in \Omega : i \in T\}$, a constant $b_T \in \Omega$ and its Fourier coefficients fulfill the following condition:

$$\sum_{U \subseteq T} \phi_S(A)\hat{f}(U) = b_T.$$

Proof: From Definition 2.4, it follows directly that

$$\mathbf{E}\left[f^{(T,A)}(\mathbf{X})\right] = b_T.$$

Since the zero order coefficient gives the expectation of a BF (see Eq. (2.6)) it follows that

$$\hat{f}^{(T,A)}(\emptyset) = b_T.$$

From Proposition 2.3 we know:

$$\hat{f}^{(T,A)}(\emptyset) = \sum_{U \subseteq T} \phi_U(A) \cdot \hat{f}(U),$$

which concludes the proof. ■

2.3.2. Nested Canalizing Functions

Nested canalizing functions (NCFs), which form a subclass of CFs, were introduced by Kauffman et al. [KPST03] and they also appear in a large number of regulatory networks. Since then, they have been studied mainly with respect to their positive influence on the network stability [KPST04, LAM+13]. Further, in [JRL07] it has been shown that the class of NCFs is identical to the class of the so-called *unate cascade* BFs, which are used in the area of logic design [Mai62, Muk69] and show an optimal average path length in binary decision diagrams [BSM05]. Hence, NCFs are also a subclass of unate functions, which are defined later.

Before formally introducing NCFs, let us examine the following example to understand the concept of NCFs.

Example 2.5

The truth-table below represents a BF f, which is clearly canalizing in x_1. The canalizing value is $a_1 = -1$, the canalized value $b_1 = +1$. One can see from the truth-table that the function is not canalizing in either x_2 or x_3. However, if we restrict f to $x_1 = -a_1 = +1$ the restricted BF $f^{(1,+1)}$ is canalizing in x_2 with canalizing value $a_2 = +1$ and $b_2 = -1$. If we now restrict $f^{(1,+1)}$ again, we get another CF. Hence, we call f nested canalizing.

x_1	x_2	x_3	$f(x)$
$+1$	$+1$	$+1$	-1
$+1$	$+1$	-1	-1
$+1$	-1	$+1$	-1
$+1$	-1	-1	$+1$
-1	$+1$	$+1$	$+1$
-1	$+1$	-1	$+1$
-1	-1	$+1$	$+1$
-1	-1	-1	$+1$

Now, we let's give the definition of NCFs:

Definition 2.5

For $k = 1$ and $k = 0$ any BF with $k \leq n$ relevant variables is a NCF.
For $k > 1$ a BF is a NCF, if there exists at least one variable i and constants $a_i, b_i \in \Omega$, such that $f^{(i,a_i)} = b_i$ and $f^{(i,-a_i)}$ is a NCF with $k-1$ relevant variables.

Let π be a permutation of length k and $x_{\pi(1)}, \ldots, x_{\pi(k)}$ the variable order for which a NCF fulfills the properties of Definition 2.5, then we call, following [LAM$^+$13], such a function $\{\pi : A : B\}$ nested canalizing, where A and B are sets containing all a_i and b_i, respectively. For convenience we will abbreviate $\pi(i)$ by π_i.

Example 2.6 (*[LAM$^+$13]*)

Let f be a BF with the following truth-table:

x_1	x_2	x_3	$f(x)$
$+1$	$+1$	$+1$	$+1$
$+1$	$+1$	-1	$+1$
$+1$	-1	$+1$	-1
$+1$	-1	-1	$+1$
-1	$+1$	$+1$	-1
-1	$+1$	-1	-1
-1	-1	$+1$	-1
-1	-1	-1	-1

We can see that f is a $\{\pi : A : B\} = \{(1,2,3) : (-1,+1,-1) : (-1,+1,+1)\}$-NCF as well as a $\{(1,3,2) : (-1,-1,-1) : (-1,+1,-1)\}$-NCF.

Similar to Theorem 2.2, we can give the spectral properties of NCFs:

Proposition 2.5

Let f be a BF, then f is a $\{\pi : A : B\}$-NCF, if and only if the Fourier coefficients fulfill the following condition for each $j \in [n]$:

$$\sum_{S \subseteq [j]} \left(\phi_{\{\pi(j)\}}(a_j)^{|\widetilde{S} \cap \{\pi(j)\}|} \cdot \phi_{\widetilde{S} \setminus \{\pi_j\}}(\bar{A}) \cdot \hat{f}(\widetilde{S}) \right) = b_j,$$

where $\bar{A}$ is a set containing all negated a_i, i.e., $\bar{a}_i = -a_i$ and $\widetilde{S}$ is a set, which is retrieved by applying the permutation π to the elements of S.

Proof: The proof follows from Proposition 2.3 and Theorem 2.2. ■

This proposition does not give a simple test like Theorem 2.2, where only one equation has to be fulfilled. Here, we have a set of k equations, which have to be fulfilled. In order to illustrate the spectral properties of a NCF, consider the following two examples:

Example 2.7

Let f be a $\{\pi : A : B\}$-NCF with $k = 3$ uniformly distributed relevant variables and π such that $\tilde{S} = \pi(S) = S$, then

$$\begin{aligned}
b_1 =& \hat{f}(\emptyset) + a_1\hat{f}(\{1\}) \\
b_2 =& \hat{f}(\emptyset) - a_1\hat{f}(\{1\}) + a_2\hat{f}(\{2\}) - a_1a_2\hat{f}(\{1,2\}) \\
b_3 =& \hat{f}(\emptyset) - a_1\hat{f}(\{1\}) - a_2\hat{f}(\{2\}) + a_3\hat{f}(\{3\}) + a_1a_2\hat{f}(\{1,2\}) - a_1a_3\hat{f}(\{1,3\}) \\
& - a_2a_3\hat{f}(\{2,3\}) + a_1a_2a_3\hat{f}(\{1,2,3\}).
\end{aligned}$$

Example 2.8

Let us consider the BF from Example 2.5, assuming uniformly distributed variables. Hence, the corresponding Fourier coefficients are given as:

S	$\emptyset$	$\{1\}$	$\{2\}$	$\{1,2\}$	$\{3\}$	$\{1,3\}$	$\{2,3\}$	$\{1,2,4\}$
$\hat{f}(S)$	$+0.25$	-0.75	-0.25	-0.25	-0.25	-0.25	$+0.25$	$+0.25$

Set $a_3 = +1$ and $b_3 = +1$, then from the previous Example 2.7, we get

$$\begin{aligned}
\hat{f}(\emptyset) - \hat{f}(\{1\}) &= +1 \\
\hat{f}(\emptyset) + \hat{f}(\{1\}) + \hat{f}(\{2\}) + \hat{f}(\{1,2\}) &= -1 \\
\hat{f}(\emptyset) + \hat{f}(\{1\}) - \hat{f}(\{2\}) + \hat{f}(\{3\}) - \hat{f}(\{1,2\}) + \hat{f}(\{1,3\}) - \hat{f}(\{2,3\}) - \hat{f}(\{1,2,3\}) &= +1,
\end{aligned}$$

which shows that f is a NCF.

In this example we picked $a_3 = +1$. However, we could have also picked $a_3 = -1$. The functions would have still been a NCF, as every BF with only one variable is a CF. This concept is formally expressed in the following proposition taken from [LAM+13].

Proposition 2.6 (*[LAM+13, Proposition 3.4]*)

The function f is a $\{\pi : A : B\}$-NCF, if and only if f is a $\{\pi : A^n : B^n\}$-NCF, where A^n, B^n denote the sets A, B, respectively, with the n-th element flipped, i.e., $a_n = -a_n$ and $b_n = -b_n$.

2.3.2.1. Properties of Nested Canalizing Functions

Let us now state some properties of NCFs. From their definition it becomes clear that NCFs are somehow recursive. The following proposition shows the recursive behavior of the Fourier coefficients of NCFs.

Proposition 2.7

The Fourier coefficients of a $\{\pi : A : B\}$-NFC f can be recursively written as:

$$\hat{f}(U) = \begin{cases} P_{X_{\pi_1}}(\bar{a}_1) \cdot \left(\phi_{\{\pi_1\}}(\bar{a}_1)\right)^{|U \cap \{\pi_1\}|} \cdot \hat{f}^{(\pi_1, \bar{a}_1)}(U \setminus \{\pi_1\}) & U \neq \emptyset \\ P_{X_{\pi_1}}(\bar{a}_1) \cdot \hat{f}^{(\pi_1, \bar{a}_1)}(\emptyset) + P_{X_{\pi_1}}(a_1) \cdot b_1 & \text{else} \end{cases}.$$

Proof: The claim follows from Proposition 2.2 and the fact that the restriction $f^{(\pi_1,a_1)}$ of a NCF is a constant function. ■

If we look closer at the zero coefficient for uniformly distributed variables, we see from the following proposition that it can be bounded as follows:

Proposition 2.8

The absolute value of the zero coefficient of a NCF f with $k > 1$ relevant and uniformly distributed input variables can be bounded as:

$$\frac{1}{2^{k-1}} \leq |\hat{f}(\emptyset)| \leq 1 - \frac{1}{2^{k-1}}.$$

Proof: First, we prove the right hand side: Using the triangle inequality we get from Proposition 2.7:

$$|\hat{f}(\emptyset)| \leq \frac{1}{2}|\hat{f}^{(\pi_1,\alpha_1)}(\emptyset)| + \frac{1}{2}.$$

Obviously the zero coefficient of a function with only one relevant variable i is zero. The proposition now follows by induction. The left hand side can be shown using the inverse triangle inequality and induction. ■

We will see later (Corollaries 2.3 and 2.5) that there exist two subclasses of the NCFs that fulfill the upper and lower bound of this proposition with equality. We only investigate these bounds for the uniform case, since the zero coefficient, i.e., the expectation of a BF, in the product distributed case, can take any real value between -1 and $+1$, be choosing the input distributions accordingly (see Example 2.2).

Next, we address most dominant variables, which are defined as follows:

Definition 2.6 (*[LAM$^+$13, Def 4.5]*)

The variable i is called a most dominant variable of f, if there exists a permutation π, such that $\pi(1) = i$, for which f is $\{\pi : A : B\}$ nested canalizing.

Hence, the set of most dominant variables of a BF f, is the set of variables in which f is canalizing. This set has an impact on a number of Fourier coefficients, which is summarized in the following proposition:

Proposition 2.9

Let K be the set of most dominant variables of a $\{\pi : A : B\}$-NCF f. Then the corresponding Fourier coefficients all fulfill the following property , i.e., for all $U \subseteq K$, $U \neq \{\emptyset\}$:

$$(-1)^{|U|-1}\phi_U(A) \cdot \hat{f}(U) = c, \qquad c \in \mathbb{R}, \tag{2.11}$$

and

$$\hat{f}(\emptyset) = b - c, \tag{2.12}$$

where $b \in \Omega$ is a constant and

$$b_i = b \quad \forall i \in K. \tag{2.13}$$

Proof: Let us first show Eq. (2.13). Let $i, j \in K$, then from Theorem 2.2 we know:

$$\hat{f}(\emptyset) + \phi_i(a_i) \cdot \hat{f}(\{i\}) = b_i \tag{2.14}$$
$$\hat{f}(\emptyset) + \phi_j(a_j) \cdot \hat{f}(\{j\}) = b_j \tag{2.15}$$

Further the restrictions of f to $x_i = -a_i$ is canalizing for $x_j = a_j$ and vice versa, hence:

$$\hat{f}^{(j,-a_j)}(\emptyset) + \phi_i(a_i) \cdot \hat{f}^{(j,-a_j)}(\{i\}) = b_i$$
$$\hat{f}^{(i,-a_i)}(\emptyset) + \phi_j(a_j) \cdot \hat{f}^{(i,-a_i)}(\{j\}) = b_j,$$

applying Theorem 2.1 we get:

$$\hat{f}(\emptyset) + \phi_j(-a_j) \cdot \hat{f}(\{j\}) + \phi_i(a_i) \cdot \hat{f}(\{i\}) + \phi_j(-a_j) \cdot \phi_i(a_i) \cdot \hat{f}(\{i,j\}) = b_i$$
$$\hat{f}(\emptyset) + \phi_i(-a_i) \cdot \hat{f}(\{i\}) + \phi_j(a_j) \cdot \hat{f}(\{j\}) + \phi_i(-a_i) \cdot \phi_j(a_j) \cdot \hat{f}(\{i,j\}) = b_j.$$

Using Eq. (2.14) and (2.15) yields:

$$\hat{f}(\{j\}) + \phi_i(a_i) \cdot \hat{f}(\{i,j\}) = 0 \tag{2.16}$$
$$\hat{f}(\{i\}) + \phi_j(a_j) \cdot \hat{f}(\{i,j\}) = 0. \tag{2.17}$$

Solving Eq. (2.16) for $\hat{f}(\{i,j\})$ and substituting it into Eq. (2.17) leads to:

$$\phi_i(a_i) \cdot \hat{f}(\{i\}) = \phi_j(a_j) \cdot \hat{f}(\{j\}) = c.$$

From this, Eq. (2.11) follows directly for $|U| = 1$ and together with Theorem 2.2 it proves Eqs. (2.12) and (2.13).

We will show the remaining parts by induction and use Eq. (2.11) as the induction hypothesis for Fourier coefficients with order $|U|$ and smaller. We need to show that Eq. (2.11) is also valid for coefficients with order $|U| + 1$. Using Proposition 2.3,

$$\hat{f}^{(U,\bar{A})}(T) = \sum_{S \subseteq U} \left(\phi_S(\bar{A}) \cdot \hat{f}(S \cup U) \right),$$

and that, if f is canalizing in some variable $i \in K$, it follows that every restriction of f must also be canalizing in variable i, i.e., $\hat{f}^{(U,\bar{A})}(\emptyset) + \phi_i(a_i) \cdot \hat{f}^{(U,\bar{A})}(\{i\}) = b$, we get:

$$b = \sum_{S \subseteq U} \left(\phi_S(\bar{A}) \cdot \hat{f}(S) \right) + \phi_i(a_i) \sum_{S \subseteq U} \left(\phi_S(\bar{A}) \hat{f}(S \cup \{i\}) \right)$$
$$b = \sum_{S \subseteq U, S \neq \emptyset} \phi_S(\bar{A}) \left(\hat{f}(S) + \phi_i(a_i) \cdot \hat{f}(S \cup \{i\}) \right) + \hat{f}(\emptyset) + \phi_i(a_i) \cdot \hat{f}(\{i\})$$
$$0 = \sum_{S \subseteq U, S \neq \emptyset} \phi_S(\bar{A}) \left(\hat{f}(S) + \phi_i(a_i) \cdot \hat{f}(S \cup \{i\}) \right).$$

Now multiplying with $(-1)^{|U|-1} \cdot \phi_U(A) = (-1)^{|U \setminus S|} \cdot (-1)^{|S|-1} \cdot \phi_{U \setminus S}(A) \cdot \phi_S(A)$ yields:

$$0 = \sum_{S \subseteq U, S \neq \emptyset} \phi_S(\bar{A}) \cdot (-1)^{|U \setminus S|} \cdot \phi_{U \setminus S}(A) \Big((-1)^{|S|-1} \cdot \phi_S(A) \cdot \hat{f}(S)$$
$$- (-1)^{|S|} \cdot \phi_{S \cup \{i\}}(A) \cdot \hat{f}(S \cup \{i\}) \Big). \tag{2.18}$$

For all S and for all $|S \cup \{i\}| \leq |U|$ we can apply the induction hypothesis, hence, all summands of the sum in Eq. (2.18) are zero, except for the one where $S = U$. This leads us to:

$$0 = \phi_U(\bar{A}) \left(c - (-1)^{|U \cup \{i\}|-1} \phi_{U \cup \{i\}}(A) \hat{f}(U \cup \{i\}) \right),$$

and hence to

$$c = (-1)^{|U \cup \{i\}|-1} \phi_{U \cup \{i\}}(A) \hat{f}(U \cup \{i\}),$$

which concludes to proof. ■

If we restrict our observations to uniformly distributed inputs the following corollary applies:

Corollary 2.2

Let K be the set of most dominant variables of a $\{\pi : A : B\}$-NCF f with uniformly distributed inputs. Then the absolute values of the corresponding Fourier coefficients are all equal, i.e., $\forall U \subseteq K$,

$$\left| \hat{f}(U) \right| = \begin{cases} 1 - c & \text{if } U = \emptyset \\ c & \text{else} \end{cases}$$

where

$$c > 0.$$

2.3.2.2. Special cases of Nested Canalizing Functions

Now, let us examine some special cases or subclasses of NCFs as we will see later that these functions have special properties with respect to noise sensitivity, average sensitivity and mutual information.

NCFs with only most dominant variables (NCF-MDV) The first class we are examining, is the class of NCFs, whose relevant variables are all most dominant (see Definition 2.6). For these functions we can infer the following two corollaries from Proposition 2.9:

Corollary 2.3

Let f be a $\{\pi : A : B\}$-NCF with n uniformly distributed variables of which k are relevant, and all relevant variables are most canalizing, i.e., $b_i = b$ for all $i \in rel(f)$. Then the absolute values of the Fourier coefficients fulfill the following conditions:

$$\left|\hat{f}(S)\right| = c \qquad \forall S \subseteq rel(f), S \neq \{\emptyset\},$$
$$\left|\hat{f}(\emptyset)\right| = 1 - c$$

with

$$c = 2^{-(k-1)}.$$

It can be seen that each of these functions is completely described by A and b, as the permutation π does not have any impact. Hence, there are 2^{k+1} such functions. Further, it becomes clear from Corollary 2.3 that the zero coefficient of such a NCF, assuming uniform distribution, fulfills the upper bound in Proposition 2.8 with equality, i.e., the absolute value of a NCF, whose variables are all most canalizing, is maximized. Further, the sign of the zero coefficient is given by b.

NCFs with alternating canalized values (NCF-ACV) Another important subclass of NCFs are the functions with alternating b_i, i.e., where $b_i = -b_{i-1}$ and $b_1 \in \Omega$. The following corollary to Proposition 2.7 gives an exact value for the zero order coefficient of such functions.

Corollary 2.4

The absolute value of the zero coefficient of a NCF-ACV f with k relevant and uniformly distributed variables and alternating b_i, i.e., with $b_i = -b_{i-1}$ and $b_1 \in \Omega$, is given as:

$$|\hat{f}(\emptyset)| = \frac{1}{3}\left(\frac{1}{2^{k-1}}(-1)^k + 1\right).$$

NCFs with second order most dominant variables (NCF-SMD) This subclass is quite similar to the NCF-MDV. A function f is called NCF-SMD, if it is canalizing in x_{π_1}, i.e., $f^{(\pi_1,a_1)} = b_1 \in \Omega$ and $f^{(\pi_1,\bar{a}_1)}$ is a NCF-MDV with $b = -b_1$. Let's look at the zero coefficient of these functions:

Corollary 2.5

Let f be a NCF-SMD f with k relevant and uniformly distributed variables, then the absolute value of the zero coefficient is given as:

$$\hat{f}(\emptyset) = \frac{1}{2^{k-1}},$$

and hence fulfills the lower bound in Proposition 2.8 with equality.

Proof: Starting from Proposition 2.7 we know:

$$\left|\hat{f}(\emptyset)\right| = \left|\frac{1}{2}\hat{f}^{(\pi_1,\bar{a}_1)}(\emptyset) + \frac{1}{2}b_1\right|.$$

From Corollary 2.3 we know that $|\hat{f}^{(\pi_1,\bar{a}_1)}(\emptyset)| = 1 - \frac{1}{2^{k-2}}$, further $\operatorname{sgn}\left(\hat{f}^{(\pi_1,\bar{a}_1)}(\emptyset)\right) = -b_1$, hence:

$$\begin{aligned}\left|\hat{f}(\emptyset)\right| &= \left|\frac{1}{2}b_1\frac{1}{2^{k-2}}\right| \\ &= \frac{1}{2^{k-1}}.\end{aligned}$$

■

2.3.3. Unate Functions

Unate Boolean functions (UBFs), i.e., local monotone BFs, have been investigated in the field of computer science since at least the early 60s [McN61]. However, their importance for biological applications has been recognized later. As mentioned in [GKK06] a regulator can only be either activating or repressing, i.e., its influence is monotone, hence BFs, which are used to model regulatory relations, are always unate [GKK06].

To define unate functions we first need to take a look at monotone functions, as UBFs are a simple extension of this class:

Definition 2.7

A BF $f : \Omega^n \to \Omega$ is called monotone, if for each $i \in \{1, \ldots, n\}$ it holds that

$$f(x_1, \ldots, x_i = -1, \ldots, x_n) \leq f(x_1, \ldots, x_i = 1, \ldots, x_n).$$

Now, UBFs can be defined as follows:

Definition 2.8

A BF f is unate, if there exists a vector $(a_1, a_2, \ldots, a_n) \in \Omega^n$ such that the function $f(a_1 \cdot x_1, \ldots, a_n \cdot x_n)$ is monotone.

Hence, a BF is unate, if it is monotone (either increasing or decreasing) in each of its variables. The class of UBFs is closed under restrictions, since every restriction of a locally monotone function yields again in a locally monotone function. To test whether a function is unate or not, it is sufficient to use the definition, however, a necessary condition in the Fourier domain for a function to be unate is given by the following proposition:

Proposition 2.10 (*e.g., [BT96]*)

If f is a unate function, then for each relevant variable i

$$\hat{f}(\{i\}) \neq 0.$$

As mentioned before, the NCFs form a subclass of the unate functions.

Chapter 3

Noise Sensitivity and Average Sensitivity

The noise sensitivity and the average sensitivity are two measures using different channel models to quantify the impact of disturbances or errors at the functions input on the output. The average sensitivity expresses the effect of randomly flipping one input variable, while the noise sensitivity measures the error probability taking binary erasure channels into account. The impact of a single input, i.e., the *influence* of a variable, was introduced in [BOL85] and applied to BF in [KKL88]. The authors of the latter paper also showed that the sum of all influences equals the *average sensitivity*. Further, they investigated the influence intensively using Fourier analysis in the uniform case and a direct relation between this entity and the Fourier coefficients was derived. The *sensitivity*, i.e., the error tolerance of a BF at a certain input, was first studied under the term *critical complexity* (e.g., [CDR86, Weg87]) in the context of parallel random access memories (PRAM). In [Sim83] an upper bound for the maximum sensitivity was presented and further improved for symmetric functions in [Ber98].

In [KKL88] the authors investigated monotone functions, which were also studied in [Shm05], further an upper bound for locally monotone functions was presented in [Hec10]. In [LAM+13] an upper bound for the average sensitivity of NCFs was conjectured. Later in this chapter we will prove this bound. As mentioned earlier, the expectation of the average sensitivity is also an order parameter of random Boolean networks [SB07].

As stated in [O'D03], the noise sensitivity of BFs was first studied by the authors of [KKL88] and expressed in terms of Fourier coefficients, although they did not use the term "noise sensitivity" nor did their definition match the one we will introduce later. In [BKS99] the noise sensitivity of BFs was studied with focus on asymptotically noise stable behavior and the weighted majority functions. Further, it was shown that in learning theory BFs, which have low noise sensitivity, are more efficiently learnable [O'D03].

3.1. Definitions and Examples

Before we will introduce and define noise sensitivity and average sensitivity we need to look at the *influence* of variables and the *sensitivity* of input vectors.

3.1.1. Influence

As discussed in the previous chapter (see Section 2.2.1) not all variables are relevant. However, within the set of relevant variables there exist variables that have more impact on the functions output than others. Let us illustrate this in the following example:

Example 3.1

Let us consider the $(NOTx_1)AND(x_2ORx_3)$ function given by the following truth-table:

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
$+1$	$+1$	$+1$	$+1$
$+1$	$+1$	-1	-1
$+1$	-1	$+1$	-1
$+1$	-1	-1	-1
-1	$+1$	$+1$	$+1$
-1	$+1$	-1	$+1$
-1	-1	$+1$	$+1$
-1	-1	-1	$+1$

If we assume uniformly distributed input variables, each row, i.e., each input vector $\mathbf{x} = \{x_1, x_2, x_3\}$ is equal probable. Hence, one can see that x_1 almost always has an impact on the functions output (except if $x_2 = x_3 = +1$). Here, by impact we mean that the output changes if the input variable is flipped. As this happens in 6 out of 8 rows, it occurs with probability $\frac{3}{4}$. Thus, we say x_1 has an influence of $I_1(f) = \frac{3}{4}$. Let us now investigate x_2. It has only an impact if $x_1 = +1$ and $x_3 = -1$, i.e., in 2 cases. Hence, the influence of x_2 is $I_2(f) = \frac{1}{4}$. Finally, one can also see that $I_3(f) = \frac{1}{4}$.

As seen in this example we define the influence of a variable as the probability that the functions output changes if this variable is flipped. The notion of influence has been introduced by the authors of [BOL85] and was first applied to BFs in [KKL88]. Formally we can write:

Definition 3.1 (*[HM91]*)

The influence of a variable x_i on the function f is defined as:

$$I_i = Pr\left[f(\mathbf{x}) \neq f(\mathbf{x} \oplus e_i)\right],$$

where as before $x \oplus e_i$ is the vector obtained from $\mathbf{x}$ by flipping its i-th entry.

The following proposition establishes the relationship between the influence and the Fourier coefficients:

Proposition 3.1 (*e.g., [BT96]*)

Given any BF f with product distributed inputs and let $\hat{f}(S), S \subseteq [n]$ be its Fourier coefficients, then the influence of variable x_i is given by

$$I_i(f) = \sum_{S:i \in S} \frac{1}{\sigma_i^2} \hat{f}^2(S),$$

where σ_i is the standard deviation of input i.

Let us illustrate this proposition by continuing the previous example:

Example 3.2

The Fourier spectrum of the function from Example 3.1 is given by

S	$\emptyset$	$\{1\}$	$\{2\}$	$\{1,2\}$	$\{3\}$	$\{1,3\}$	$\{2,3\}$	$\{1,2,3\}$
$\hat{f}(S)$	$\frac{1}{4}$	$-\frac{3}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$

Applying Proposition 3.1 we get for the influences:

$$I_1(f) = \left(-\frac{3}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2 = \frac{12}{36} = \frac{3}{4}$$
$$I_2(f) = \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2 = \frac{4}{36} = \frac{1}{4}$$
$$I_3(f) = \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2 = \frac{4}{36} = \frac{1}{4}$$

Please note that we still assume uniform distribution, hence, $\sigma_i = 1, \forall i \in [n]$.

The influence $I_i(f)$ for a unate function f is directly related to the corresponding Fourier coefficient:

$$I_i(f) = \frac{|\hat{f}(\{i\})|}{\sigma_i}, \tag{3.1}$$

as it was shown for monotone functions in [BT96, Lemma 4.5] and can be easily extended to unate functions.

The measure of influence can also be used to determine relevant variables of a BF (see Section 2.2.1). One can see that a variable is relevant, if and only if its influence is larger than zero.

3.1.2. Sensitivity

While the influence gives the probability that the output changes, if one input is flipped, the sensitivity of a BF on a certain input vector $\mathbf{x}$ gives the number of variables for which the output changes, if one of these variables is flipped. More formally we can write:

Definition 3.2 (*e.g. [Ber98]*)

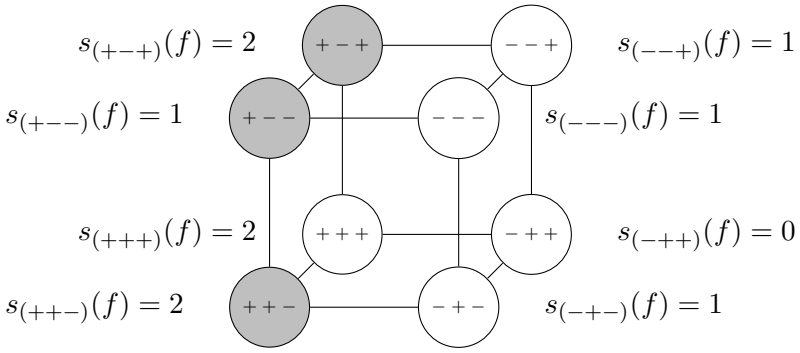
The sensitivity $s_{\mathbf{x}}(f)$ *of a BF* f *on a certain input vector* $\mathbf{x}$ *is defined as the number of variables* $x_i, i \in [n]$ *for which* $f(\mathbf{x}) \neq f(\mathbf{x} \oplus e_i)$, *i.e.,*

$$s_{\mathbf{x}}(f) = \frac{1}{4} \sum_{i \in [n]} \left(f(\mathbf{x}) - f(\mathbf{x} \oplus e_i)\right)^2 .$$

The following example illustrates this concept graphically:

Example 3.3

Let us draw the BF from Examples 3.1 and 3.2 as a 3-dimensional graph, where each dimension represents one input variable. Each node represents one input vector and is labeled accordingly, e.g., $+-+$ means, $x_1 = +1$, $x_2 = -1$ and $x_3 = +1$.



A shaded node means that the functions output is -1, otherwise the output is $+1$. The sensitivity of a node, i.e., $s_{\mathbf{x}}(f)$ is the number of neighbors with a different color and can be found in the graph.

If we now take the average of all sensitivities, we get the *average sensitivity*, which is in this case (assuming uniform distribution) $\mathtt{as}(f) = \frac{5}{4}$. We will formally introduce the average sensitivity in the next subsection.

3.1.3. Average Sensitivity

As mentioned before the average sensitivity (as) gives the influence of random disturbances at the input on the output of a BF. This can be interpreted as an indicator for the robustness of this BF.

The average sensitivity $\mathtt{as}(f)$ is the expected value over all arguments $\mathbf{x}$:

$$\mathtt{as}(f) = \mathbf{E}\left[s_{\mathbf{x}}(f)\right].$$

Hence, the as depends on the distribution of the input vector. For example a function having a low average sensitivity for the uniform distribution may have a large average sensitivity for other distributions. In general $\mathtt{as}(f)$ can be as large as the number of relevant variables k, i.e.,

$$0 \leq \mathtt{as}(f) \leq k.$$

Alternatively, the as can be defined using the influence. Referring to [KKL88], it can be shown that,

$$\mathtt{as}(f) = \mathbf{E}\left[s_{\mathbf{x}}(f)\right] = \sum_{i \in [n]} I_i(f).$$

Consequently the average sensitivity can also be expressed in terms of the Fourier coefficients [Fri98] as:

$$\mathtt{as}(f) = \sum_{S \subseteq [n], S \neq \emptyset} \hat{f}(S)^2 \sum_{i \in S} \frac{1}{\sigma_i^2}.$$

In the uniform case this equation reduces to

$$\mathtt{as}(f) = \sum_{S\subseteq[n],S\neq\emptyset} \hat{f}(S)^2|S|.$$

From its definition it becomes obvious that in general the as is upper-bounded by the number of relevant variables k. Further from Poincare's inequality a general lower bound follows [KKL88], hence, we can state:

$$\mathbf{Var}\left[f\right] \leq \mathtt{as}(f) \leq k. \tag{3.2}$$

For unate functions we can write using Eq. (3.1):

$$\mathtt{as}(f) = \sum_{i=1}^{n} \frac{|\hat{f}(\{i\})|}{\sigma_i},$$

and using the Cauchy-Schwarz inequality one can show that [Hec10]

$$\mathtt{as}(f) \leq \sqrt{\mathbf{Var}\left[f\right]}\sqrt{\sum_{i=1}^{n}\left(\frac{1}{\sigma_i}\right)^2}.$$

For uniformly distributed inputs this bound reduces to

$$\mathtt{as}(f) \leq \sqrt{n}. \tag{3.3}$$

It can be shown, that some functions get close to this upper bound. For example, it is well-known that the average sensitivity of the majority function behaves like $O(\sqrt{n})$ ([O'D08]).

3.1.4. Noise Sensitivity

While the average sensitivity only measures the impact of a flip in the input, the *noise sensitivity* (NS) also takes the probability with which this distortion may happen into account. Let us recall the noise-operator $N_\epsilon(\cdot)$ from Definition 2.1 on page 7, then the NS is the probability that $f(\mathbf{X})$ differs from $f(N_\epsilon(\mathbf{X}))$:

Definition 3.3 (*[O'D03]*)

Let $f : \Omega^n \to \Omega$ be a BF and let $\mathbf{X}$ be a product distributed random variable and $\mathbf{X}' = N_\epsilon(\mathbf{X})$ its noise copy. The noise sensitivity (NS) is defined as:

$$\mathtt{NS}_\epsilon(f) = Pr\left[f(\mathbf{X}) \neq f(\mathbf{X}')\right].$$

Note, that in general the NS can be defined for any kind of distortion. In this work, however, we focus on the noise introduced by n parallel BSCs and the corresponding noise-operator N_ϵ. Similar to the notion of sensitivity we can define a conditional noise sensitivity (CNS) as the NS of a BF with fixed inputs $\mathbf{x}$:

Definition 3.4

Given a BF with a fixed input $\mathbf{X} = \mathbf{x}$ *and let* $\mathbf{X}'$ *be the corresponding noisy copy, then*

$$\mathit{NS}_{\epsilon,\mathbf{x}}(f) = Pr\left[f(\mathbf{X}) \neq f(\mathbf{X}')|\mathbf{X} = \mathbf{x}\right].$$

As the as is the expectation of the sensitivities, the NS is the expectation of the CNS:

$$\mathtt{NS}_\epsilon(f) = \sum_{\mathbf{x}} \Pr\left[\mathbf{X} = \mathbf{x}\right] \mathtt{NS}_{\epsilon,\mathbf{x}}(f). \tag{3.4}$$

For any Boolean function f the noise sensitivity with respect to the uniform distribution can be expressed in terms of Fourier coefficients as [O'D03]:

$$\mathtt{NS}_\epsilon(f) = \frac{1}{2}\left(1 - \sum_U \hat{f}^2(U)\rho^{|U|}\right), \tag{3.5}$$

where

$$\rho = 1 - 2\epsilon.$$

By shifting the energy of all Fourier coefficients except $\hat{f}(\emptyset)$ to the highest order $|U| = k$, we can easily give an upper bound on the noise sensitivity for a fixed zero coefficient:

$$\mathtt{NS}_\epsilon(f) \leq \frac{1}{2}\left(1 - \hat{f}^2(\emptyset) - (1 - \hat{f}^2(\emptyset))\rho^k\right). \tag{3.6}$$

Likewise, by shifting all energy to the first order coefficient we get the following lower bound:

$$\mathtt{NS}_\epsilon(f) \geq \frac{1}{2}\left(1 - \hat{f}^2(\emptyset) - (1 - \hat{f}^2(\emptyset))\rho\right). \tag{3.7}$$

3.1.5. Relation between Average Sensitivity and Noise Sensitivity

We mentioned before that the noise sensitivity and average sensitivity are closely related. The following proposition formally describes this fact:

Proposition 3.2 (*Uniform case [O'D03], Product case e.g., [Hec10]*)

Let f *be a BF with product distributed input values and* $0 \leq \epsilon \leq \frac{1}{2}$ *the error probability as defined in Definition 2.1, then*

$$\left.\frac{d\mathit{NS}_\epsilon(f)}{d\epsilon}\right|_{\epsilon=0} = \mathtt{as}(f).$$

With other words, the as is the slope of the NS at $\epsilon = 0$. From that one can derive the following commonly known bound for uniform distributed variables (e.g., [O'D03]):

$$\mathtt{NS}_\epsilon(f) \leq \mathtt{as}(f) \cdot \epsilon. \tag{3.8}$$

Let us now look at some examples and calculate the as and NS for some functions:

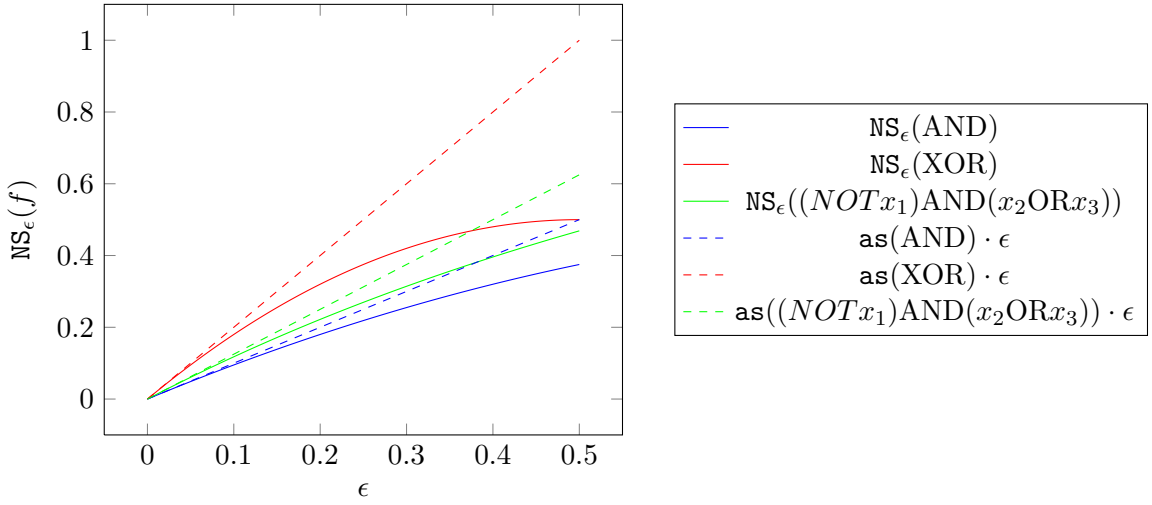


Figure 3.1.: Plot of Noise Sensitivity and Average Sensitivity for some exemplary Boolean functions.

Example 3.4

The following table illustrates the as *and the NS of the AND, OR, XOR and the* $(NOTx_1)AND(x_2ORx_3)$ *(see Example 3.1) functions in the uniform case:*

f	as(f)	NS$_\epsilon(f)$
x_1ANDx_2	1	$\frac{1}{2}\left(\frac{3}{4}-\frac{1}{2}\rho-\frac{1}{4}\rho^2\right)$
x_1ORx_2	1	$\frac{1}{2}\left(\frac{3}{4}-\frac{1}{2}\rho-\frac{1}{4}\rho^2\right)$
x_1XORx_2	2	$\frac{1}{2}\left(1-\rho^2\right)$
$(NOTx_1)AND(x_2ORx_3)$	$\frac{5}{4}$	$\frac{1}{2}\left(\frac{15}{16}-\frac{11}{16}\rho-\frac{3}{16}\rho^2-\frac{1}{16}\rho^3\right)$

Please remember that $\rho = 1-2\epsilon$*. We can see that the NS and consequently the* as *of the AND and OR functions are equal, since their Fourier coefficients have the same absolute value. Further, in Figure 3.1 we plotted the NS and the corresponding bound of Eq. (3.8).*

As one can see from the example and from Eq. (3.5) the NS is lower for more energy being concentrated on the Fourier coefficients with low order. It is minimized for the two constant functions, i.e., $f(\mathbf{x}) = +1$ and $f(\mathbf{x}) = -1$. On the other hand, functions with energy concentrated on the high order coefficients have a large NS, hence, it is maximized by the XOR functions.

However, finding non-trivial functions minimizing the NS is not easy, as not every set of Fourier coefficients is a valid BF. In the next section we will discuss the NS of canalizing and nested canalizing functions, as they show somehow an optimal behavior.

3.2. Noise Sensitivity for Different Classes of Boolean Functions

In this section we will investigate the noise sensitivity, mainly for CFs and NCFs. In the next subsection we will summerize our findings and discuss them in Subsection 3.2.2. After that we will state the proofs for the different results in the remaining subsections.

3.2.1. Main Results of this Section

In Eq. (3.8) we stated a general upper bound for the NS. However, it is often desired to compare only BFs with the same bias. The following proposition gives an upper and a lower bound for a fixed bias.

Corollary 3.1

Let f be a BF with k uniformly distributed and relevant input variables with a fixed zero coefficient $\hat{f}(\emptyset)$, then the NS is bounded with:

$$\frac{1}{2}\left(1-\hat{f}^2(\emptyset)-(1-\hat{f}^2(\emptyset))\cdot(1-2\epsilon)\right) \leq \mathtt{NS}_\epsilon(f) \leq \frac{1}{2}\left(1-\hat{f}^2(\emptyset)-(1-\hat{f}^2(\emptyset))\cdot(1-2\epsilon)^k\right).$$

The upper bound is tight.

Proof: The result follows from Proposition 3.3 on page 37. ■

Let us now obverse CFs. From Proposition 3.4 on page 39 it follows:

Corollary 3.2

Let f be a CF with the zero order coefficient $\hat{f}(\emptyset)$, then the NS is upper bounded as:

$$\mathtt{NS}_\epsilon(f) \leq \frac{1}{2}\left(1-\left(1-2|\hat{f}(\emptyset)|+2\hat{f}(\emptyset)^2\right)(1-2\epsilon)\right).$$

This bound can further be made independent of $\hat{f}(\emptyset)$ by applying the fact that

$$\left(1-2|\hat{f}(\emptyset)|+2\hat{f}(\emptyset)^2\right) \geq \frac{1}{2},$$

which leads us to:

Corollary 3.3

Let f be a CF, then the NS is upper bounded as:

$$\mathtt{NS}_\epsilon(f) \leq \frac{1}{4}+\frac{1}{2}\epsilon.$$

For NCFs we can again obtain bounds with or without taking the bias into account. The following corollary states an upper bound for NCFs only depended on the number of relevant variables and ϵ:

Corollary 3.4

Let f be a NCF with $k = rel(f)$ relevant variables, then the NS is upper bounded as:

$$\mathtt{NS}_\epsilon(f) \leq \frac{4\epsilon-2^{-k}\cdot(1-\epsilon)^{k-2}\cdot\left(3-4\epsilon+7\epsilon^2-2\epsilon^3+(-1)^k(1+\epsilon)^2\right)\cdot\epsilon}{3+2\epsilon-\epsilon^2}.$$

Proof: The result directly follows from Theorem 3.2 on page 40. ■

This result can be further bounded by:

$$\mathtt{NS}_\epsilon(f) \leq \frac{4\epsilon}{3+2\epsilon-\epsilon^2}. \tag{3.9}$$

As one can see, these bounds are only dependent on ϵ and k. However, as mentioned before, sometimes it is desired to only compare functions with the same bias, i.e., the same $\hat{f}(\emptyset)$. Therefore, from Proposition 3.6 on page 42 we can derive the following additional bound taking the bias into account:

Corollary 3.5

Let f be a $\{\pi : A : B\}$-NCF with a fixed bias of $\hat{f}(\emptyset)$, then the NS is bounded as

$$\mathtt{NS}_\epsilon(f) \leq \epsilon \frac{4+(1-\epsilon)^2-\left(4-(1-\epsilon)^2\right)\cdot|\hat{f}(\emptyset)|)}{(3-\epsilon)(1-\epsilon^2)}.$$

Finally, let us investigate the previously introduced subclasses of NCFs. The following three corollaries state the corresponding results:

Corollary 3.6

Let f be a $\{\pi : A : B\}$-NCF-MDV with k uniformly distributed relevant input variables, then

$$\mathtt{NS}_\epsilon = 2^{-k+1}\left(1-(1-\epsilon)^k\right).$$

Proof: The result directly follows from Proposition 3.7 on page 42. ■

If the number of relevant variables increases, these functions become more and more biased, i.e., $|\hat{f}(\emptyset)| \to 1$, the NS tend to zero:

$$\begin{matrix} r_{\epsilon,c}(f) = 0 \\ \mathtt{NS}_\epsilon(f) = 0 \end{matrix} \quad \text{for } k \to \infty.$$

Corollary 3.7

Let f be a $\{\pi : A : B\}$-NCF-ACV with k uniformly distributed relevant input variables, then

$$\mathtt{NS}_\epsilon(f) = \frac{4\epsilon}{3(1+\epsilon)} - \frac{2\epsilon(-2)^{-k}}{3(2-\epsilon)} + \frac{2^{-k+1}\epsilon(1-\epsilon)^{k+1}}{2+\epsilon-\epsilon^2}.$$

Proof: The result directly follows from Proposition 3.8 on page 43. ■

If we now let k grow to infinity, i.e., if we observe a large number of relevant variables, we obtain the following corollary:

Corollary 3.8

Let f be a $\{\pi : A : B\}$-NCF-ACF, then for an infinite number of relevant variables we can write

$$\mathtt{NS}_\epsilon(f) = \frac{4\epsilon}{3(1+\epsilon)}.$$

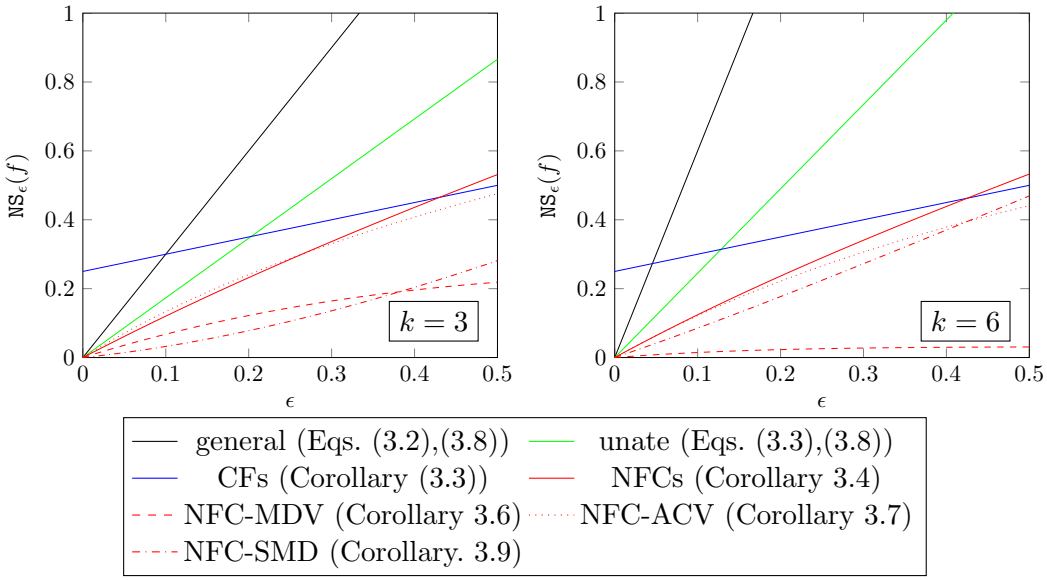


Figure 3.2.: Upper bounds on the NS for different classes of functions, with $k = 3$ and $k = 6$ relevant variables, respectively. (The bound for CFs is independend of k.)

Further, it is worth to note that in this case the bias tends to $\frac{1}{3}$.

Corollary 3.9

Let f be a $\{\pi : A : B\}$-NCF-ACV with k uniformly distributed relevant input variables, then

$$\mathtt{NS}_\epsilon(f) = \epsilon - 2^{-k+1} + 2^{-k+1}(1-\epsilon)^k.$$

Proof: The result directly follows from Proposition 3.9 on page 44. ■

Again, if we investigate these functions for an infinite number of relevant inputs, we obtain the following corollary. Further, these function then become balanced.

Corollary 3.10

Let f be a $\{\pi : A : B\}$-NCF-SMD, then for an infinite number of relevant variables we can write

$$\mathtt{NS}_\epsilon(f) = \epsilon.$$

3.2.2. Discussion and Implications

In Figure 3.2 the upper bounds of the NS versus the error probability ϵ for all BFs, unate BFs, CFs, and NCFs have been plotted for $k = 3$ and $k = 6$, respectively. We can see the general upper bound for all BFs, which is the combination of Eqs. (3.2) and (3.8), i.e., $\mathtt{NS}_\epsilon(f) \leq k \cdot \epsilon$ is highest, followed by the bound for unate BFs, $\mathtt{NS}_\epsilon(f) \leq \sqrt{k} \cdot \epsilon$ (see Eqs. (3.3) and (3.8)). These bounds both increase strongly with k. The bounds of the CFs and NCFs,

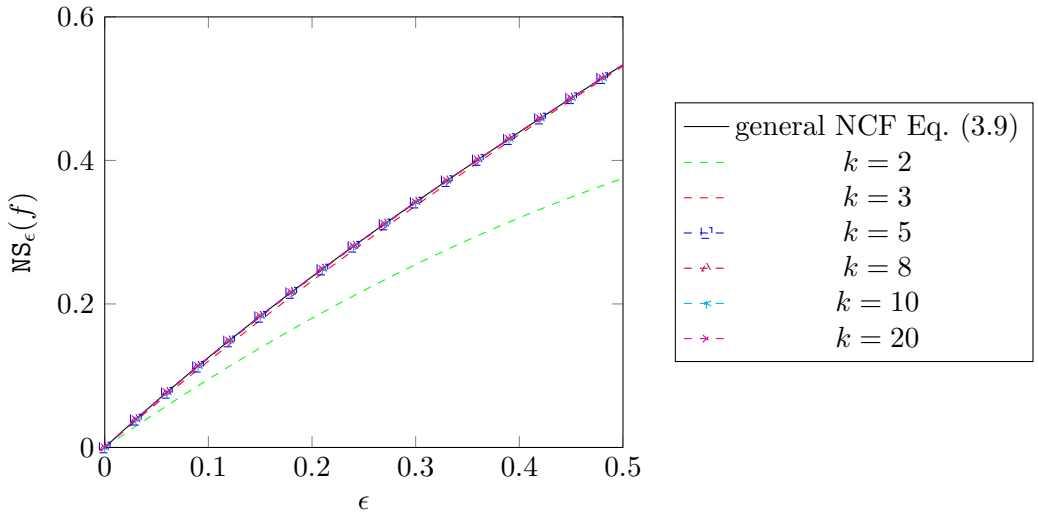


Figure 3.3.: Upper bounds on the NS for NCFs with different numbers of relevant variables k.

however, remain lower. In fact, the bound for CFs is independent of k, while the NCFs bound only changes a little with k. This is further investigated in Figure 3.3, where the upper bound for NCFs derived in Corollary 3.4 for different numbers of relevant variables k are depicted and compared to the general bound for NCFs (Eq. 3.9). One can see that all curves – except for $k = 2$ – are relatively close to the general bound, in fact, they tend towards this bound for increasing k. Hence, it is sufficient to use only this general bound. Further, for comparison we added the NSs for NCF-MDVs, NCF-ACVs and NCF-SMDs to the graphs of Figure 3.2.

In Figure 3.4 we plotted the NS of all BFs with $k = 3$ and $k = 4$ relevant variables against their bias for $\epsilon = 0.1$. The CFs and NCFs are marked explicitly. Additionally, we added the bounds derived above. One can see, as indicated by the bounds, that the NCFs have optimal, i.e., lowest, NS and CFs also perform quite well. From this we conjecture a optimality of NCFs with respect to the NS.

Finally, we can state that in opposition to BFs in general and unate BFs, the NS of canalizing and nested canalizing functions does not become larger than a certain bound, which is independent of the number of relevant variables. Hence, they are more robust to random changes of the inputs.

3.2.3. General Bounds

We will derive our results in the remainder of this section for a generalization of the NS, as we will need this later to derive some bounds for the mutual information. Let us define a function $r_{\epsilon,c}(f)$ as follows:

$$r_{\epsilon,c}(f) = \frac{1}{2}(1 - \sum_{U} \hat{f}(U)^2(1-2\epsilon)^{c|U|}), \tag{3.10}$$

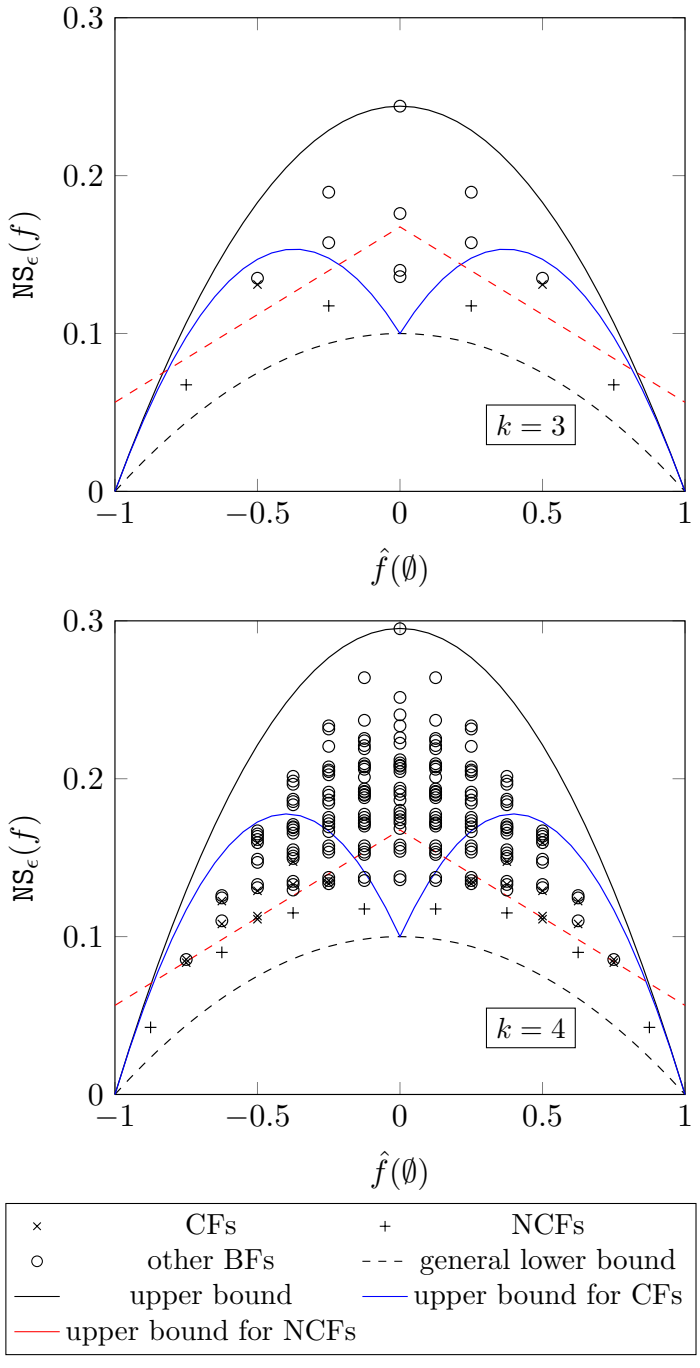


Figure 3.4.: NS of all BFs with $k = 3$ and $k = 4$ relevant variables, respectively, and $\epsilon = 0.1$ compared to some bounds.

where $c \in \mathbb{R}^+$ is some constant and $0 \leq \epsilon \leq \frac{1}{2}$. Obviously,

$$\mathtt{NS}_\epsilon(f) = r_{\epsilon,1}(f).$$

The following proposition gives an upper and a lower bound on $r_{\epsilon,c}(f)$ for a fixed bias.

Proposition 3.3

Let f be a BF with k uniformly distributed and relevant input variables with a fixed zero coefficient $\hat{f}(\emptyset)$, then $r_{\epsilon,c}$ is bounded with:

$$\frac{1}{2}\left(1 - \hat{f}^2(\emptyset) - (1 - \hat{f}^2(\emptyset)) \cdot \rho^c\right) \leq r_{\epsilon,c}(f) \leq \frac{1}{2}\left(1 - \hat{f}^2(\emptyset) - (1 - \hat{f}^2(\emptyset)) \cdot \rho^{c \cdot k}\right).$$

The upper bound is tight.

Proof: The proof is analog to Eqs. (3.6) and (3.7). The XOR function fulfills the upper bound. ■

3.2.4. Noise Sensitivity of Restricted Functions

To investigate CFs and NCFs we first need to look at the effect of restricted BFs. As seen before, such restricted functions play an important role when describing CFs and NCFs. First, let us define the function $\xi : \mathcal{F}_n \times \mathcal{F}_n \to \mathbb{R}$ by

$$\xi_{\epsilon,c}(f,g) = \frac{1}{2}\left(1 - \sum_{U \subseteq [n]} \hat{f}(U)\hat{g}(U)(1-2\epsilon)^{c|U|}\right), \tag{3.11}$$

where f and g are two BFs.

If we now apply the concept of restricted functions (see Subsection 2.2.3) to the definition of $r_{\epsilon,c}$, we obtain the following theorem for the uniform case:

Theorem 3.1

Let $f^{(i,+)}, f^{(i,-)}$ be the restrictions of f to some relevant variable i of f. Then

$$r_{\epsilon,c}(f) = \lambda^{(1)}_{\epsilon,c} \cdot r_{\epsilon,c}(f^{(i,+)}) + \lambda^{(1)}_{\epsilon,c} \cdot r_{\epsilon,c}(f^{(i,-)}) + \lambda^{(2)}_{\epsilon,c} \cdot \xi_\epsilon(f^{(i,+)}, f^{(i,-)}), \tag{3.12}$$

where

$$\lambda^{(1)}_{\epsilon,c} = \frac{1}{4}\left(1 + (1-2\epsilon)^c\right)$$
$$\lambda^{(2)}_{\epsilon,c} = \frac{1}{2}\left(1 - (1-2\epsilon)^c\right).$$

Proof: Starting from the definition of $r_{\epsilon,c}(f)$, we can partition the Fourier coefficients

according to Proposition 2.2. This yields into:

$$\begin{aligned}
r_{\epsilon,c}(f) =& \frac{1}{2}\left(1 - \sum_U \left(\frac{1}{2}\left(\hat{f}^{(i,+)}(U\setminus\{i\}) + (-1)^{|U\cap\{i\}|}\cdot \hat{f}^{(i,-)}(U\setminus\{i\})\right)\right)^2 (1-2\epsilon)^{c|U|}\right) \\
=& \frac{1}{2}\Bigg(1 - \frac{1}{4}\sum_U \Big(\hat{f}^{(i,+)^2}(U\setminus\{i\}) + \underbrace{\left((-1)^{|U\cap\{i\}|}\right)^2}_{=1}\cdot \hat{f}^{(i,-)^2}(U\setminus\{i\}) \\
& + 2\hat{f}^{(i,+)}(U\setminus\{i\})(-1)^{|U\cap\{i\}|}\hat{f}^{(i,-)}(U\setminus\{i\})\Big)\cdot(1-2\epsilon)^{c|U|}\Bigg),
\end{aligned}$$

which leads us to

$$\begin{aligned}
r_{\epsilon,c}(f) = \frac{1}{2}\Bigg(1 &- \frac{1}{4}\sum_U \hat{f}^{(i,+)^2}(U\setminus\{i\})\cdot(1-2\epsilon)^{c|U|} - \frac{1}{4}\sum_U \hat{f}^{(i,-)^2}(U\setminus\{i\})\cdot(1-2\epsilon)^{c|U|} \\
&- \frac{1}{2}\sum_U \hat{f}^{(i,+)}(U\setminus\{i\})\cdot(-1)^{|U\cap\{i\}|}\cdot\hat{f}^{(i,-)}(U\setminus\{i\})\cdot(1-2\epsilon)^{c|U|}\Bigg).
\end{aligned}$$

Since $f^{(i,a)}(U) = 0$ for all $U : i \in U$, we can write

$$\begin{aligned}
r_{\epsilon,c}(f) = \frac{1}{2}\Bigg(1 &- \frac{1}{4}\sum_U \hat{f}^{(i,+)^2}(U)\cdot(1-2\epsilon)^{c|U|} - \frac{1}{4}\sum_U \hat{f}^{(i,+)^2}(U)\cdot(1-2\epsilon)^{c|U|+c} \\
&- \frac{1}{4}\sum_U \hat{f}^{(i,-)^2}(U)\cdot(1-2\epsilon)^{c|U|} - \frac{1}{4}\sum_U \hat{f}^{(i,-)^2}(U)\cdot(1-2\epsilon)^{c|U|+c} \\
&- \frac{1}{2}\sum_U \hat{f}^{(i,+)}(U)\cdot\hat{f}^{(i,-)}(U)\cdot(1-2\epsilon)^{c|U|} \\
&+ \frac{1}{2}\sum_U \hat{f}^{(i,+)}(U)\cdot\hat{f}^{(i,-)}(U)\cdot(1-2\epsilon)^{c|U|+c}\Bigg),
\end{aligned}$$

$$\begin{aligned}
r_{\epsilon,c}(f) = \frac{1}{2}\Bigg(1 &- \underbrace{\frac{1}{4}\left(1+(1-2\epsilon)^c\right)}_{=\lambda^{(1)}_{\epsilon,c}}\sum_U \hat{f}^{(i,+)^2}(U)\cdot(1-2\epsilon)^{c|U|} \\
&- \underbrace{\frac{1}{4}\left(1+(1-2\epsilon)^c\right)}_{=\lambda^{(1)}_{\epsilon,c}}\sum_U \hat{f}^{(i,-)^2}(U)\cdot(1-2\epsilon)^{c|U|} \\
&- \underbrace{\frac{1}{2}\left(1-(1-2\epsilon)^c\right)}_{=\lambda^{(2)}_{\epsilon,c}}\sum_U \hat{f}^{(i,+)}(U)\cdot\hat{f}^{(i,-)}(U)\cdot(1-2\epsilon)^{c|U|}\Bigg).
\end{aligned}$$

Since $1 = 2\lambda^{(1)}_{\epsilon,c} + \lambda^{(2)}_{\epsilon,c}$, we can write:

$$\begin{aligned}
r_{\epsilon,c}(f) =& \lambda^{(1)}_{\epsilon,c}\cdot\frac{1}{2}(1 - \sum_U \hat{f}^{(i,+)^2}(U)\cdot(1-2\epsilon)^{c|U|}) + \lambda^{(1)}_{\epsilon,c}\cdot\frac{1}{2}(1 - \sum_U \hat{f}^{(i,-)^2}(U)\cdot(1-2\epsilon)^{c|U|}) \\
&+ \lambda^{(2)}_{\epsilon,c}\cdot\frac{1}{2}(1 - \sum_U \hat{f}^{(i,+)}(U)\cdot\hat{f}^{(i,-)}(U)\cdot(1-2\epsilon)^{c|U|}),
\end{aligned}$$

and by applying the definition of $r_{\epsilon,c}$ we finally get:

$$r_{\epsilon,c}(f) = \lambda^{(1)}_{\epsilon,c} \cdot r_{\epsilon,c}(f^{(i,+)}) + \lambda^{(1)}_{\epsilon,c} \cdot r_{\epsilon,c}(f^{(i,-)}) + \lambda^{(2)}_{\epsilon,c} \cdot \xi_{\epsilon,c}(f^{(i,+)}, f^{(i,-)}).$$

■

This theorem shows that the NS of any BF f is basically the sum of the NSs of its two restrictions plus a correlation term. This is an important property, which will be used in the remainder of this chapter.

3.2.5. Canalizing Functions

Based on our findings so far, we can directly state and prove the following proposition, giving an upper bound on the NS, respectively $r_{\epsilon_c}(f)$, for a fixed bias.

Proposition 3.4

Let f be a Boolean function with k uniformly distributed input variables and a zero coefficient of $f(\emptyset)$, which is canalizing in some variable j, i.e. $\hat{f}(\emptyset) + a_j\hat{f}(j) = b_j$, then $r_{\epsilon,c}$ is upper-bounded with:

$$r_{\epsilon,c}(f) \leq \frac{1}{2}\left(1 - \hat{f}^2(\emptyset) - \left(1 - |\hat{f}(\emptyset)|\right)^2 \cdot (1-2\epsilon)^c - \left(2|\hat{f}(\emptyset)| - 2\hat{f}^2(\emptyset)\right) \cdot (1-2\epsilon)^{c\cdot k}\right) \tag{3.13}$$

$$\leq \frac{1}{2}\left(1 - \left(1 - 2|\hat{f}(\emptyset)| + 2\hat{f}^2(\emptyset)\right) \cdot (1-2\epsilon)^c\right) \tag{3.14}$$

Proof: Since $\hat{f}(\{j\}) = a_j(b_j - \hat{f}(\emptyset))$ (see Theorem 2.2) and by applying Eq. (2.10), i.e., $b_j = \operatorname{sgn}\left(\hat{f}(\emptyset)\right)$, we can write:

$$\begin{aligned}\hat{f}(\{j\})^2 = (b_j - \hat{f}^2(\emptyset)) &= 1 - 2b_j\hat{f}(\emptyset) + \hat{f}^2(\emptyset) \\ &= 1 - 2|\hat{f}(\emptyset)| + \hat{f}(\emptyset)^2.\end{aligned}$$

Starting from Eq. (3.10) we get:

$$r_{\epsilon,c}(f) = \frac{1}{2}\left(1 - \hat{f}^2(\emptyset) - \hat{f}^2(\{j\}) \cdot \rho^c - \sum_{i\in[k],i\neq j} \hat{f}^2(\{i\}) \cdot \rho^c - \sum_{U,|U|>1} \hat{f}^2(U) \cdot \rho^{c|U|}\right).$$

The energy of $\hat{f}(\emptyset)$ and $\hat{f}(\{j\})$ is fixed. Hence, we shift the remaining energy, $1 - \hat{f}^2(\emptyset) - \hat{f}^2(\{j\}$, to the highest order coefficient:

$$r_{\epsilon,c}(f) \leq \frac{1}{2}\left(1 - \hat{f}^2(\emptyset) - \hat{f}^2(\{j\}) \cdot \rho^c - \left(1 - 2|\hat{f}(\emptyset)| + \hat{f}^2(\emptyset)\right)\rho^{c\cdot k}\right),$$

from which Eq. (3.13) follows. This term can be further simplified:

$$r_{\epsilon,c}(f) \leq \frac{1}{2}\left(1 - \hat{f}^2(\emptyset) - \hat{f}^2(\{j\}) \cdot \rho^c\right),$$

which leads us to Eq. (3.14) and concludes the proof. ■

3.2.6. Nested Canalizing Functions

First, we take advantage of the recursive behavior of NCFs to derive the following proposition, which gives us a recursive description of $r_{\epsilon,c}(f)$.

Proposition 3.5

Let f be a $\{\pi : A : B\}$-NCF, then $r_{\epsilon,c}(f)$ can be recursively written as:

$$r_{\epsilon,c}(f) = \lambda_{\epsilon,c}^{(1)} \cdot r_{\epsilon,c}(f^{(\pi_1,\bar{a}_1)}) + \frac{1}{2}\lambda_{\epsilon,c}^{(2)} \cdot \left(1 - \hat{f}^{(\pi_1,\bar{a}_1)}(\emptyset) \cdot b_1\right).$$

Proof: This proposition follows directly from Proposition 3.1 by applying the spectral properties of NCFs. ■

Let us illustrate this in a small example:

Example 3.5

Let f be a BF with $k = 1$ relevant variable, i.e., the identity function or its negation, then

$$r_{\epsilon,c}(f) = \lambda_{\epsilon,c}^{(2)}.$$

Now, let f be a NCF with two relevant variables, from Proposition 3.5 follows:

$$r_{\epsilon,c}(f) = \lambda_{\epsilon,c}^{(1)} \cdot \lambda_{\epsilon,c}^{(2)} + \frac{1}{2}\lambda_{\epsilon,c}^{(2)},$$

since $f^{(\pi_1,\bar{a}_1)}$ is a BF with only one relevant variable, hence, $r_{\epsilon,c}(f^{(\pi_1,\bar{a}_1)}) = \lambda_{\epsilon,c}^{(2)}$, $b_1 = sgn\left(\hat{f}^{(\pi_1,\bar{a}_1)}(\emptyset)\right)$ and $|\hat{f}^{(\pi_1,\bar{a}_1)}(\emptyset)| = 1$.

Now, we can derive an upper bound on $r_{\epsilon,c}(f)$ for NCFs:

Theorem 3.2

Let f be a $\{\pi : A : B\}$-NCF with k relevant and uniformly distributed variables. Then

$$r_{\epsilon,c}(f) \leq \frac{\lambda_{\epsilon,c}^{(2)}}{1 - {\lambda_{\epsilon,c}^{(1)}}^2} - \frac{1 + (-1)^k(1 - \lambda_{\epsilon,c}^{(1)})^2 - 2\lambda_{\epsilon,c}^{(1)} + {\lambda_{\epsilon,c}^{(1)}}^2 + 4{\lambda_{\epsilon,c}^{(1)}}^3}{4{\lambda_{\epsilon,c}^{(1)}}^2(1 - {\lambda_{\epsilon,c}^{(1)}}^2)} {\lambda_{\epsilon,c}^{(1)}}^k \lambda_{\epsilon,c}^{(2)}, \quad (3.15)$$

or more general

$$r_{\epsilon,c}(f) \leq \frac{\lambda_{\epsilon,c}^{(2)}}{1 - {\lambda_{\epsilon,c}^{(1)}}^2}. \quad (3.16)$$

Proof: Let us recall Proposition 3.5:

$$r_{\epsilon,c}(f) = \lambda_{\epsilon,c}^{(1)} \cdot r_{\epsilon,c}(f^{(\pi_1,\bar{a}_1)}) + \frac{1}{2}\lambda_{\epsilon,c}^{(2)} \cdot \left(1 - \hat{f}^{(\pi_1,\bar{a}_1)}(\emptyset) \cdot b_1\right).$$

If we apply it again and use Proposition 2.7, we get:

$$\begin{aligned}
r_{\epsilon,c}(f) =&\lambda_{\epsilon,c}^{(1)}\left(\lambda_{\epsilon,c}^{(1)}\cdot r_{\epsilon,c}(f^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}})+\frac{\lambda_{\epsilon,c}^{(2)}}{2}\left(1-\hat{f}^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}}(\emptyset)\cdot b_2\right)\right)\\
&+\frac{\lambda_{\epsilon,c}^{(2)}}{2}\left(1-\left(\hat{f}^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}}(\emptyset)+b_2\right)\cdot\frac{b_1}{2}\right)\\
=&\lambda_{\epsilon,c}^{(1)^2}\cdot r_{\epsilon,c}(f^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}})+\frac{\lambda_{\epsilon,c}^{(1)}\cdot\lambda_{\epsilon,c}^{(2)}}{2}-\frac{b_2\cdot\lambda_{\epsilon,c}^{(1)}\cdot\lambda_{\epsilon,c}^{(2)}}{2}\hat{f}^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}}(\emptyset)\\
&+\frac{\lambda_{\epsilon,c}^{(2)}}{2}-\frac{b_1\lambda_{\epsilon,c}^{(2)}}{4}\hat{f}^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}}(\emptyset)-\frac{\lambda_{\epsilon,c}^{(2)}\cdot b_1\cdot b_2}{4}\\
=&\lambda_{\epsilon,c}^{(1)^2}\cdot r_{\epsilon,c}(f^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}})+\frac{\lambda_{\epsilon,c}^{(1)}\cdot\lambda_{\epsilon,c}^{(2)}}{2}+\frac{\lambda_{\epsilon,c}^{(2)}}{2}\\
&-\frac{\lambda_{\epsilon,c}^{(2)}}{2}\left(b_2\cdot\lambda_{\epsilon,c}^{(1)}+\frac{b_1}{2}\right)\hat{f}^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}}(\emptyset)-\frac{\lambda_{\epsilon,c}^{(2)}\cdot b_1\cdot b_2}{4}.
\end{aligned}$$

Since $b_1, b_2 \in \{-1,+1\}$ and $|\hat{f}^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}}(\emptyset)| \leq 1$, we can upper bound as:

$$-\frac{\lambda_{\epsilon,c}^{(2)}}{2}\left(b_2\cdot\lambda_{\epsilon,c}^{(1)}+\frac{b_1}{2}\right)\hat{f}^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}}(\emptyset)-\frac{\lambda_{\epsilon,c}^{(2)}\cdot b_1\cdot b_2}{4}\leq\frac{\lambda_{\epsilon,c}^{(2)}}{2}-\frac{\lambda_{\epsilon,c}^{(1)}\cdot\lambda_{\epsilon,c}^{(2)}}{2}$$

and, finally, upper bound $r_{\epsilon,c}(f)$ as

$$r_{\epsilon,c}(f)\leq\lambda_{\epsilon,c}^{(1)^2}\cdot r_{\epsilon,c}(f^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}})+\lambda_{\epsilon,c}^{(2)},\tag{3.17}$$

where $f^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}}$ has $k-2$ relevant variables. We will now show the theorem by induction. For $k=1$, Eq. (3.15) simplifies to

$$r_{\epsilon,c}(f)\leq\lambda_{\epsilon,c}^{(2)},$$

which is obviously true. For $k=2$, Eq. (3.15) results in

$$r_{\epsilon,c}(f)\leq\left(\frac{1}{2}+\lambda_{\epsilon,c}^{(1)}\right)\lambda_{\epsilon,c}^{(2)},$$

which is also true (see Example 3.5).

Using Eq. (3.15) as the induction hypothesis, we will now show that Eq. (3.15) is true for k, if it is true for $k-2$. Since $f^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}}$ in Eq. (3.17) has $k-2$ relevant variables, we can apply our hypothesis:

$$\begin{aligned}
r_{\epsilon,c}(f)\leq&\lambda_{\epsilon,c}^{(1)^2}\left(\frac{\lambda_{\epsilon,c}^{(2)}}{1-\lambda_{\epsilon,c}^{(1)^2}}-\frac{1+(-1)^k(1-\lambda_{\epsilon,c}^{(1)})^2-2\lambda_{\epsilon,c}^{(1)}+\lambda_{\epsilon,c}^{(1)^2}+4\lambda_{\epsilon,c}^{(1)^3}}{4\lambda_{\epsilon,c}^{(1)^2}\cdot(1-\lambda_{\epsilon,c}^{(1)^2})}\lambda_{\epsilon,c}^{(1)^k}\cdot\lambda_{\epsilon,c}^{(2)}\right)+\lambda_{\epsilon,c}^{(2)}\\
=&\left(\frac{\lambda_{\epsilon,c}^{(1)^2}}{1-\lambda_{\epsilon,c}^{(1)^2}}+1-\frac{1+(-1)^k(1-\lambda_{\epsilon,c}^{(1)})^2-2\lambda_{\epsilon,c}^{(1)}+\lambda_{\epsilon,c}^{(1)^2}+4\lambda_{\epsilon,c}^{(1)^3}}{4(1-\lambda_{\epsilon,c}^{(1)^2})}\cdot\lambda_{\epsilon,c}^{(1)^k}\right)\lambda_{\epsilon,c}^{(2)}\\
=&\left(\frac{1}{1-\lambda_{\epsilon,c}^{(1)^2}}-\frac{1+(-1)^k(1-\lambda_{\epsilon,c}^{(1)})^2-2\lambda_{\epsilon,c}^{(1)}+\lambda_{\epsilon,c}^{(1)^2}+4\lambda_{\epsilon,c}^{(1)^3}}{4(1-\lambda_{\epsilon,c}^{(1)^2})}\lambda_{\epsilon,c}^{(1)^k}\right)\lambda_{\epsilon,c}^{(2)},
\end{aligned}$$

which concludes the induction. The generalization follows from the positivity of the second term of Eq. (3.15). ■

Proposition 3.6

Let f be a $\{\pi : A : B\}$-NCF with a fixed bias of $\hat{f}(\emptyset)$, then $r_{\epsilon,c}(f)$ is bounded as

$$r_{\epsilon,c}(f) < \lambda_{\epsilon,c}^{(2)} \frac{1 + {\lambda_{\epsilon,c}^{(1)}}^2 - \left(1 - {\lambda_{\epsilon,c}^{(1)}}^2\right) \cdot |\hat{f}(\emptyset)|}{2(1 - {\lambda_{\epsilon,c}^{(1)}}^2)\lambda_{\epsilon,c}^{(1)}}.$$

Proof: Using Proposition 3.5 and Theorem 3.2 we get:

$$\lambda_{\epsilon,c}^{(1)} r_{\epsilon,c}(f^{(\pi_1,\bar{a}_1)}) + \lambda_{\epsilon,c}^{(2)} \frac{1}{2}\left(1 - \hat{f}^{(\pi_1,\bar{a}_1)}(\emptyset) b_1\right) \leq \frac{\lambda_{\epsilon,c}^{(2)}}{1 - {\lambda_{\epsilon,c}^{(1)}}^2}$$

$$\lambda_{\epsilon,c}^{(1)} r_{\epsilon,c}(f^{(\pi_1,\bar{a}_1)}) - \lambda_{\epsilon,c}^{(2)} \frac{1}{2} \hat{f}^{(\pi_i,\bar{a}_1)}(\emptyset) b_1 \leq \frac{\lambda_{\epsilon,c}^{(2)}}{1 - {\lambda_{\epsilon,c}^{(1)}}^2} - \frac{\lambda_{\epsilon,c}^{(2)}}{2}.$$

Since $b_1 \in \{+1, -1\}$ and b_1 is independent of $f^{(\pi_1,\bar{a}_1)}(\emptyset)$, we can write:

$$\lambda_{\epsilon,c}^{(1)} r_{\epsilon,c}(f^{(\pi_1,\bar{a}_1)}) + \lambda_{\epsilon,c}^{(2)} \frac{1}{2} |\hat{f}^{(\pi_1,\bar{a}_1)}(\emptyset)| \leq \frac{\lambda_{\epsilon,c}^{(2)}}{1 - {\lambda_{\epsilon,c}^{(1)}}^2} - \frac{\lambda_{\epsilon,c}^{(2)}}{2},$$

$$r_{\epsilon,c}(f^{(\pi_i,\bar{a}_i)}) \leq \frac{\lambda_{\epsilon,c}^{(2)}\left(1 + {\lambda_{\epsilon,c}^{(1)}}^2\right) - \lambda_{\epsilon,c}^{(2)} |\hat{f}^{(\pi_i,\bar{a}_i)}(\emptyset)| \left(1 - {\lambda_{\epsilon,c}^{(1)}}^2\right)}{2(1 - {\lambda_{\epsilon,c}^{(1)}}^2)\lambda_{\epsilon,c}^{(1)}}.$$

Substituting $f^{(\pi_i,\bar{a}_i)}$ by f concludes the proof. ■

3.2.7. Noise Sensitivity for specific Nested Canalizing Functions

In this subsection we will investigate the NS, respectively $r_{\epsilon,c}$, for the special cases of NCFs as introduced in the previous chapter.

NCFs with only most dominant variables (NCF-MDV)

Let us first look at NCFs, whose relevant variables are all most canalizing, then the following proposition states an upper bound:

Proposition 3.7

Let f be a $\{\pi : A : B\}$-NCF-MDV with k uniformly distributed input variables, then

$$r_{\epsilon,c}(f) = \frac{2(2^{-k} - {\lambda_{\epsilon,c}^{(1)}}^k)}{1 - 2\lambda_{\epsilon,c}^{(1)}} \lambda_{\epsilon,c}^{(2)}.$$

Proof: Recalling Proposition 3.5 and with $|\hat{f}(\emptyset)| = 1 - \frac{1}{2^{k-1}}$ and $b_i = \operatorname{sgn}\left(\hat{f}(\emptyset)\right)$ for all i, we can write:

$$r_{\epsilon,c}(f) = \lambda_{\epsilon,c}^{(1)} r_{\epsilon,c}(f^{(\pi_1,\bar{a}_1)}) + \lambda_{\epsilon,c}^{(2)} \frac{1}{2^{k-1}}.$$

Solving the recursion concludes the proof. ■

NCFs with alternating canalized values (NCF-ACV)

In the case of alternating b_is we can derive the following formula for $r_{\epsilon,c}(f)$:

Proposition 3.8

Let f be a $\{\pi : A : B\}$-NCF-ACV under uniform distribution, then

$$r_{\epsilon,c}(f) = \frac{2\lambda_{\epsilon,c}^{(2)}}{3(1-\lambda_{\epsilon,c}^{(1)})} - \frac{2(-2)^{-k}\lambda_{\epsilon,c}^{(2)}}{3(1+2\lambda_{\epsilon,c}^{(1)})} + \frac{2{\lambda_{\epsilon,c}^{(1)}}^{k+1}\lambda_{\epsilon,c}^{(2)}}{(1-\lambda_{\epsilon,c}^{(1)})(1+2\lambda_{\epsilon,c}^{(1)})}, \tag{3.18}$$

where $k = rel(f)$ is the number of relevant variables of f.

Proof: Since the bias of a NCF-ACV is given as (see Corollary 2.4):

$$|\hat{f}(\emptyset)| = \frac{1}{3}\left(\frac{(-1)^k}{2^{k-1}} + 1\right),$$

and $\operatorname{sgn}\left(\hat{f}^{(\pi_1,\bar{a}_1)}\right) = b_2 = -b_1$, we can rewrite Proposition 3.5 as

$$\begin{aligned} r_{\epsilon,c}(f) &= \lambda_{\epsilon,c}^{(1)} r_{\epsilon,c}(f^{(\pi_1,\bar{a}_1)}) + \lambda_{\epsilon,c}^{(2)} \frac{1}{2}\left(1 + \frac{1}{3}\left(\frac{(-1)^{k-1}}{2^{k-2}} + 1\right)\right) \\ &= \lambda_{\epsilon,c}^{(1)} r_{\epsilon,c}(f^{(\pi_i,\bar{a}_i)}) + \lambda_{\epsilon,c}^{(2)} \frac{1}{3}\left(-\frac{(-1)^k}{2^{k-1}} + 2\right). \end{aligned} \tag{3.19}$$

Let us now show the proposition using induction. For $k = 1$, Eq. (3.18) becomes

$$r_{\epsilon,c}(f) = \lambda_{\epsilon,c}^{(2)},$$

which is obviously true. If we now apply our induction hypothesis (Eq. (3.18)) on Eq. (3.19) we get

$$\begin{aligned} r_{\epsilon,c}(f) &= \frac{2\lambda_{\epsilon,c}^{(1)}\lambda_{\epsilon,c}^{(2)}}{3(1-\lambda_{\epsilon,c}^{(1)})} - \frac{2(-2)^{-k+1}\lambda_{\epsilon,c}^{(1)}\lambda_{\epsilon,c}^{(2)}}{3(1+2\lambda_{\epsilon,c}^{(1)})} + \frac{2{\lambda_{\epsilon,c}^{(1)}}^{k+1}\lambda_{\epsilon,c}^{(2)}}{(1-\lambda_{\epsilon,c}^{(1)})(1+2\lambda_{\epsilon,c}^{(1)})} + \lambda_{\epsilon,c}^{(2)}\frac{1}{3}\left(-2(-2)^{-k}+2\right) \\ &= \frac{2\lambda_{\epsilon,c}^{(2)}}{3(1-\lambda_{\epsilon,c}^{(1)})} - \frac{2(-2)^{-k}\lambda_{\epsilon,c}^{(2)}}{3(1+2\lambda_{\epsilon,c}^{(1)})} + \frac{2{\lambda_{\epsilon,c}^{(1)}}^{k+1}\lambda_{\epsilon,c}^{(2)}}{(1-\lambda_{\epsilon,c}^{(1)})(1+2\lambda_{\epsilon,c}^{(1)})}, \end{aligned}$$

which concludes the induction. ■

If we now let k grow to infinity, i.e., if we observe a large number of relevant variables, we obtain the following corollary:

Corollary 3.11

Let f be a $\{\pi : A : B\}$-NCF-ACF, then for an infinite number of relevant variables we can write

$$r_{\epsilon,c}(f) = \frac{2\lambda_{\epsilon,c}^{(2)}}{3(1-\lambda_{\epsilon,c}^{(1)})}.$$

NCFs with second order most dominant variables (NCF-SMD)

Finally, we will discuss NCFs, whose restriction to $x_{\pi_1} = \bar{a}_1$ consists of only most dominating variables. For this class, which has a bias of 2^{-k+1}(see Corollary 2.5), we obtain the following $r_{\epsilon,c}(f)$:

Proposition 3.9

Let f be a $\{\pi : A : B\}$-NCF-SMD, i.e., with $b_1 \in \{-1, +1\}$ and $b_i = -b_1\ \forall i \in \{2, \dots, k\}$, with k uniformly distributed relevant variables, then

$$r_{\epsilon,c}(f) = \frac{1 - 2^{-k+1} - 2\lambda_{\epsilon,c}^{(1)} - 2{\lambda_{\epsilon,c}^{(1)}}^k + 2^{-k+3}\lambda_{\epsilon,c}^{(1)}}{1 - 2\lambda_{\epsilon,c}^{(1)}}\lambda_{\epsilon,c}^{(2)}.$$

Proof: Since $f^{(\pi_1,\bar{a}_1)}$ is a function, whose variables are all most dominant, we can rewrite Proposition 3.5 using Proposition 3.7 as:

$$r_{\epsilon,c}(f) = \lambda_{\epsilon,c}^{(1)}\frac{2(2^{-k+1} - {\lambda_{\epsilon,c}^{(1)}}^{k-1})}{1 - 2\lambda_{\epsilon,c}^{(1)}}\lambda_{\epsilon,c}^{(2)} + \frac{1}{2}\lambda_{\epsilon,c}^{(2)}\left(1 - \hat{f}^{(\pi_1,\bar{a}_1)}(\emptyset)b_1\right),$$

further, since $\operatorname{sgn}\left(\hat{f}^{(\pi_1,\bar{a}_1)}\right) = -b_1$ and $|\hat{f}^{(\pi_1,\bar{a}_1)}| = 1 - \frac{1}{2^{k-2}}$, we get

$$r_{\epsilon,c}(f) = \lambda_{\epsilon,c}^{(1)}\frac{2(2^{-k+1} - {\lambda_{\epsilon,c}^{(1)}}^{k-1})}{1 - 2\lambda_{\epsilon,c}^{(1)}}\lambda_{\epsilon,c}^{(2)} + \lambda_{\epsilon,c}^{(2)}\left(1 - \frac{1}{2^{k-1}}\right),$$

which concludes the proof. ■

Again, if we investigate these functions for an infinite number of relevant inputs, we obtain the following corollary. Further, these function then become balanced.

Corollary 3.12

Let f be a $\{\pi : A : B\}$-NCF-SMD, then for an infinite number of relevant variables we can write

$$r_{\epsilon,c}(f) = \lambda_{\epsilon,c}^{(2)}.$$

3.3. Average Sensitivity of Canalizing and Nested Canalizing Functions

3.3.1. Restricted and Canalizing Functions

First, we show the relation between the average sensitivity of a BF and the average sensitivity of its two restricted functions.

Theorem 3.3

Let $f^{(i,+)}, f^{(i,-)}$ be the restrictions of f to some relevant variable i of f. Then

$$\mathtt{as}(f) = \frac{1}{2}\mathtt{as}(f^{(i,+)}) + \frac{1}{2}\mathtt{as}(f^{(i,-)}) + \xi_{\epsilon,1}(f^{(i,+)}, f^{(i,-)}).$$

Proof: Follows directly from Theorem 3.1 using:

$$\mathtt{as}(f) = \left.\frac{dr_{\epsilon,1}(f)}{d\epsilon}\right|_{\epsilon=0}.$$

■

Using this result we can prove the following upper bound of the `as` of CFs:

Proposition 3.10

Let f be a CF with k relevant and uniformly distributed variables, then its `as` *is bounded as follows:*

$$\mathtt{as}(f) \leq \frac{1}{2}(k+1) - |\hat{f}(\emptyset)|$$

or

$$\mathtt{as}(f) \leq \frac{1}{2}(k-1).$$

Proof: Let us start with Theorem 3.3 and apply the facts that $\mathtt{as}(f^{(i,a_i)}) = 0$ and $\mathtt{as}(f^{(i,\bar{a}_i)})$ can be any BF with $k-1$ relevant variables, hence, $\mathtt{as}(f^{(i,\bar{a}_i)}) \leq k-1$. This leads us to:

$$\mathtt{as}(f) \leq \frac{1}{2}(k-1) + \frac{1}{2}\left(1 - \hat{f}^{(i,\bar{a}_i)}(\emptyset)\cdot b_i\right).$$

From Proposition 2.2 we know that:

$$\hat{f}(\emptyset) = \frac{1}{2}\left(\hat{f}^{(i,a_i)}(\emptyset) + \hat{f}^{(i,\bar{a}_i)}(\emptyset)\right) = \frac{1}{2}\left(\hat{f}^{(i,\bar{a}_i)}(\emptyset) + b_i\right).$$

Hence,

$$\frac{1}{2}\hat{f}^{(i,\bar{a}_i)}(\emptyset)\cdot b_i = \hat{f}(\emptyset)\cdot b_i - \frac{1}{2}.$$

Finally, we can write:

$$\mathtt{as}(f) \leq \frac{1}{2}(k-1) + 1 - \hat{f}(\emptyset)\cdot b_i.$$

Applying $|\hat{f}(\emptyset)| = \hat{f}(\emptyset)\cdot b_i$ leads to the first statement and from $|\hat{f}(\emptyset)| \leq 1$ the second statement follows. ■

Let us recall that in general the as of BFs is upper-bounded by k, hence, CFs have a lower as. However, it is still larger than the as of unate functions. Note, that although all NCFs are unate, not all CFs are. For example, a CF, whose restriction is an XOR function, is not unate. Our findings go along with those of [SK04], where an upper bound on the expected average sensitivity of random CFs has been shown.

3.3.2. Average Sensitivity of Nested Canalizing Functions

Using Theorem 3.3 we can directly obtain the following corollary, recursively describing the as of NCFs:

Corollary 3.13

The average sensitivity of a $\{\pi : A : B\}$-NCF with uniformly distributed input variables can recursively be described as:

$$\mathtt{as}(f) = \frac{1}{2}\left(\mathtt{as}(f^{(\pi_1,\bar{a}_1)}) + 1 - \hat{f}^{(\pi_1,\bar{a}_1)}(\emptyset)\cdot b_1\right). \tag{3.20}$$

In [LAM+13] an upper bound on the average sensitivity of NCFs has been conjectured. In the following theorem, we prove this conjecture to be correct.

Theorem 3.4

The average sensitivity of a NCF with $k = rel(f)$ relevant and uniformly distributed variables is bounded by

$$\frac{k}{2^{k-1}} \leq \mathtt{as}(f) \leq \frac{4}{3} - 2^{-k} - \frac{1}{3}\cdot 2^{-k}\cdot(-1)^k. \tag{3.21}$$

The bounds in Eq. (3.21) will turn out to be tight.

Proof: We first prove the upper bound in Eq. (3.21). Let us recall Corollary 3.13:

$$\mathtt{as}(f) = \frac{1}{2}\left(\mathtt{as}(f^{(\pi_1,\bar{a}_1)}) + 1 - \hat{f}^{(\pi_1,\bar{a}_1)}(\emptyset)\cdot b_1\right).$$

If we apply Corollary 3.13 again on $\mathtt{as}(f^{(\pi_1,\bar{a}_1)})$ and use Proposition 2.7 on $\hat{f}^{(\pi_1,\bar{a}_1)}(\emptyset)$, we get:

$$\begin{aligned}
\mathtt{as}(f) =& \frac{1}{2}\left[\frac{1}{2}\left(\mathtt{as}(f^{(\pi_1,\bar{a}_1)(\pi_2,\bar{a}_2)}) + 1 - \hat{f}^{(\pi_1,\bar{a}_1)(\pi_2,\bar{a}_2)}(\emptyset)\cdot b_2\right)\right.\\
&\left. + 1 - \left(\frac{1}{2}\hat{f}^{(\pi_1,\bar{a}_1)(\pi_2,\bar{a}_2)}(\emptyset) + \frac{1}{2}b_2\right)b_1\right]\\
=& \frac{1}{4}\mathtt{as}(f^{(\pi_1,\bar{a}_1)(\pi_2,\bar{a}_2)}) + \frac{3}{4} - \frac{1}{4}\hat{f}^{(\pi_1,\bar{a}_1)(\pi_2,\bar{a}_2)}(\emptyset)\cdot b_2\\
&- \frac{1}{4}\hat{f}^{(\pi_1,\bar{a}_1)(\pi_2,\bar{a}_2)}(\emptyset)\cdot b_1 - \frac{1}{4}b_2\cdot b_1\\
=& \frac{1}{4}\mathtt{as}(f^{(\pi_1,\bar{a}_1)(\pi_2,\bar{a}_2)}) - \frac{1}{4}\hat{f}^{(\pi_1,\bar{a}_1)(\pi_2,\bar{a}_2)}(\emptyset)\cdot(b_1+b_2)\\
&- \frac{1}{4}b_2.b_1 + \frac{3}{4}.
\end{aligned}$$

Since $b_1, b_2 \in \{-1, +1\}$ and $|\hat{f}^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}}(\emptyset)| \leq 1$,

$$-\frac{1}{4}\hat{f}^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}}(\emptyset)\cdot(b_1+b_2)-\frac{1}{4}b_2\cdot b_1 \leq \frac{1}{4}.$$

Thus we obtain

$$\mathtt{as}(f) \leq \frac{1}{4}\mathtt{as}(f^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}})+1, \tag{3.22}$$

where $f^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}}$ has $k-2$ relevant variables. We will now show the theorem by induction. For $k=1$ the upper bound in Eq. (3.21) simplifies to

$$\mathtt{as}(f) \leq 1,$$

which is obviously true by definition. For $k=2$ the upper bound in Eq. (3.21) results in

$$\mathtt{as}(f) \leq 1,$$

which is also true and can be verified by inspecting all possible functions.

Using Eq. (3.21) as the induction hypothesis, and applying it on $f^{(\pi_1,\bar{a}_1)^{(\pi_2,\bar{a}_2)}}$ in Eq. (3.22), which has $k-2$ relevant variables, yields to:

$$\begin{aligned}\mathtt{as}(f) &\leq \frac{1}{4}\left(\frac{4}{3}-2^{-(k-2)}-\frac{1}{3}\cdot 2^{-(k-2)}\cdot(-1)^{k-2}\right)+1\\ &= \frac{4}{3}-2^{-k}-\frac{1}{3}2^{-k}(-1)^k,\end{aligned}$$

which concludes the induction.

The lower bound in Eq. (3.21) can be proven along the lines of the proof of the upper bound, using the following inequality, which follows from Corollary 3.13 and Proposition 2.8:

$$\mathtt{as}(f) \geq \frac{1}{2}\left(\mathtt{as}(f^{(\pi_1,\bar{a}_1)})+\frac{1}{2^{k-2}}\right).$$

■

The tightness of the bounds in Eq. (3.21) is shown in Propositions 3.11 and 3.12.

We can further upper-bound the right hand side of Theorem 3.4 in order to make it independent of the number of relevant variables k:

Corollary 3.14

The average sensitivity of a NCF with uniformly distributed variables satisfies

$$\mathtt{as}(f) < \frac{4}{3}.$$

We next show that the bounds in Theorem 3.4 are tight.

Proposition 3.11

Let f be a NCF-MDV, then f satisfies the lower bound in Theorem 3.4 with equality.

Proof: Starting from Corollary 3.13 and using that $|\hat{f}(\emptyset)| = 1 - \frac{1}{2^{k-1}}$ (see Corollary 2.3) and $b_i = sgn(\hat{f}(\emptyset))$ for all i, we get:

$$\begin{aligned} \texttt{as}(f) &= \frac{1}{2}\left(\texttt{as}(f^{(\pi_1,\bar{a}_1)}) + 1 - \left(1 - \frac{1}{2^{k-2}}\right)\right) \\ &= \frac{1}{2}\left(\texttt{as}(f^{(\pi_1,\bar{a}_1)}) + \frac{1}{2^{k-2}})\right). \end{aligned} \tag{3.23}$$

Since $\texttt{as}(f)$ depends on k relevant variables, while $\texttt{as}(f^{(\pi_1,\bar{a}_1)})$ depends only on $k-1$ relevant variables, Eq. (3.23) becomes:

$$\texttt{as}(k) = \frac{1}{2}\left(\texttt{as}(k-1) + \frac{1}{2^{k-2}})\right).$$

The proof is concluded by solving this recursion using induction. ■

Proposition 3.12

Let f be a NCF-ACV, i.e., with alternating b_i, then f fulfills the upper bound in Eq. (3.21) of Theorem 3.4 with equality.

Proof: Similar to the proof of the previous proposition we start from Corollary 3.13 and use $|\hat{f}(\emptyset)| = \frac{1}{3}\left(\frac{1}{2^{k-1}}(-1)^k + 1\right)$. The proof is established by solving the recursion. ■

Proposition 3.11 shows that the maximum is achieved, if the absolute value of the zero coefficient is minimal. If we are interested in the behavior of the as, if the zero coefficient is maximal, we need to investigate NFC-SMDs:

Proposition 3.13

The as of a NCF-SMD with k relevant and uniformly distributed input variables is given as:

$$\texttt{as}(f) = 1 + 2^{-k+1}(1-k).$$

Proof: The proof goes along the lines of the proofs of the two previous propositions. ■

Hence, for $k \to \infty$ the as becomes 1.

In general, we can give a bound on the average sensitivity for fixed $|\hat{f}(\emptyset)|$, as shown in the following proposition:

Proposition 3.14

Let f be a NCF with uniformly distributed inputs. Then

$$\texttt{as}(f) \leq \frac{5}{3} - |\hat{f}(\emptyset)|.$$

Proof: Combining Corollaries 3.13 and 3.14, we get:

$$\texttt{as}(f^{(\pi_1,\bar{a}_1)}) - b_1 \cdot \hat{f}^{(\pi_1,\bar{a}_1)}(\emptyset) \leq \frac{5}{3},$$

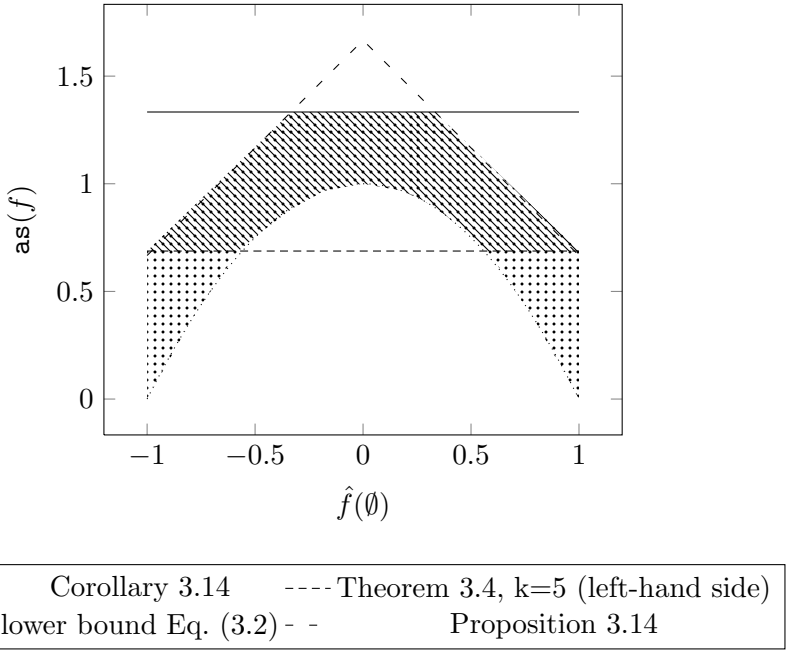


Figure 3.5.: Bounds on the average sensitivity: The dotted-area corresponds to the possible values for the average sensitivity of a NCF, the lined area to BFs with $k = 5$ input variables.

and since $b_i \in \{-1, +1\}$:

$$\mathtt{as}(f^{(\pi_1,\bar{a}_1)}) + |\hat{f}^{(\pi_1,\bar{a}_1)}(\emptyset)| \leq \frac{5}{3}.$$

Substituting $f^{(\pi_1,\bar{a}_1)}$ by f concludes the proof. ■

3.3.3. Discussion and Implications

In Figure 3.5 we summarize the bounds from the previous subsections. Specifically, we plot the average sensitivity versus the zero coefficient. Additionally, we include a lower bound on the average sensitivity, which is independent of the number of relevant variables and applies for any BF and can be found in Eq. (3.2). One can see that this bound intersects with our lower bound from Theorem 3.4 (which we plotted for $k = 5$), though we stated that our bound is tight. However, this is not a contradiction, since this lower bound is achieved for functions with large absolute zero coefficients, which are located outside the intersection.

For $k = 5$ our lower bound forms a triangle with the upper bound as formulated in Proposition 3.14. The NCFs with all variables being most dominant are located in the left and right corners of that triangle. However the lower bound decreases in k and with it the most dominant NCFs.

The upper bound in Corollary 3.14 also intersects with the bound of Proposition 3.14. Again, this is not a contradiction, since NCFs reach this bound only for small absolute zero coefficients.

In general the average sensitivity is upper bounded by k, i.e., $\mathtt{as}(f) \leq k$, as previously discussed. Further, for unate functions, the average sensitivity is upper bounded by $\mathtt{as}(f) \leq \sqrt{(1-\hat{f}(\emptyset))k}$. This bound is tight up to a multiplicative constant, see e.g., [MO03]. NCFs form a subclass of unate functions. Thus, our results show that even within the class of unate functions, the average sensitivity of NCFs is remarkably low. Since a low average sensitivity has a positive effect on the stability of Boolean networks [SK04], our result gives an explanation for the remarkable stability of Boolean networks with NCFs.

It is worth noting that all those results rely on the assumption of uniformly distributed inputs. This rises the question, if the results can be generalized to other distributions. The recursive representations can easily be extended to product distributed input variables. But without further constraints there always exists a distribution, which maximizes the average sensitivity, i.e., for any function with k relevant variables the average sensitivity can be k.

Chapter 4

Mutual Information of Boolean Functions

Mutual Information of Boolean functions is a key measure of information processing abilities of Boolean networks. It has been stated that networks, whose dynamical behavior is critical, i.e., at the edge between stable and chaotic behavior, have somehow an optimized information processing ability [Lan90, SML+96, LF00]. We have seen before that canalizing and nested canalizing Boolean functions seem to have a stabilizing effect as they have a low average sensitivity.

The mutual information based on Shannon's theory [Sha48] is a measure for the statistical relation between the input and output of a system. It has already been applied in the context of Boolean networks and functions, such as cellular automata [Lan90], random Boolean networks [LF00, RKLP+08] and iterated function systems [CY90].

In this chapter, we will investigate two scenarios. In this first scenario, we examine the mutual information between one input or a set of inputs and the function's output(Section 4.2). Secondly, we investigate the mutual information between all inputs, which are distributed by BSCs, and the function's output (Section 4.3). To do so, let us first revisit the fundamentals of mutual information.

4.1. Fundamentals

The mutual information (MI) between two random variables is defined as:

$$MI(Y;X) = H(Y) - H(Y|X),$$

where

$$H(X) = -\sum_{x\in\mathcal{X}} \mathrm{P}_X(x)\log_2(\mathrm{P}_X(x))$$

is Shannon's entropy in bits of some discrete random variable X with its domain $\mathcal{X}$. For the special case that $|\mathcal{X}| = 2$, it reduces to the binary entropy function:

$$h(p) = -p\log_2(p) - (1-p)\log_2(1-p),$$

with $p = \mathrm{P}_X(+1)$. Further, $H(Y|X)$ is the conditional entropy between two discrete random variables $X \in \mathcal{X}$ and $Y \in \mathcal{Y}$

$$H(Y|X) = \sum_{x\in\mathcal{X}} \mathrm{P}_X(x)\, H(Y|X=x),$$

with

$$H(Y|X=x) = -\sum_{y\in\mathcal{Y}} \mathrm{P}_{Y|X}(y|x)\log_2 \mathrm{P}_{Y|X}(y|x)\,.$$

Similar, we can give the conditional entropy between a discrete vector $\mathbf{X}\in\mathcal{X}^n$, n being an integer, and Y as

$$H(Y|\mathbf{X}) = \sum_{\mathbf{x}\in\mathcal{X}^n} \mathrm{P}_{\mathbf{X}}(\mathbf{x})\, H(Y|\mathbf{X}=\mathbf{x}),$$

with

$$H(Y|\mathbf{X}=\mathbf{x}) = -\sum_{y\in\mathcal{Y}} \mathrm{P}_{Y|\mathbf{X}}(y|\mathbf{x})\log_2 \mathrm{P}_{Y|\mathbf{X}}(y|\mathbf{x})\,.$$

4.2. Mutual Information of Boolean Functions

Recently, a relation between the mutual information and the Fourier spectra has been derived [HSB13]. It was shown that the MI between a set of input variables $T\subseteq[n]$ and the functions output is given as:

$$MI(f(\mathbf{X});\mathbf{X}_T) = h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)\right)\right) - \mathop{\mathbf{E}}_{\mathbf{X}_T}\left[h\left(\frac{1}{2}\left(1+\sum_{S\subseteq T}\phi_S(\mathbf{X}_T)\hat{f}(S)\right)\right)\right], \tag{4.1}$$

where $\mathbf{X}_T = \{X_i : i\in T\}$. In particular, it has been shown that the mutual information between one input variable and the output only depends on two coefficients, namely the zero coefficient and the first order coefficient of this variable, i.e.,

$$MI(f(\mathbf{X});X_i) = h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)\right)\right) - \mathop{\mathbf{E}}_{X_i}\left[h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)+\hat{f}(\{i\})\cdot\phi_{\{i\}}(X_i)\right)\right)\right]. \tag{4.2}$$

We will first investigate the single-variable MI in the following section before we will generalize our results for the multi-variable case.

4.2.1. Single-Variable Mutual Information of Boolean Functions

The fact that the MI depends only on $\hat{f}(\emptyset)$ and $\hat{f}(\{i\})$ coincides with our statement in Chapter 2 that the canalizing property also depends on these two coefficients. Hence, we will only focus on those two Fourier coefficients in the following considerations. The remaining coefficients can be chosen arbitrarily and have no influence on our findings. Also, the number of input variables n does not restrict our investigations, it only determines the possible values $\hat{f}(\emptyset)$ and $\hat{f}(\{i\})$, since they are a multiple of 2^{-n}.

4.2.1.1. Mutual Information under Uniform Distribution

For sake of clarity and ease of comprehension we will first focus on canalizing functions in the uniform case. After that we will then generalize this result to product distributed variables.

Theorem 4.1

Let $\mathbf{X} \in \Omega^n$ be a uniformly distributed Boolean vector and let $-1 \leq \mu \leq +1$ be a constant. A BF g maximizes the mutual information between one input variable X_i and the output of a BF with fixed expectation μ, i.e.,

$$\max_{f:\mathbf{E}[f]=\mu} \{MI(f(\mathbf{X}); X_i)\} = MI(g(\mathbf{X}); X_i),$$

if and only if g is canalizing in x_i.

Proof: Since $\hat{f}(\emptyset) = \mathbf{E}[f(\mathbf{X})]$ is constant, the only remaining degree of freedom in Eq. (4.2) is $\hat{f}(\{i\})$. First, we show that the mutual information is convex with respect to $\hat{f}(\{i\})$. Since the first summand of Eq. (4.2) only depends on $\hat{f}(\emptyset)$ we can consider it constant. We hence can focus on the second part, which we can write as:

$$\mathop{\mathbf{E}}_{X_i}\left[-h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)+\hat{f}(\{i\})\cdot\chi_{\{i\}}(X_i)\right)\right)\right].$$

The binary entropy function $h(\cdot)$ is concave and since its argument is an affine mapping, $-h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)+\hat{f}(\{i\})\chi_{\{i\}}(X_i)\right)\right)$ is convex [BV04]. Finally, the expectation is a non-negative weighted sum, which preserves convexity, i.e., the mutual information is convex.

Obviously, for $\hat{f}(\{i\}) = 0$ the mutual information is minimized, hence, due to the convexity, the maximum can be found on the boundaries of the domain. The domain is limited by the non-negativity of the arguments of h, i.e.,

$$0 \leq \frac{1}{2}\left(1+\hat{f}(\emptyset)+\hat{f}(\{i\})\cdot\chi_{\{i\}}(x_i)\right) \leq 1.$$

Thus, the boundaries are given by

$$\hat{f}(\emptyset)+\hat{f}(\{i\})\cdot\chi_{\{i\}}(x_i) = \pm 1. \tag{4.3}$$

Hence, a BF g, which maximizes the MI, satisfies Eq. (4.3) for a particular x_i.

It can be seen from Theorem 2.2 that all functions, which are canalizing in variable i, are located on the boundary of the domain of the mutual information. The converse also holds, i.e., any function on the boundary of the domain is canalizing. Thus, any BF g, which maximizes the MI, is canalizing.

Finally, we need to show that any canalizing function with fix $\mathbf{E}[f(\mathbf{X})] = \mu$ maximizes the MI. We will do so by showing that all these canalizing functions have the same MI. We have seen before (Theorem 2.2) that these functions are constrained with:

$$\hat{f}(\{i\}) = \frac{b_i - \hat{f}(\emptyset)}{\chi_{\{i\}}(a_i)}.$$

Since $a_i, b_i \in \{-1, +1\}$, there exist four such types of functions on the boundary. Examining their mutual information leads us to:

$$MI(f(\mathbf{X}); X_i) = h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)\right)\right) - \mathop{\mathbf{E}}_{X_i}\left[h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)+\left(b_i-\hat{f}(\emptyset)\right)\frac{\chi_{\{i\}}(X_i)}{\chi_{\{i\}}(a_i)}\right)\right)\right],$$

which yields into:

$$\begin{aligned} MI(f(\mathbf{X}); X_i) &= h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)\right)\right) \\ &- \mathrm{P}_{X_i}(a_i)\underbrace{h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)+\left(b_i-\hat{f}(\emptyset)\right)\frac{\chi_{\{i\}}(a_i)}{\chi_{\{i\}}(a_i)}\right)\right)}_{=0} \\ &- \mathrm{P}_{X_i}(-a_i)\,h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)+\left(b_i-\hat{f}(\emptyset)\right)\frac{\chi_{\{i\}}(-a_i)}{\chi_{\{i\}}(a_i)}\right)\right), \end{aligned}$$

and hence:

$$\begin{aligned} MI(f(\mathbf{X}); X_i) &= h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)\right)\right) \\ &- \mathrm{P}_{X_i}(-a_i)\,h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)+\left(b_i-\hat{f}(\emptyset)\right)\frac{\chi_{\{i\}}(-a_i)}{\chi_{\{i\}}(a_i)}\right)\right). \end{aligned} \tag{4.4}$$

For the uniformly distributed case we write:

$$MI(f(\mathbf{X}); X_i) = h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)\right)\right) - \frac{1}{2}h\left(\frac{1-b_i}{2}+\hat{f}(\emptyset)\right).$$

Due to $b_i = \mathrm{sgn}\left(\hat{f}(\emptyset)\right)$ and the symmetry of h, we finally get:

$$MI(f(\mathbf{X}); X_i) = h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)\right)\right) - \frac{1}{2}h\left(|\hat{f}(\emptyset)|\right).$$

Hence, the mutual information is independent from a_i and b_i, which concludes the proof. ■

4.2.1.2. Mutual Information under Product Distribution

The result from the previous section can be extended to product distributed input variables. We will see that the probability distribution of the canalizing variable plays a key role in maximizing the MI.

Theorem 4.2

Let $\mathbf{X} \in \Omega^n$ be a product distributed Boolean vector and let $-1 \leq \mu \leq +1$ be a constant. A BF g maximizes the mutual information between one input variable X_i and the output of a BF with fixed expectation μ, i.e.,

$$\max_{f:\mathbf{E}[f]=\mu}\{MI(f(\mathbf{X}); X_i)\} = MI(g(\mathbf{X}); X_i),$$

if and only if g is $< i : a_i : b_i >$-canalizing, where a_i and b_i are chosen as follows:

$$(a_i, b_i) = \begin{cases} (sgn(\mu_i), sgn(\mathbf{E}[f(\mathbf{X})])) & |\mathbf{E}[f(\mathbf{X})]| \geq |\mu_i| \\ (-sgn(\mu_i), -sgn(\mathbf{E}[f(\mathbf{X})])) & |\mathbf{E}[f(\mathbf{X})]| < |\mu_i|. \end{cases} \tag{4.5}$$

Proof: The first part of this proof follows the proof of Theorem 4.1, where we simply replace $\chi_U(\mathbf{x})$ by $\phi_U(\mathbf{x})$. Hence, we can again show that the MI is convex and that the boundary consists of the canalizing functions. Hence, any function g, which maximizes the MI, is canalizing. Now, we need to show that any canalizing function that satisfies the constraints from Eq. (4.5) maximizes the MI.

Starting from Eq. (4.4), we get

$$MI(f(\mathbf{X}); X_i) = h\left(\frac{1}{2}\left(1 + \hat{f}(\emptyset)\right)\right) - H(f(\mathbf{X})|X_i),$$

where

$$H(f(\mathbf{X})|X_i) = \mathrm{P}_{X_i}(-a_i)\, h\left(\frac{1}{2}\left(1 + \hat{f}(\emptyset) - \left(b_i - \hat{f}(\emptyset)\right)\frac{(\mu_i + a_i)^2}{\sigma_i^2}\right)\right).$$

Obviously, there are four possible sets of a_i and b_i. However, for each choice of $\hat{f}(\emptyset)$ there exist two possible choices of a_i and b_i, of which only one maximizes the MI.

Let's first look at the possible combinations of a_i and b_i in dependence of $\hat{f}(\emptyset)$. From Parseval's theorem we know that

$$\hat{f}(\emptyset)^2 + \hat{f}(\{i\})^2 \leq 1,$$

and hence

$$\hat{f}(\emptyset)^2 + \left(\frac{b_i - \hat{f}(\emptyset)}{\phi_{\{i\}}(a_i)}\right)^2 \leq 1.$$

Solving this inequation for $\hat{f}(\emptyset)$ leads us to the possible sets of a_i and b_i:

$$\begin{aligned}
&a_i = \pm 1 \text{ and } b_i = -1 && \text{if } -1 \leq \hat{f}(\emptyset) \leq -|\mu_i| \\
&a_i = -\mathrm{sgn}(\mu_i) \text{ and } b_i = \pm 1 && \text{if } -|\mu_i| \leq \hat{f}(\emptyset) \leq |\mu_i| \\
&a_i = \pm 1 \text{ and } b_i = 1 && \text{if } |\mu_i| \leq \hat{f}(\emptyset) \leq 1.
\end{aligned}$$

Hence, to maximize the MI, we have to minimize $H(f(\mathbf{X})|X_i)$ for each possible choice of $\hat{f}(\emptyset)$. We can rewrite $H(f(\mathbf{X})|X_i)$ for all four combinations of a_i and b_i as follows:

$$H(f(\mathbf{X})|X_i) = \begin{cases}
s(\hat{f}(\emptyset)) = \mathrm{P}_{X_i}(-1)\, h\left(\frac{\hat{f}(\emptyset) - \mu_i}{1 - \mu_i}\right) & \text{if } a_i = +1 \text{ and } b_i = +1 \\
t(\hat{f}(\emptyset)) = \mathrm{P}_{X_i}(+1)\, h\left(\frac{\hat{f}(\emptyset) + \mu_i}{1 + \mu_i}\right) & \text{if } a_i = -1 \text{ and } b_i = +1 \\
q(\hat{f}(\emptyset)) = \mathrm{P}_{X_i}(-1)\, h\left(\frac{\hat{f}(\emptyset) + 1}{1 - \mu_i}\right) & \text{if } a_i = +1 \text{ and } b_i = -1 \\
r(\hat{f}(\emptyset)) = \mathrm{P}_{X_i}(+1)\, h\left(\frac{\hat{f}(\emptyset) + 1}{1 + \mu_i}\right) & \text{if } a_i = -1 \text{ and } b_i = -1.
\end{cases}$$

Now, let's assume $-1 \leq \hat{f}(\emptyset) \leq -|\mu_i|$, i.e., $b_i = -1$. Hence, we have to compare $q(\hat{f}(\emptyset))$ and $r(\hat{f}(\emptyset))$ and search for the correct choice of a_i that maximizes the MI. Lemma 1 (see Appendix A.1) shows that this is achieved by choosing $a_i = \mathrm{sgn}(\mu_i)$.

If $|\mu_i| \leq \hat{f}(\emptyset) \leq 1$, i.e., $b_i = +1$, we have again to compare the choices of $a_i = +1$ and $a_i = -1$. As above, a_i must be chosen to be $\operatorname{sgn}(\mu_i)$ in order to maximize the MI (see Lemma 2 in Appendix A.2).

Now let $-|\mu_i| \leq \hat{f}(\emptyset) \leq |\mu_i|$, hence we need to choose between $b_i = -1$ and $b_i = +1$. Lemma 3 (Appendix A.3) shows that in this case the MI is maximized, if $b_i = -\operatorname{sgn}\left(\hat{f}(\emptyset)\right) = -\operatorname{sgn}(\mathbf{E}[f(\mathbf{X})])$, which concludes the proof. ■

4.2.2. Multi-Input Mutual Information

So far, we only considered the MI between one variable and the function's output. Now, let's look at the MI between a set $T \subseteq [n]$ of variables and the output. The following theorem shows that this MI is maximized by jointly canalizing functions (see Subsection 2.3.1.2).

Theorem 4.3

Let $\mathbf{X} \in \Omega^n$ be a product distributed Boolean and let $-1 \leq \mu \leq +1$ be a constant. A BF g maximizes the mutual information between a set $T \subseteq [n]$ of variables and the output of a BF with fixed expectation μ, i.e.,

$$\max_{f:\mathbf{E}[f]=\mu} \{MI(f(\mathbf{X}); X_T)\} = MI(g(\mathbf{X}); X_T),$$

if g is jointly canalizing in T.

Proof: Again, the MI is convex with respect to $\hat{f}(S), S \subseteq T$ (see proof of Theorem 4.1) and has its minimum at $\hat{f}(S) = 0 \forall S \subseteq T$. Hence, the maximum is again at the boundary of the domain. The domain is limited by the non-negativity of the arguments of h, i.e.,

$$0 \leq \frac{1}{2}\left(1 + \sum_{S \subseteq T} \phi_S(\mathbf{X}_T)\hat{f}(S)\right) \leq 1,$$

and hence, we can write

$$\sum_{S \subseteq T} \phi_S(\mathbf{X}_T)\hat{f}(S) = \pm 1.$$

One can see from Proposition 2.4 that the boundary consists only of jointly canalizing functions, which concludes the proof. ■

It remains open, which actual choice of A and b_T has to be done in order to determine the actual jointly canalizing functions, which maximize the MI.

4.2.3. Discussion and Implications

To visualize our findings we plotted in Figure 4.1 a 3D diagram of the single variable mutual information of a BF with uniformly distributed input variables versus $\hat{f}(\emptyset)$ and $\hat{f}(\{i\})$. In Figure 4.2 we present a projection of the surface in the $(\hat{f}(\emptyset), MI)$-plane. It can be seen from these pictures that the canalizing functions form the boundary of the domain of the MI. Further, the symmetry with respect to a_i and $b_i = \operatorname{sgn}\left(\hat{f}(\emptyset)\right)$ can be seen. In addition, it

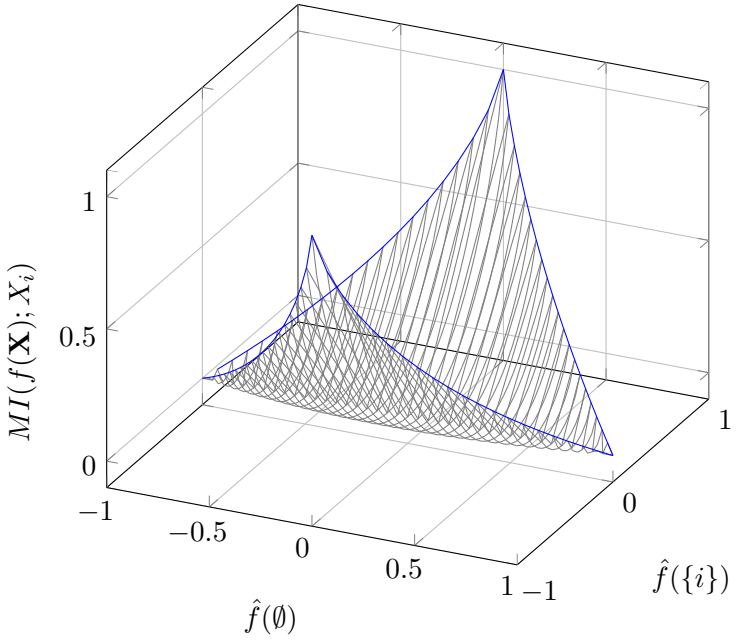


Figure 4.1.: Mutual information of BFs with uniformly distributed input variables versus $\hat{f}(\emptyset)$ and $\hat{f}(\{i\})$, all in i canalizing functions are located on the border (blue line).

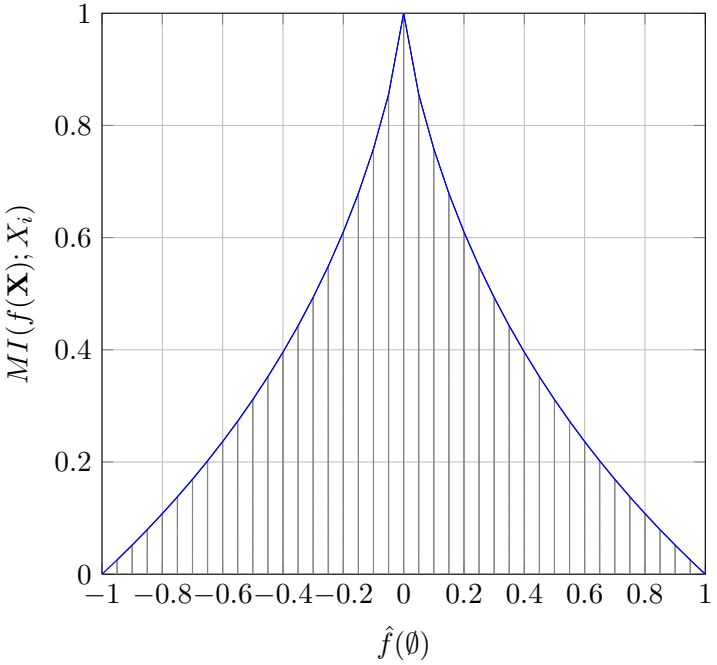


Figure 4.2.: Mutual information of Fig. 4.1 projected in the $(\hat{f}(\emptyset), MI)$-plane, all in variable i canalizing functions are located on the border (blue line).

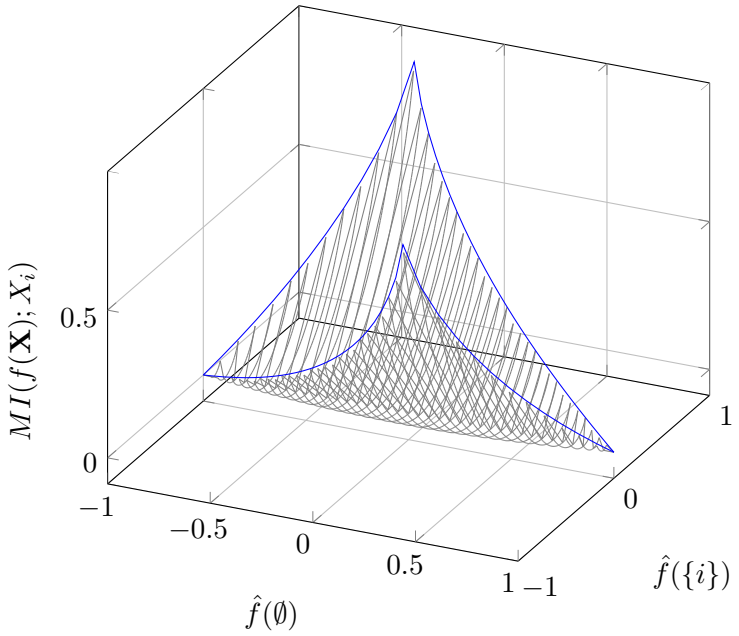


Figure 4.3.: Mutual information of a BF with product distributed input variables versus $\hat{f}(\emptyset)$ and $\hat{f}(\{i\})$, $p_i = 0.3$, all in variable i canalizing functions are located on the border (blue line).

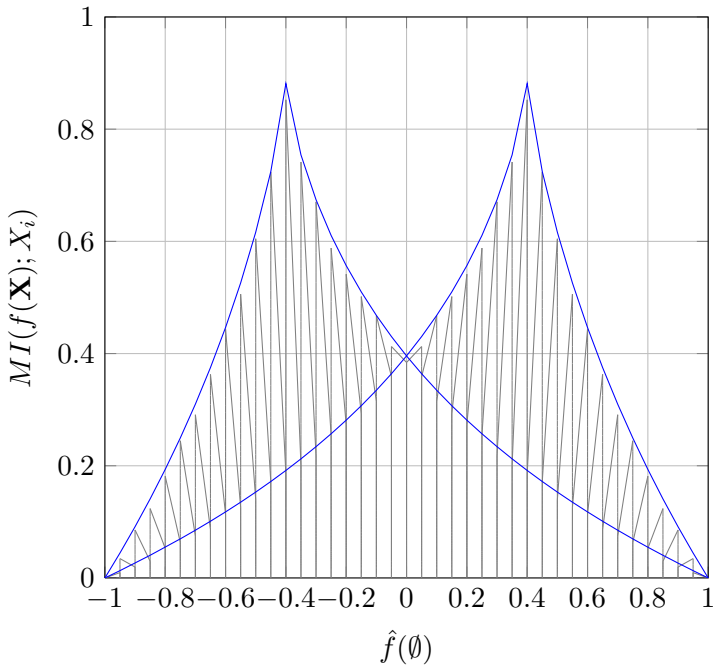


Figure 4.4.: Mutual information of Fig. 4.3 projected in the $(\hat{f}(\emptyset), MI)$-plane, all in i canalizing functions are located on the border (blue line).

becomes clear, that the mutual information also depends on the actual zero coefficient. This is mainly due to the first term of the MI (Eq. 4.2), the entropy of the function's output.

In Figures 4.3 and 4.4 the same plots can be found for product distributed input variables, with $p_i = 0.3$. Here, the skew of the mutual information towards the more probable canalizing value and the symmetry with respect to b_i can bee seen.

One can clearly see from the plots of the uniform case that the maximal MI becomes larger, the smaller the absolute value of $\hat{f}(\emptyset)$. Due to the fact that for canalizing variables $|\hat{f}(\emptyset)| + |\hat{f}(\{i\})| = 1$, $|\hat{f}(\{i\})|$ has to become large. Hence, the BF becomes similar to a dictatorship function, which is controlled by only one input. We have seen before that $|\hat{f}(\emptyset)|$ of NCF-SMDs becomes close zero for a large number of relevant inputs k. Further, we have shown before that these functions have a remarkably low $\mathtt{as}$, namely $\mathtt{as} \leq \frac{4}{3}$. This proves that these functions are epsilon-close to a function depending only on a few variables as shown in [Fri98]. Two functions f, g are called epsilon-close, if $\Pr[f \neq g] \leq \epsilon$. This leads us to the conclusion that those functions, which maximize the MI, have a bias close to zero and, though they have a large number of relevant variables, only a few variables have a significant influence.

4.3. Mutual Information with Noisy Inputs

The channel model, we will discuss in this section, is equal to the one for investigating the noise sensitivity, which is formally defined as:

Definition 4.1

Let $\mathbf{X}$ *be a uniformly distributed Boolean vector, which is transmitted over* n *parallel BSCs, i.e.,* $\mathbf{X'} = N_\epsilon(\mathbf{X})$*. The obtained noisy copy* $\mathbf{X'}$ *is then applied to the BF* f *(see Figure 4.5).*

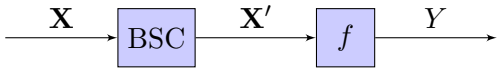


Figure 4.5.: Boolean Function with Noisy Inputs.

In the following subsection we will derive some relations between the MI and the NS, which we will then use to establish some bounds. We limit our investigations to the uniform case. This implicates that $\mathbf{X}$ and consequently $\mathbf{X'} = N_\epsilon(\mathbf{X})$ are both uniformly distributed. Furthermore, the probability distribution of the function's output also remains unchanged when adding noise to the input.

4.3.1. Relationship between Noise Sensitivity and Mutual Information

First, let us prove a relation between the noisy MI and the conditional noise sensitivity (CNS):

Proposition 4.1

Given the system of Definition 4.1, the mutual information between an input $\mathbf{X}$ *and the output* Y *is given as:*

$$MI_f(Y,\mathbf{X}) = h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)\right)\right) - \mathbf{E}\left[h\left(\mathtt{NS}_{\epsilon,\mathbf{x}}(f)\right)\right].$$

Proof: If we define $p(\mathbf{x}) = \Pr[\mathbf{X}=\mathbf{x}]$, we can write:

$$\begin{aligned} MI_f(Y,\mathbf{X}) &= H(Y) - H(Y|\mathbf{X}) \\ &= H(Y) - \sum_{\mathbf{x}} p(\mathbf{x}) H(Y|\mathbf{X}=\mathbf{x}) \\ &= H(Y) - \sum_{\mathbf{x}} p(\mathbf{x}) h\left(\Pr[Y=+1|\mathbf{X}=\mathbf{x}]\right) \\ &= H(Y) - \sum_{\mathbf{x}} p(\mathbf{x}) h\left(\Pr[Y=-1|\mathbf{X}=\mathbf{x}]\right), \end{aligned}$$

where the last step follows from $\Pr[Y=+1|\mathbf{X}=\mathbf{x}] = 1-\Pr[Y=-1|\mathbf{X}=\mathbf{x}]$ and the symmetry of $h(p)$. Since $f(\mathbf{X})$ is either $+1$ or -1, we can write

$$\begin{aligned} MI_f(Y,\mathbf{X}) &= H(Y) - \sum_{\mathbf{x}} p(\mathbf{x}) h\left(\Pr[Y \neq f(\mathbf{X})|\mathbf{X}=\mathbf{x}]\right) \\ &= H(Y) - \sum_{\mathbf{x}} p(\mathbf{x}) h\left(\Pr\left[f(\mathbf{X}') \neq f(\mathbf{X})|\mathbf{X}=\mathbf{x}\right]\right) \\ &= H(Y) - \sum_{\mathbf{x}} p(\mathbf{x}) h\left(\mathtt{NS}_{\epsilon,\mathbf{x}}(f)\right), \end{aligned}$$

which concludes the proof. ■

Using this proposition we can directly give a relation between the MI and the NS of a BF.

Proposition 4.2

Given the system of Definition 4.1, the mutual information between an input $\mathbf{X}$ *and the output* Y *can be lower-bounded as:*

$$MI_f(Y,\mathbf{X}) \geq h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)\right)\right) - h\left(\mathtt{NS}_{\epsilon}(f)\right).$$

Proof: The proposition follows from the convexity of $-h(p)$, Jensen's inequality and Eq. (3.4). ■

Hence, with every upper bound on the NS we can directly give a lower bound on the MI. This on the other hand, means that a low NS directly yields a high MI, if $\hat{f}(\emptyset)$ is fixed. In particular, this indicates that in general CFs and NCFs have a higher MI than other BFs. A further very simple general lower bound on the MI can be derived by combining Eqs. (3.2)

and (3.8) with Proposition 4.2:

$$MI(Y, \mathbf{X}) \geq h\left(\frac{1}{2}\left(1 + \hat{f}(\emptyset)\right)\right) - h(\mathtt{as}(f) \cdot k). \tag{4.6}$$

Let us establish another relationship, this time between $r_{\epsilon,2}$ as defined in Eq. (3.10) on page 35 and the MI.

Proposition 4.3 (*Steffen Schober, personal communication [Sch]*)

Given the system of Definition 4.1, then

$$2 \cdot r_{\epsilon,2}(f) \leq H(Y|\mathbf{X}) \leq (2 \cdot r_{\epsilon,2}(f))^{\frac{1}{\ln 4}}$$

and hence

$$h\left(\frac{1}{2}\left(1 + \hat{f}(\emptyset)\right)\right) - (2 \cdot r_{\epsilon,2}(f))^{\frac{1}{\ln 4}} \leq MI(Y, \mathbf{X}) \leq h\left(\frac{1}{2}\left(1 + \hat{f}(\emptyset)\right)\right) - 2 \cdot r_{\epsilon,2}(f).$$

Proof: From Topsøe [Top01] we know that, if $0 \leq p \leq 1$, then,

$$4p(1-p) \leq h(p) \leq 4p(1-p)^{\frac{1}{\ln 4}},$$

and equivalently, if $X \in \Omega$,

$$\mathbf{Var}[X] \leq H(X) \leq \mathbf{Var}[X]^{\frac{1}{\ln 4}},$$

since

$$\begin{aligned} H(Y|\mathbf{X}) &= H(f(N_\epsilon(\mathbf{X})|\mathbf{X}) \\ &= \mathbf{E}\left[H(f(N_\epsilon(\mathbf{x})|\mathbf{X} = \mathbf{x})\right] \\ &= \mathbf{E}\left[H(f(N_\epsilon(\mathbf{x}))\right]. \end{aligned}$$

Thus, we are interested in $\mathbf{E}[\mathbf{Var}[f(N_\epsilon(\mathbf{x})]]$. For a fixed $\mathbf{x}$ we get

$$\mathbf{Var}[f(N_\epsilon(\mathbf{x})] = 1 - \mathbf{E}\left[f(N_\epsilon(\mathbf{x})\right]^2,$$

where

$$\begin{aligned} \mathbf{E}[f(N_\epsilon(\mathbf{x})] &= \sum_U \hat{f}(U)\mathbf{E}\left[\chi_U(N_\epsilon(\mathbf{x}))\right] \\ &= \sum_U \hat{f}(U) \prod_{i \in U} \mathbf{E}\left[\chi_U(N_{\epsilon,i}(x_i))\right] \\ &= \sum_U \hat{f}(U) \prod_{i \in U} (1-2\epsilon)^{|U|}\chi_U(x). \end{aligned}$$

This leads us to

$$\begin{aligned} \mathbf{E}[\mathbf{Var}[f(N_\epsilon(\mathbf{x})]] &= 1 - \mathbf{E}\left[\mathbf{E}\left[f(N_\epsilon(\mathbf{x})\right]\right]^2 \\ &= 1 - \mathbf{E}\left[\hat{f}(U) \prod_{i \in U} (1-2\epsilon)^{|U|}\chi_U(x)\right]^2. \end{aligned}$$

■

Thus, for any given $r_{\epsilon,2}(f)$ we can give a corridor using these upper and lower bounds, in which the MI of a BF is located.

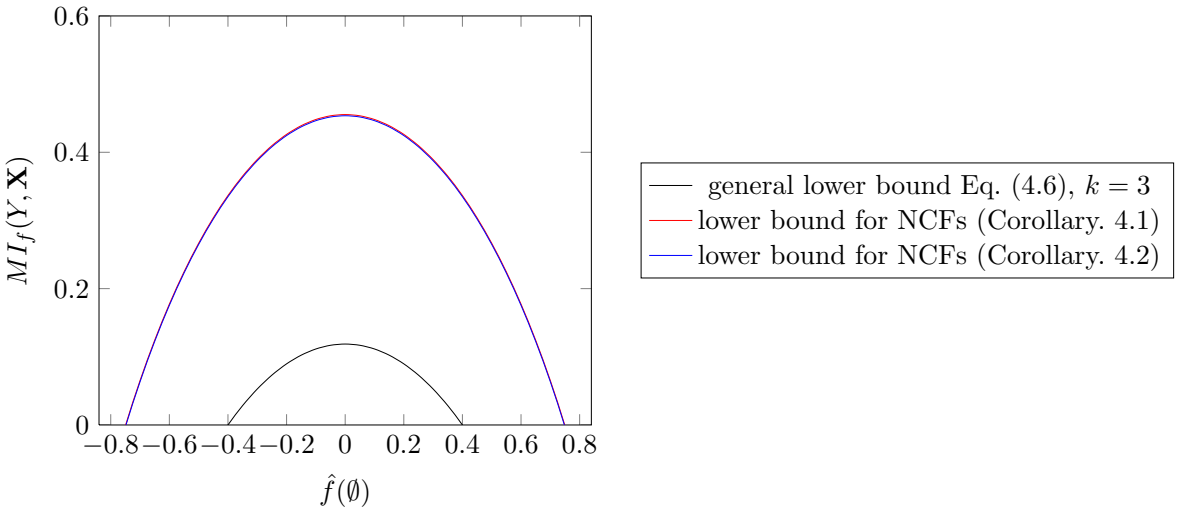


Figure 4.6.: Lower bounds for the MI of BFs, $\epsilon = 0.1$.

4.3.2. Bounds on the Mutual Information for Specific Functions

Using the relations from the previous section we can now derive some bounds on the noisy MI for different classes of BFs. In Theorem 3.2 we derived an upper bound on $r_{\epsilon,c}$ for NCFs. Let us now combine this bound with Propositions 4.2 and 4.3 to obtain the following two corollaries.

Corollary 4.1

Given the system of Definition 4.1, and let f be an NCF with uniformly distributed input variables, then the MI is lower-bounded by:

$$MI_f(Y, \mathbf{X}) \geq h\left(\frac{1}{2}\left(1 + \hat{f}(\emptyset)\right)\right) - h\left(\frac{4\epsilon}{3 + 2\epsilon - \epsilon^2}\right).$$

Corollary 4.2

Given the system of Definition 4.1, and let f be an NCF with uniformly distributed input variables, then the MI is lower-bounded by:

$$MI_f(Y, \mathbf{X}) \geq h\left(\frac{1}{2}\left(1 + \hat{f}(\emptyset)\right)\right) - \frac{16\epsilon(1-\epsilon)}{3 + 4\epsilon - 8\epsilon^2 + 8\epsilon^3 + 4\epsilon^4}.$$

Figure 4.6 depicts the bounds discussed so far for $\epsilon = 0.1$. It becomes visible that the lower bounds for NCFs are much higher than the general bound. Although, we did not prove the tightness of the general bound, this indicates a higher MI of NCFs. Further, we see that the bounds from Corollaries 4.1 and 4.2 are almost equivalent.

Next, let us take the bias into account, when estimating the conditional entropy. Combining Proposition 3.3 with Proposition 4.3 leads us to the following corollary, giving us a general lower and upper bound on the MI for all BF.

Corollary 4.3

Given the system of Definition 4.1, and let f be a BF with k uniformly distributed and relevant input variables with a fixed zero coefficient $\hat{f}(\emptyset)$, then the MI is bounded with:

$$h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)\right)\right) - \left(1-\hat{f}^2(\emptyset)-(1-\hat{f}^2(\emptyset))\cdot\rho^{2\cdot k}\right)^{\frac{1}{\ln 4}} \leq MI(Y,\mathbf{X}) \leq h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)\right)\right) - \left(1-\hat{f}^2(\emptyset)-(1-\hat{f}^2(\emptyset))\cdot\rho^{2}\right),$$

which is equivalent to

$$h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)\right)\right) - \left(\mathbf{Var}\left[f\right]\left(1-\rho^{2\cdot k}\right)\right)^{\frac{1}{\ln 4}} \leq MI(Y,\mathbf{X}) \leq h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)\right)\right) - \mathbf{Var}\left[f\right]\left(1-\rho^{2}\right),.$$

Please note that according to [Top01] the entropy of the function's output can be further bounded by $h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)\right)\right) \leq \mathbf{Var}\left[f\right]^{\frac{1}{\ln 4}}$, which leads us to

$$MI(Y,\mathbf{X}) \leq \mathbf{Var}\left[f\right]^{\frac{1}{\ln 4}} - \mathbf{Var}\left[f\right]\left(1-\rho^{2}\right).$$

Next, let us investigate CFs. In Proposition 3.4 we gave an upper bound on $r_{\epsilon,c}$ for CFs. Again, we can apply this to the lower bound in Proposition 4.3 to obtain the following lower bound on the MI of CFs.

Corollary 4.4

Given the system of Definition 4.1, and let f be a BFs with k uniformly distributed input variables and a zero coefficient of $\hat{f}(\emptyset)$, which is canalizing in some variable j, then the $MI(Y,\mathbf{X})$ is lower-bounded as follows:

$$MI(Y,\mathbf{X}) \geq h\left(\frac{1}{2}\left(1+\hat{f}(\emptyset)\right)\right) - \left(1-\hat{f}^2(\emptyset)-\left(1-|\hat{f}(\emptyset)|\right)^2\cdot\rho^2-\left(2|\hat{f}(\emptyset)|-2\hat{f}^2(\emptyset)\right)\cdot\rho^{2\cdot k}\right)^{\frac{1}{\ln 4}}.$$

Finally, let us examine NCFs. Therefore, we combine Proposition 3.6 with Proposition 4.3 to obtain the following relation:

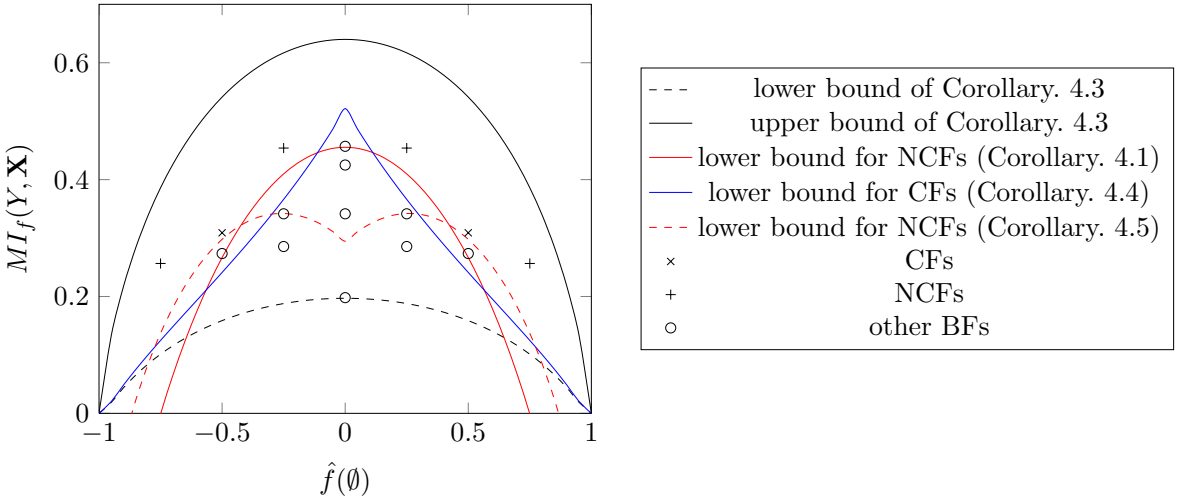


Figure 4.7.: Bounds on the MI of BFs, $k = 3$, $\epsilon = 0.1$.

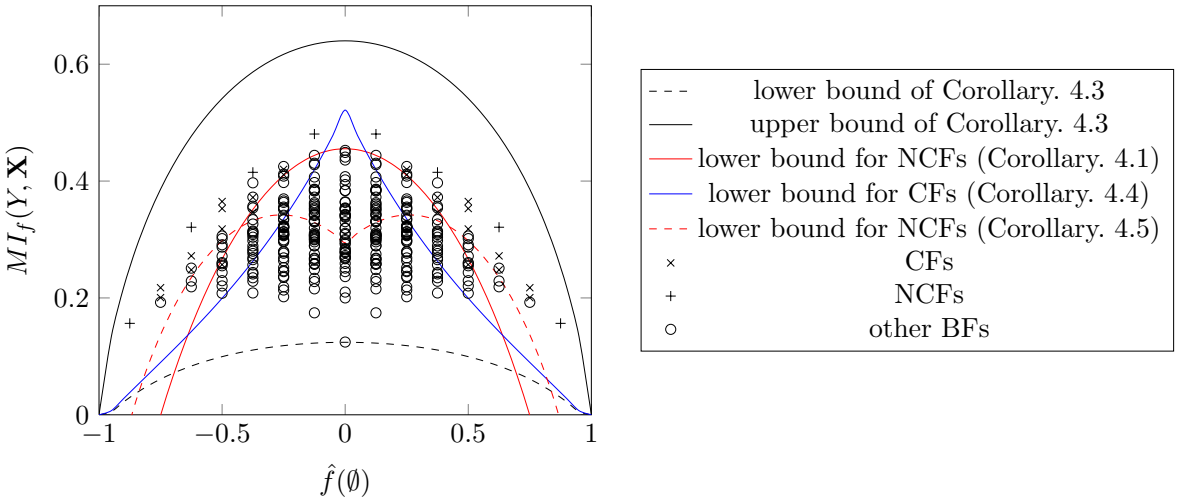


Figure 4.8.: Bounds on the MI of BFs, $k = 4$, $\epsilon = 0.1$.

Corollary 4.5

Given the system of Definition 4.1, and let f be an NCF with a fixed bias of $\hat{f}(\emptyset)$, then the mutual information $MI(Y, \mathbf{X})$ is bounded as

$$MI(Y, \mathbf{X}) \geq h\left(\frac{1}{2}\left(1 + \hat{f}(\emptyset)\right)\right) - \left(2|\hat{f}(\emptyset)|\left(1 - \frac{2}{1+\rho^2}\right) + 2\left(1 + \frac{2}{1+\rho^2} + \frac{2}{-3+\rho^2} - \frac{6}{5+\rho^2}\right)\right)^{\frac{1}{\ln 4}},$$

with $\rho = 1 - 2\epsilon$.

The results of this subsection are summarized in Figures 4.7 and 4.8 for BFs with $k = 3$ and $k = 4$, respectively. We included the actual values of the MI of all BFs, splitted into

CFs, NCFs and the remaining functions. One can see that the lower bound for NCFs gives a higher MI than the one for CFs, while the general lower bound is the lowest. This leads us to the conjecture that in general the CFs have a higher MI than non-canalizing functions, and, in particular, NCFs have a very high MI. Actually, it follows from the figures that for a fixed bias the MI is almost always maximized by either a CF or an NCF. This coincides with the findings for the non-noisy MI, which is maximized by CFs. It seems that for a fixed bias maximizing the MI is equivalent to minimizing the NS and, hence, finding a BF with a low *Fourier concentration*. One can see that Eq. (3.5) is minimized by shifting as much energy to the Fourier coefficients with low order, while still obtaining a valid BF (not all Fourier coefficients which satisfy $\sum_U \hat{f}(U)^2 = 1$ are valid BFs). We conjecture that the class of NCFs, contains the BFs with the lowest possible Fourier concentration.

4.3.3. Mutual Information of some Special Classes of Nested Canalizing Functions

To conclude our investigations of the MI of BFs, let us examine some specific subclasses of NCFs, namley the previously discussed NCF-MDVs, NCF-ACVs, and NCF-SMDs.

In Subsection 3.2.7 we derived exact expressions for $r_{\epsilon,c}$ for those classes. Hence, we can use the bounds of Proposition 4.3 to estimate the MI, which leads us to the following three corollaries:

Corollary 4.6

Let f be a $\{\pi : A : B\}$-NCF-MDV with k uniformly distributed input variables, then the MI of the system from Definition 4.1 is bounded by:

$$h\left(1-2^{-k}\right) - \left(2^{-k+2}\left(1-(1-2(1-\epsilon)\epsilon)^k\right)\right)^{\frac{1}{\ln 4}} \leq$$
$$MI(Y,\mathbf{X}) \leq h\left(1-2^{-k}\right) - 2^{-k+2}\left(1-(1-2(1-\epsilon)\epsilon)^k\right).$$

Proof: The result follows from Propositions 3.7, 4.3 and Corollary 2.3. ■

Corollary 4.7

Let f be a $\{\pi : A : B\}$-NCF-ACV under uniform distribution, then the MI of the system from Definition 4.1 is bounded by:

$$h\left(\frac{(-1)^k}{3\cdot 2^k}+\frac{2}{3}\right) - \left(\frac{4}{3}\tau\left(\frac{4}{1+2\tau}-\frac{(-2)^{-k}}{1-\tau}+\frac{3\cdot 2^{-k}(1-2\tau)^{k+1}}{1+\tau(1-2\tau)}\right)\right)^{\frac{1}{\ln 4}} \leq$$
$$MI(Y,\mathbf{X}) \leq h\left(\frac{(-1)^k}{3\cdot 2^k}+\frac{2}{3}\right) - \frac{4}{3}\tau\left(\frac{4}{1+2\tau}-\frac{(-2)^{-k}}{1-\tau}+\frac{3\cdot 2^{-k}(1-2\tau)^{k+1}}{1+\tau(1-2\tau)}\right),$$

where $k = rel(f)$ is the number of relevant variables of f and $\tau = (1-\epsilon)\cdot\epsilon$. In the case of an infinite number of input variables we obtain:

$$h\left(\frac{2}{3}\right) - \left(\frac{16\tau}{3+6\tau}\right)^{\frac{1}{\ln 4}} \leq MI(Y,\mathbf{X}) \leq h\left(\frac{2}{3}\right) - \frac{16\tau}{3+6\tau}.$$

Proof: The result follows from Propositions 3.8, 4.3 and Corollaries 3.11 and 2.4. ■

Corollary 4.8

Let f be a $\{\pi : A : B\}$-NCF-SMD with k uniformly distributed relevant variables, then

$$h\left(\frac{1}{2}\left(1+\frac{1}{2^{k-1}}\right)\right)-\left(2^{-k+2}\left(1-4\tau+2^k\tau-(1-2\tau)^k\right)\right)^{\frac{1}{\ln 4}} \leq$$
$$MI(Y,\mathbf{X}) \leq h\left(\frac{1}{2}\left(1+\frac{1}{2^{k-1}})\right)\right)-2^{-k+2}\left(1-4\tau+2^k\tau-(1-2\tau)^k\right),$$

where $\tau=(1-\epsilon)\cdot\epsilon$. In the case of an infinite number of input variables we obtain:

$$1-(4\tau)^{\frac{1}{\ln 4}} \leq MI(Y,\mathbf{X}) \leq 1-4\tau.$$

Proof: The result follows from Propositions 3.9, 4.3 and Corollaries 3.12 and 2.5. ■

In Figure 4.9 we plotted the results from these corollaries for $k=3$ and $k=6$, respectively. It becomes visible that, equivalent to our findings regarding the NS, NCF-SMDs perform best, followed by NCF-ACVs, while NCF-MDVs have the lowest MI. In particular, one can see that the MI of NCF-MDVs becomes smaller and tends to zero with increasing number of relevant variables. This can be explained by looking at the absolute values of the zero coefficient of these functions, which is given by (see Corollary 2.3)

$$|\hat{f}(\emptyset)| = 1-2^{-(k-1)}.$$

For growing k this value tends to one, hence, the functions get closer to the constant function, hence the MI tends to zero.

Further, the MI of NCF-ACVs remains - again as their bias - relatively stable, while the MI of NFC-SMDs grows. This is also reflected in Figure 4.10, where we plotted the asymptotic results. Interestingly, if we use Corollary 4.1 and the asymptotic NS of NCF-SMDs (Corollary 3.12) to obtain an alternative lower bound, we get:

$$MI(Y,\mathbf{X}) = 1-h(\epsilon),$$

which coincides with a conjectured upper bound on the MI of BFs in general by the authors of [KC13]. Hence, if this conjecture holds, NCF-SMDs asymptotically maximize the MI.

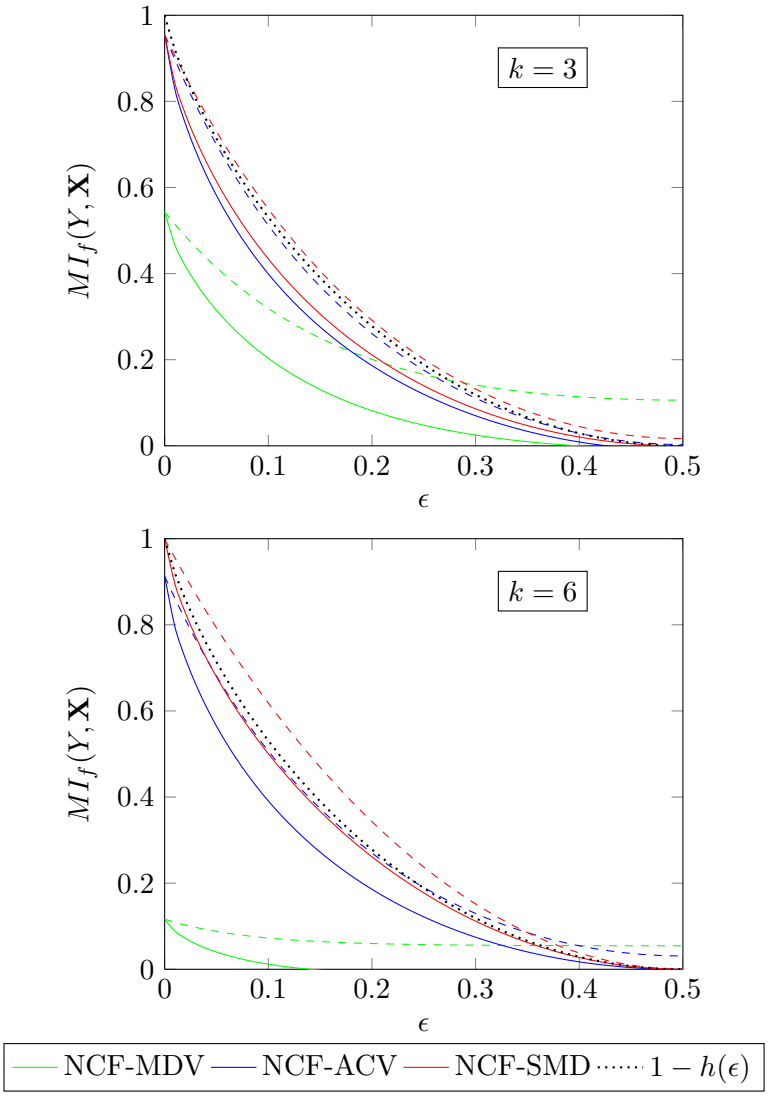


Figure 4.9.: Upper (dashed) and lower (solid) bounds on the MI (see Figure 4.5) for NCF-MDVs, NCF-ACVs and NCF-SMDs witk $k = 3$ and $k = 6$ relevant variables, respectivitly, compared to the conjectured upper bound of [KC13].

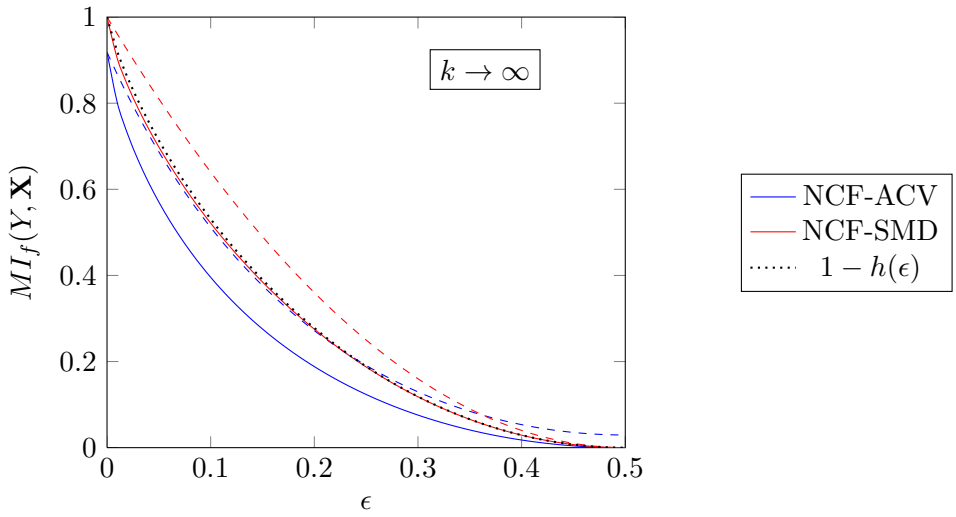


Figure 4.10.: Upper (dashed) and lower (solid) bounds on the MI (see Figure 4.5) for NCF-ACVs and NCF-SMDs witk $k \to \infty$ relevant variables.

Chapter 5

The Preimage Problem of Boolean Networks

Boolean networks for biological applications have been first studied by Kauffman in the late 1960s [Kau69b, Kau69a]. He investigated random Boolean networks to model large scale regulatory control networks. In a cell complex chemical reactions and reaction chains take place in order to fulfill the task of the cell. These reactions and their interactions can be represented by using the so-called metabolic flux model [FHR$^+$07]. Some of the reactions are regulated by one or more proteins, i.e., the reactions need certain proteins to take place. These proteins are coded in the genome of the cell, i.e., a certain gene has to be expressed in order to obtain a protein. Which genes are expressed and which not is again regulated. Regulative factors are external environmental conditions, such as the presence of oxygen or the temperature, external metabolites and other proteins. These influences, which can be inhibiting or activating, can be represented in a Boolean regulative network. Using this model and given a set of variables describing the environmental conditions and the presence of external metabolites, we can directly get a list of possibly expressed genes by evaluating the network. However, it is also of interest to reverse that problem: To check, if a set of fluxes, which is necessary to perform a certain task or to produce a certain side product, is feasible, the Boolean network needs to be inverted. The aim is to get a set of possible environmental variables, which lead to the expression of desired genes. This so-called *predecessor problem* or *preimage problem* has been addressed by Wuensche in [Wue94] and has been shown to be NP-hard in general [AHZ$^+$09], which makes it infeasible to solve for large networks.

Besides randomly generated networks, we will examine a large-scale Boolean model of the transcriptional network of *Escherichia coli*, a well-known and well-studied gut bacteria. We extended the network model of the transcriptional network of *E. coli* (Covert et al. [CKR$^+$04]) by mapping genes to their corresponding fluxes in the flux-balance model presented by [FHR$^+$07]. The network has a layered feed-forward structure and shows characteristic topological features, such as a long-tail like out-degree distribution.

In this chapter, we will first formally introduce Boolean networks, and then in Section 5.2 we will address the preimage problem and propose two algorithms, one for networks consisting mainly of canalizing functions and another based on the well-known sum-product algorithm.

5.1. Fundamentals of Boolean Networks

A Boolean network (BN) can be represented as a directed graph consisting of a set of K nodes connected by edges. Each node $i \in [K]$ has a binary state, i.e., it is turned either *on*

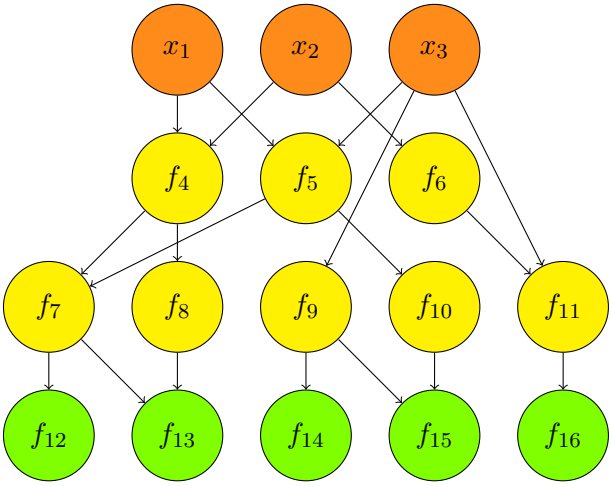


Figure 5.1.: Example of a Feed-Forward Network.

or *off*. In terms of biological interpretation, this means that the gene is either expressed or not. We use values from Ω to represent these states. *On* is denoted by -1, *off* by $+1$. An edge between two nodes implicates that one node has a controlling influence on the other node. As we are only considering feed-forward networks, i.e., networks without feedback loops, there is a set of nodes $\mathbb{I}$, which have no incoming edges. These so-called input-nodes are the inputs to the network and may represent the environmental conditions. Consequently, there exists also a set of nodes $\mathbb{O}$, which have no outgoing edges. These nodes contain the outcome or result of the network and are therefore called output-nodes.

Each node except the input-nodes of this graph $i \in [K] \setminus \mathbb{I}$ contains a BF f_i to determine the actual state x_i based on the states of the nodes located on the other side of the incoming edges, i.e., an edge from node $j \in [K]$ to node i indicates that the state of node i is an input to the BF of node j. In this case a node i is called a *controlling* node of j. Obviously the number of edges leading to a node is equal to the in-degree n_i of its function f_i. The number of edges emerging from a node is called out-degree m_i. Input-nodes have an in-degree of zero, while output-nodes have an out-degree of zero. In Figure 5.1 we depicted an exemplary network with $K = 16$ nodes. It consists of $N = 3$ input-nodes $\mathbb{I} = \{1, 2, 3\}$ and $M = 5$ output-nodes $\mathbb{O} = \{12, 13, 14, 15, 16\}$.

Further, let's define the set $\tilde{n}(f_j)$ as the incoming nodes of node j. For example in Figure 5.1, $\tilde{n}(f_5) = \{1, 3\}$.

Due to the fact that we are limiting ourselves to feed-forward networks, after a certain number l of synchronous updates, the states of the network, in particular the output nodes willnot change anymore, if we let the input nodes unchanged. An update means that each node updates its one value by using its BF and the values stored in the nodes, located at the end of the incoming edges. With other words, if we set the input nodes to certain states, the *outcome* of the BN, if formed by the states of the output-nodes after l updates. l can also be seen as the maximum number of *hops* between an input and an output node. In this context one often speaks of a layered network, where $\tilde{l} = l + 1$ is the number of layers, i.e., we can partition the BN into layers $L_1, L_2, \ldots, L_l, L_{\tilde{l}}$. If a node i is an element of layer L_h, all controlling nodes are elements of layer L_m with $m < h$. The first (highest) layer L_1 consists of the input-nodes, while the lowest layer $L_{\tilde{l}}$ consists of the output-nodes.

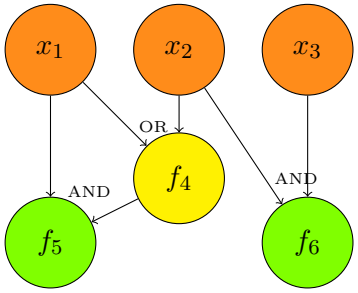


Figure 5.2.: Small exemplary Boolean Network

Hence, we can represent such BNs as functions mapping the N input values uniquely to the M output values:

$$\mathbf{f} : \Omega^N \to \Omega^M.$$

Given a vector of input values $\mathbf{x} \in \Omega^N, \mathbf{x} = (x_1, x_2, \ldots, x_N)$, the corresponding output of $\mathbf{f}$ is $\mathbf{y} = \mathbf{f}(\mathbf{x}), \mathbf{y} \in \Omega^M$.

5.2. The Preimage Problem

As mentioned before we can interpret a BN as a function mapping the N input values uniquely to the M output values:

$$\mathbf{f} : \Omega^N \to \Omega^M.$$

We are interested in finding an inverse to this function, i.e., the sets of input-values $\mathbf{x}$ which lead to a specific output y. However, in general there does not exist a unique inverse function $\mathbf{f}^{-1}$. Instead the cardinality of the set

$$\Psi_{\mathbf{y}} := \{\mathbf{x} : \mathbf{f}(\mathbf{x}) = \mathbf{y}\}$$

will be larger one. We call $\Psi_{\mathbf{y}}$ the set of preimages of $\mathbf{y}$. Consequently, the problem of finding this set is called *preimage problem* or *predecessor problem* [Wue94].

Before introducing our proposed algorithms, we will first describe the general solution for the preimage problem according to Wuensche [Wue94].

5.2.1. Wuensche Algorithm [Wue94]

First, we create a global matrix where each of the K columns represents a node of our network. A row represents a possible solution vector $\mathbf{x}$ of the solution set $\Psi_{\mathbf{y}}$ for a determined $\mathbf{y}$. The entries of this matrix can be either -1, $+1$ or a wildcard $*$. The wildcard means that this input-node has no relevance within this solution vector.

For initialization only one row of this matrix is created, containing the desired output values $\mathbf{y}$ and wildcards for all other nodes.

Example 5.1 (*Initialized global matrix for the BN of Figure 5.2*)

For the exemplary network of Figure 5.2 and $\mathbf{y} = (-1, +1)$ *the initialized vector would look as follows:*

x_1	x_2	x_3	f_4	f_5	f_6
$*$	$*$	$*$	$*$	-1	$+1$

Now we traverse the graph and perform the following two steps for each node containing a Boolean function.

Step 1: Create a local copy of the global matrix and fill it with rows according to the input leading to the desired output of the function. If the node itself is marked as unknown or both values are possible in the current global matrix, we add all rows of input values leading to the output of -1 and all rows leading to an output of $+1$. The output itself is also added to the rows. Columns representing nodes, which have no influence on the output, are set to the wildcard.

Step 2: In this step the local matrix is merged with the global matrix to receive a new global one. Therefore, each row of the global matrix is compared with each row of the local one. If two rows are equal or only differ at columns, where one matrix contains wildcards, the row is added to the new global matrix. All wildcards are set to known values, if possible. If the resulting global matrix is empty, there is no solution for this inverse problem.

Example 5.2 (*Update of global matrix for the BN of Figure 5.2*)

Let us continue with the BN from the previous example and perform steps 1 and 2 for the BFs of this network. We start with f_4*, which is an OR function. Let us create a local matrix for* f_4*. As the column representing* f_4 *in the global matrix contains the wildcard, we basically need to add the whole truth-table of* f_4 *to our local matrix, which is hence given as:*

x_1	x_2	x_3	f_4	f_5	f_6
$+1$	$+1$	$*$	$+1$	$*$	$*$
$+1$	-1	$*$	-1	$*$	$*$
-1	$+1$	$*$	-1	$*$	$*$
-1	-1	$*$	-1	$*$	$*$

As the global matrix only contains non-wildcard values for f_5 *and* f_6 *and the local matrix contains only wildcards for these nodes, the merging (step 2) is straight forward and results in the new global matrix:*

x_1	x_2	x_3	f_4	f_5	f_6
$+1$	$+1$	$*$	$+1$	-1	$+1$
$+1$	-1	$*$	-1	-1	$+1$
-1	$+1$	$*$	-1	-1	$+1$
-1	-1	$*$	-1	-1	$+1$

Next, we examine f_5, which is an AND function. As the output of f_5 is supposed to be always -1 we only need to add the corresponding row to the local matrix:

x_1	x_2	x_3	f_4	f_5	f_6
-1	$*$	$*$	-1	-1	$*$

Now, we will merge this row with the global matrix. One can see that there is a contradiction between the row of the local matrix and rows 1 and 2 of the global matrix. Hence, we will delete this rows, and merge the remaining. This leads to a new global matrix:

x_1	x_2	x_3	f_4	f_5	f_6
-1	$+1$	$*$	-1	-1	$+1$
-1	-1	$*$	-1	-1	$+1$

Finally, we perform these two steps for f_6 which leads to the final global matrix:

x_1	x_2	x_3	f_4	f_5	f_6
-1	$+1$	$+1$	-1	-1	$+1$
-1	$+1$	-1	-1	-1	$+1$
-1	-1	$+1$	-1	-1	$+1$

This matrix is the whole preimage $\Psi_{\mathbf{y}}$ of the BN of Figure 5.2 for $\mathbf{y} = (-1, +1)$.

After repeating the two steps for all nodes with functions attached, there may still be columns with wildcards in the global matrix. These wildcards then have to be replaced by both possible values, i.e. Ω, and for each one, a new row has to be created. Finally, the global matrix contains the set of all input vectors $\mathbf{x}$ leading to the desired output.

As discussed before, the Wuensche algorithm itself has exponential complexity, so, if this algorithm is applied to the whole network of size K, the complexity is thus enormous that the problem becomes infeasible. However, the algorithm proposed in the next subsection will show us a way to divide the network into smaller subnets, which can then be solved independently.

5.2.2. Approach for Networks with Canalizing Functions

In this subsection we will provide an algorithm, which gives us the complete solution set $\Psi_{\mathbf{y}}$, like the algorithm in the previous subsection. Our algorithm uses the divide and conquer method and makes use of the properties of canalizing functions, which allow us to invert some Boolean functions and hence reduce the size and the in-degrees of the BN. The BN is also split into much smaller subnets. The preimages of these subnets can then be obtained using the Wuensche algorithm.

The remainder of this subsection is organized as follows: first we take a look at the possibility of inverting canalizing functions, before we discuss the method to reduce the size of the network and to split the network into subnets. After that, we will present the complete algorithm and briefly discuss the computational complexity.

5.2.2.1. Inverting Canalizing functions

As discussed in the previous chapters, canalizing and nested canalizing functions have some important properties. Another advantage of these class of functions is that they are invertible under certain conditions.

Proposition 5.1

Assume a BF f is canalizing for input variable i with canalizing value a_i and canalized value b_i. If the output of the function is $f(\mathbf{x}) = -b_i$, then the i-th input must be $x_i = -a_i$ if $y = -b_i$.

Proof: The proof follows directly from the definition of canalizing functions. ■

For nested canalizing functions even more variables can be determined by this method. Also note that functions with in-degree one are also canalizing and hence can always be inverted. We will give a short example to clearify Proposition 5.1.

Example 5.3

Let f be the AND function with two input variables x_1 and x_2, i.e., $f(\mathbf{x}) = x_1$ AND x_2. Obviously $f(\mathbf{x})$ is canalizing for $x_1 = +1$, since $f^{(1,+1)}(\mathbf{x})$ is always $+1$ independent from the value of x_2. For the same reason $f(\mathbf{x})$ is also canalizing for $x_2 = +1$.
If the output y of $f(\mathbf{x})$ is now supposed to be $y = +1$, we can now state from Proposition 5.1 that x_1 and x_2 must be both equal $+1$.

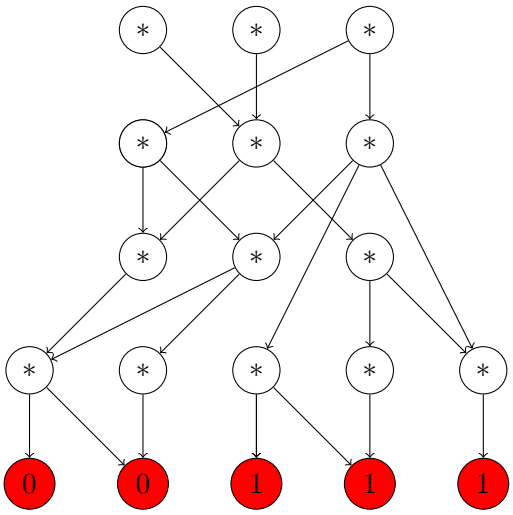
5.2.2.2. Reducing the Size of the BN and Splitting the BN into Subnets

The complete method to reduce the network is listed in Algorithm 1 on page 77 and is illustrated for a small example in Figures 5.3 and 5.4. It will be described here briefly. Let us define a vector $\mathbf{z}$ containing the values of all nodes in the network. All entries are set to a wildcard $*$, i.e., they are marked as unknown, except the out-nodes $\mathbf{y}$, which shall be located at the end of the vector $\mathbf{z}$, i.e., $\mathbf{y} = (y_1, y_2, \ldots, y_M) = (z_{K-M}, z_{K-M+1}, \ldots, z_K)$ (see Figure 5.3(a)).

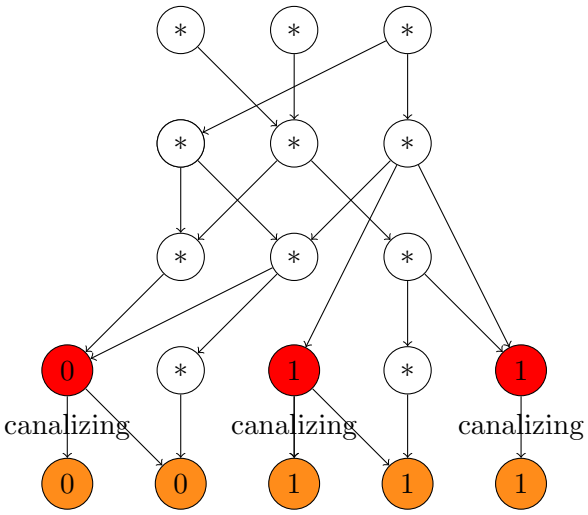
Now, let's define a set $\mathcal{C}$, which contains all fixed nodes. In this set we will store all nodes, for which the actual value is known. Initially, this set only contains the values of $\mathbf{y}$, hence $\mathcal{C} = \{(i, z_i) : i = K - M \ldots K\}$.

Now, this set is stored to $\mathcal{C}_{store}$ for later comparison. Then, for each fixed tuple $(i, z_i) \in \mathcal{C}$ we check, if f_i is canalizing towards z_i using Theorem 2.2. If the function is canalizing, then let z_j be the canalizing variable and we invert the function, if possible, using Proposition 5.1 (see Figure 5.3(b)). If the resulting input value z_j is already known and differs from the calculated one, an error is produced. This means that this Boolean network is not invertible for the given output values.

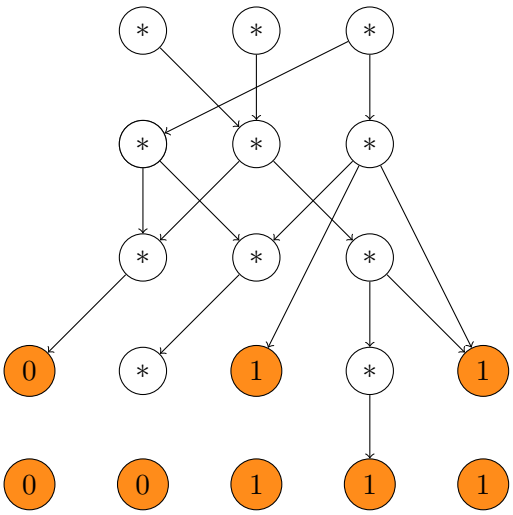
If no error occurs, the new tuple is added to $\mathcal{C}$ and the values can now be fed forward to the nodes attached to z_j, and, since this value is fixed now, the functions can be restricted to this fixed value. Hence, the in-degree of these functions are reduced and the new function can be calculated using Theorem 2.1. After that, the edges can be removed from the graph as they do not represent an influence anymore (Figure 5.3(c)). The new functions received by the restricting operation may now be canalizing so that the described methods can be



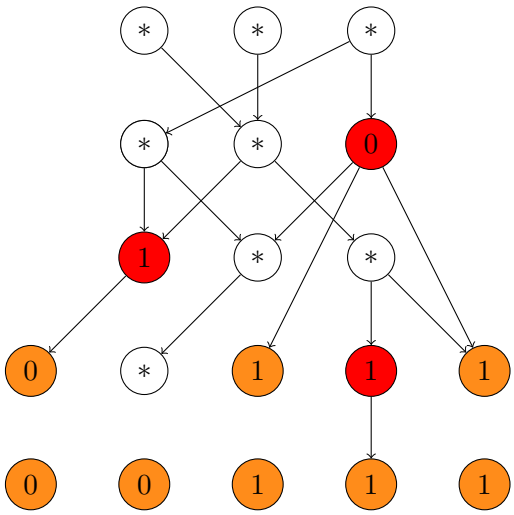
(a) **Initialization of the Network:** All nodes are set to wildcards, except for the output nodes (red), which are set to the desired output, for which the predecessors should be found.



(b) **Inverting canalizing functions:** The canalizing functions are identified and inverted, if possible. The inputs are fed back (red nodes).

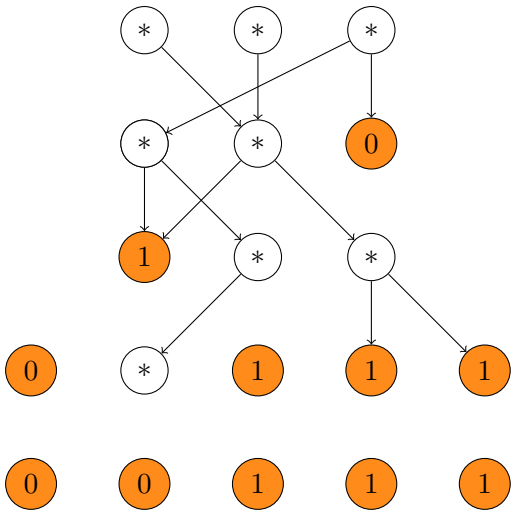


(c) **Feed Forward:** The values of recovered nodes are fed forward and the nodes are added to the constrained sets (orange nodes). Further, the functions are restricted and corresponding edges are removed.

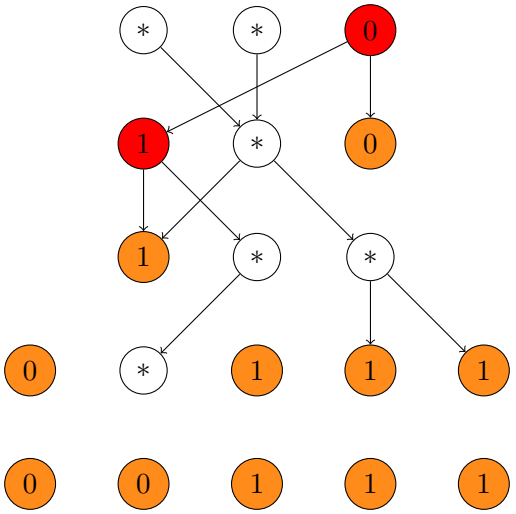


(d) **Repetiton I:** Steps (b) and (c) are repeated until there is no change in the constrained set.

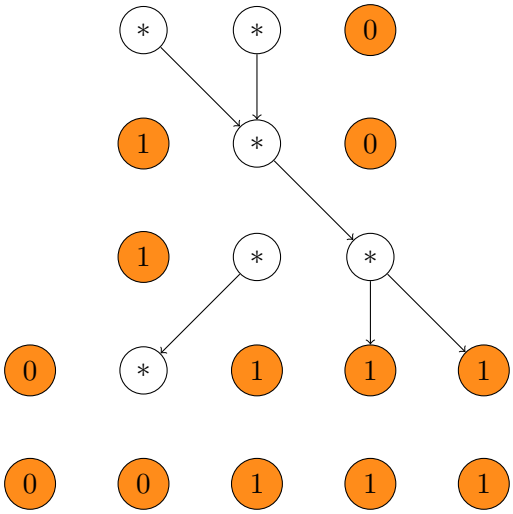
Figure 5.3.: Small example for Algorithm 1, Part I



(a) **Repetiton II:** Steps (b) and (c) are repeated until there is no change in constrained set.



(b) **Repetiton III:** Steps (b) and (c) are repeated until there is no change in constrained set.



(c) **End of Algorithm:** At the termination of the algorithm the graph has fallen apart into independent subgraphs.

Figure 5.4.: Small example for Algorithm 1, Part II

Algorithm 1

Let $\mathbf{y} = \{y_1, y_2, \ldots, y_M\} = \{z_{K-M}, z_{K-M+1}, \ldots, z_K\}$ be the set of initial determined nodes.
Let $\mathcal{C} = \{(i, z_i) : i = K - M \ldots K\}$ be the constraint set of nodes already fixed.
repeat
  Store constraint set $\mathcal{C}_{store} \leftarrow \mathcal{C}$
  for each $(i, b_i) \in \mathcal{C}$ **do**
    Calculate first order Fourier coefficients for f_i
    if f_i is canalizing towards $-b_i$ **then**
      Let j be the canalizing variable of f_i, z_j the value of the corresponding node and a_j the canalizing value, i.e. $f_i^{(j,a_j)} = -b_i$
      if z_j is already fixed, i.e., $(j, b_j) \in \mathcal{C}$ and $b_j \neq -a_j$ **then**
        Return with error, no solution possible
      else
        Set z_j to $-a_j$ and add to constraint set $\mathcal{C} = \mathcal{C} \cup (j, -a_j)$
        For all nodes k, where z_j is input of function f_k, restrict function ${f_k}^{(j,-a_j)}(\mathbf{x})$ and remove edges between j and k.
      end if
    end if
  end for
until no increase in constrained set happened, i.e., $\mathcal{C}_{store} = \mathcal{C}$

applied iteratively. Therefore, we repeat these steps beginning from storing $\mathcal{C}$, if at least one new tuple has been added to $\mathcal{C}$, i.e., as long as $\mathcal{C} \neq \mathcal{C} \cup (j, -a_j)$ (Figures 5.3(d)-5.4(b)).

Obviously, in these steps each node is only assigned one value. At the end of this process there might be a number of nodes, whose value are still set to the wildcard and, hence, unknown. However, the values found in these steps are unique, i.e., they remain constant for all solutions in the solution space $\Psi_{\mathbf{y}}$.

Due to the removal of edges and the growing number of known edges, the network will split itself into independent subnets (Figure 5.4(c)). These subnets are smaller than the original one. Due to that fact, we now can apply the Wuensche algorithm, which may now be feasible.

5.2.2.3. Complete Algorithm

Let us now give a complete and comprehensive sketch of the algorithm (see Algorithm 2). First, all nodes have to be marked as unknown and the desired output values have to be written into the graph. Then, we repeatedly perform the actions as described in Algorithm 1. As mentioned before, the values found in this step are unique, i.e., they remain constant for all solutions in the solution space $\Psi_{\mathbf{y}}$.

Due to the restrictions on the Boolean functions and the accompanying removing of edges, the graph splits itself up in several subgraphs. For each of these subgraphs the Wuensche algorithm described in Subsection 5.2.1 is performed. The results calculated in this step are not unique anymore, hence several possible input vectors are received, i.e., we receive the complete solution space $\Psi_{\mathbf{y}}$.

5.2.2.4. Complexity analysis

Finally, let us analyze the complexity of this algorithm. Again, we split our investigations into two parts. First, we will analyze the computational complexity of the first step, described in the previous subsection. We are interested in the complexity in dependence of the total number of nodes in the network K and the number of layers l.

Since, in the worst case, Algorithm 1 consists of a run through all nodes in the network, the complexity is growing linear with the number of nodes K. The complexity of the actions performed at each node depends on the in-degree of the function, which we can assume very small in relation to K. The first part is iteratively repeated. The number of the repetitions depend on the number of layers of the network l. Hence the complexity of the first step is linear with respect to l and K.

The problem, which is addressed in Section 5.2.1, is at least NP-hard. We can state this, since as shown in [AHZ+09] the problem of finding one predecessor of a state of a BN is NP-hard, hence finding the complete solution space is at least NP-hard, too. The Wuensche algorithm itself has exponential complexity. However, since in our algorithm some nodes of the network are already inverted in advance and the remaining nodes are finally split into several subnets i with a very reduced number of nodes $K_i << K$, the computational complexity is much smaller and the problem becomes feasible.

5.2.3. Approach using the Sum-Product Algorithm

Now, we will introduce an algorithm, which finds only parts of the set of preimages Ψ_y. However, this algorithm has a inear (with K), hence a much lower, complexity and works independent of canalizing functions. It solves this problem in linear time with respect to the number of nodes in the network and is based on a variation of the well-known sum-product algorithm (SPA) [KFL01], which is used for a variety of tasks, including decoding error correction codes in communication engineering [Gal63, Bos13].

Suppose there is a probability distribution $P_{\mathbf{y}}$ on Ω^N such that

$$P_{\mathbf{y}}\{\mathbf{x}\} = \begin{cases} \frac{1}{|\Psi_{\mathbf{y}}|} & \text{if } \mathbf{x} \in \Psi_{\mathbf{y}} \\ 0 & \text{else} \end{cases}.$$

If we knew the probability distribution $P_{\mathbf{y}}$, we could solve the problem, as we could draw input vectors $\mathbf{x}$ according to $P_{\mathbf{y}}\{\mathbf{x}\}$. Each of these $\mathbf{x}$ would hence be part of $\Psi_{\mathbf{y}}$. Our main idea now is to approximate $P_{\mathbf{y}}$ with the product of the marginal distributions P_i on the

Algorithm 2

For all nodes in the graph, set values to wildcard, i.e., mark nodes as unknown
Set output values to the desired values $\mathbf{y}$
Perform Algorithm 1
{The graph is now split into several independent subgraphs}
for each subgraph **do**
  solve inversion problem locally with Wuensche Algorithm (Subsection 5.2.1)
end for

individual x_i, i.e.,

$$P_{\mathbf{y}} \approx \prod_{i=1}^{N} P_i,$$

as the well-known SPA can be used to compute the marginals efficiently. If the approximation is *good enough*, sampling out the product of the marginals will yield an element in $\Psi_{\mathbf{y}}$ with reasonable probability. Thus, by sampling, we may obtain a subset $\widetilde{\Psi}_{\mathbf{y}} \subseteq \Psi_{\mathbf{y}}$.

We will first discuss the basic principles of factor graphs and the SPA. Then we will describe the BN as factor graph and will formulate the actual algorithm to find the marginals. Finally, the sampling itself is described.

5.2.3.1. Factor Graphs and Sum-Product Algorithm

Assume some function $g(x_1, \ldots, x_n)$, defined on some domain $\mathbf{A}^n$, which can be factorized in m *local* functions $h_j, j \in [m]$, i.e.,

$$g(x_1, \ldots, x_n) = \prod_j h_j(X_j),$$

where X_j is the subset of $[n]$ containing the argument of h_j. We can then define a factor graph [KFL01] as a bipartite graph consisting of n nodes representing the variables $\{x_1, \ldots, x_n\}$ (variable nodes) and of m nodes representing functions $\{h_1, \ldots h_m\}$ (function nodes). An Edge exists between a function node and a variable node, if and only if x_i is an input to function h_j.

The marginal function $g_i(x_i)$ is defined as [KFL01]

$$g_i(x_i) = \sum_{\sim\{x_i\}} g(x_1, \ldots, x_n),$$

where $\sum_{\sim\{x_i\}} g(x_1, \ldots, x_n)$ is defined as

$$\begin{aligned} &\sum_{\sim\{x_i\}} g(x_1, \ldots, x_n) \\ &= \sum_{x_1 \in \mathbf{A}} \cdots \sum_{x_{i-1} \in \mathbf{A}} \sum_{x_{i+1} \in \mathbf{A}} \cdots \sum_{x_n \in \mathbf{A}} g(x_1, \ldots, x_n). \end{aligned}$$

In general the computation of the g_i is difficult, but due to the factorization of g the task can be efficiently solved using the so-called sum-product algorithm [KFL01].

The algorithm iteratively passes *messages* between the nodes of the graph. At each iteration the messages μ are sent from the function nodes to the variable nodes, containing the corresponding marginal function of the local function. These messages are computed as follows [KFL01] (the indices of the functions are omitted):

function to variable node:

$$\mu_{h \to x}(x) = \sum_{\sim\{x\}} \left(h(n(h)) \prod_{y \in n(h) \setminus \{x\}} \lambda_{y \to h}(y) \right),$$

where $n(i)$ give the set of neighboring nodes of node i.

At the variable nodes, these messages are then combined to a marginal function λ and sent back to the function nodes [KFL01]:

variable to function node:

$$\lambda_{x\to h}(x) = \prod_{q\in n(x)\setminus\{h\}} \mu_{q\to x}(x).$$

After a certain number of iterations the messages μ converge to the desired marginal functions $g_i(x_i)$.

5.2.3.2. The Boolean Network as Factor Graph

Next, we apply the concept of factor graphs to BNs. Each node in the network represents one variable $x_i \in \Omega, i \in [K]$ of the factor graph, hence we have K variable nodes. Each BF f_j of the BN ($j \in [K] \setminus \mathbb{I}$) is a function node and is connected to the node j and the incoming nodes $\tilde{n}(f_j)$. Let's define $\widetilde{X}_j$ as the variables of the incoming nodes of node j, i.e., the argument of the BN f_j. Further, we define $\widetilde{X}_j^{(i)}$ as $\widetilde{X}_j$ without the node i.

Finally, if we consider the variables at each node as random variables, we have a common distribution of all variable nodes described by the density function,

$$g_{x_1,\dots,x_n}(x_1,\dots,x_n) \equiv g(x_1,\dots,x_n).$$

For the sake of readability, we will omit the subscripts of the density function, if they are obvious from context.

We are interested in finding the marginal distributions of the in-nodes, which can be described by the density functions

$$g_{x_i}(x_i) = \sum_{\sim x_i} g_{x_1,\dots,x_n}(x_1,\dots,x_n) \quad \forall i \in \mathbb{I}.$$

This problem is an instance of the problem described in Subsection 5.2.3.1, hence we apply the same methods here.

Update Rule: function to variable node

For one function node $j \in [n] \setminus \mathbb{I}$, there exists a common distribution of all variables relevant for this node. Namely, these relevant variables are the ones located in $\widetilde{X}_j$ of the BF f_j and the value of node j. We can write the density of this distribution as:

$$p(x_j, \widetilde{X}_j).$$

Let's define $\tilde{n}(f_j)$ as the set of indices for the input nodes of the BF f_j.

We need to send the local marginal distribution of each variable $i \in \{j\} \cup \tilde{n}(f_j)$ back to the variable node, or more formally:

$$\mu_{j\to i}(x_i) = \sum_{\sim\{x_i\}} p(x_j, \widetilde{X}_j) = \sum_{\sim\{x_i\}} p(x_j, x_i, \widetilde{X}_j^{(i)}). \tag{5.1}$$

If $i = j$, i.e., if the message is designated for the node containing the output of the BF, the density of the marginal distribution becomes:

$$\begin{aligned}\mu_{j\to j}(x_j) &= \sum_{\sim\{x_j\}} p(x_j|\widetilde{X}_j)\cdot p(\widetilde{X}_j)\\ &= \sum_{\sim\{x_j\}} f_j(\widetilde{X}_j)\cdot p(\widetilde{X}_j),\end{aligned}$$

which is the probability distribution of the function's output. We can assume that the elements of $\tilde{X}_j$ are pairwise independent:

$$p(\widetilde{X}_j) = \prod_{o\in\tilde{n}(f_j)} \lambda_o(x_o),$$

where λ_o is the probability distribution of variable node o and will be defined in Eq. 5.3.

In the other cases, i.e., $i \neq j$, Eq. (5.1) becomes:

$$\mu_{j\to i}(x_i) = \sum_{\sim\{x_i\}} p(x_i|x_j,\widetilde{X}_j^{(i)})\cdot p(x_j,\widetilde{X}_j^{(i)}).$$

We still can assume that the elements of $\widetilde{X}_j^{(i)}$ are pairwise independent, hence:

$$\begin{aligned}p(x_j, n(f_j)\setminus x_i) &= p(x_j|\widetilde{X}_j^{(i)})\cdot p(\widetilde{X}_j^{(i)})\\ &= p(x_j|\widetilde{X}_j^{(i)}) \prod_{o\in\tilde{n}(f_j)\setminus\{i\}} \lambda_o(x_o).\end{aligned}$$

If the Boolean function's output $x_j = f_j(\widetilde{X}_j)$ is already completely determined by $\widetilde{X}_j^{(i)}$, i.e., if the variable x_i has no influence on the output for this particular choice of the other variables, we assume x_i to be uniformly distributed:

$$p(x_i|x_j,\widetilde{X}_j^{(i)}) = \frac{1}{2} p_{x_j}(f(\widetilde{X}_j^{(i)}, x_i) = x_j),$$

and since x_j is completely determined by $\widetilde{X}_j^{(i)}$, we can write

$$p(x_j,\widetilde{X}_j^{(i)}) = \prod_{o\in\tilde{n}(f_j)\setminus\{i\}} \lambda_o(x_o).$$

Otherwise, x_i is totally determined by x_j and the other variables, i.e., x_i is $+1$ or -1 depending on the BF. Hence:

$$p(x_i|x_j, n(f_j)\setminus x_i) = p_{x_j}(f(\widetilde{X}_j^{(i)}, x_i) = x_j),$$

where $p_{x_j}(f(\widetilde{X}_j^{(i)}, x_i) = x_j)$ is either $+1$ or -1.

Further, we can assume x_j independent of $\widetilde{X}_j^{(i)}$:

$$p(x_j,\widetilde{X}_j^{(i)}) = \lambda_j(x_j) \prod_{o\in\tilde{n}(f_j)\setminus\{i\}} \lambda_o(x_o).$$

Finally, we can summarize for $i \neq j$:

$$\mu_{j\to i}(x_i) = \sum_{\sim\{x_i\}} \xi_{i,j} p_{x_j}(f(\widetilde{X}_j^{(i)}, x_i) = x_j) \prod_{o\in\tilde{n}(f_j)\setminus\{i\}} \lambda_o(x_o), \tag{5.2}$$

with

$$\xi_{i,j} = \begin{cases} \frac{1}{2} & \text{, if } f_j(\widetilde{X}_j^{(i)}, x_i = +1) = f_j(\widetilde{X}_j^{(i)}, x_i = -1) \\ \lambda_j(x_j) & \text{, else} \end{cases}.$$

Update Rule: variable to function node

The update rule is the same for all variable nodes $j \in [n]$ and is independent of the function node, to which they are directed:

$$\lambda_j(x_j) = \prod_{o\in\mathbb{S}_j} \mu_{o\to j}(x_j), \tag{5.3}$$

where $\mathbb{S}_j$ is the set of all function nodes, which have node j as input.

Finding the Input Distributions

In our algorithm, we use the log-likelihood ratio (LLR) to represent the probability distribution of binary variables [HOP96], defined as:

$$L_X = \ln \frac{p(x = +1)}{p(x = -1)}. \tag{5.4}$$

An overview of the algorithm is given in Algorithm 3.

The probability distribution of each node $j \in [n]$ at iteration t is given as $L_j^{(t)}$ and are initialized with $L_j^{(0)} = 0$, which is equivalent to the uniform distribution. Then we set the LLRs for the out-nodes to either $-\infty$ or $+\infty$, depending on the desired output $\mathbf{y}$ of the BN. At each iteration the algorithm can be split into two steps. The first step iterates over all function nodes $j \in [n] \setminus \mathbb{I}$ and all input variables $i \in \tilde{n}(f_j)$, calculating the LLR $L_{j\to i}^{(t)}$ using Eq. (5.2) and Eq. (5.4).

In the second step we update all variable nodes, where the LLRs L_j represents the distributions λ_j and, hence, the product of Eq. (5.3) becomes a summation. Please note that the LLR of the previous iteration is also added to the sum, in order to prevent rapid changes of the distributions.

After performing a certain number of iterations t_{max}, the desired marginal distributions of the input variables are found.

5.2.3.3. Sampling

The sampling part of this approach is straightforward. Using the marginal distributions $L_j^{(t_{max})}, j \in \mathbb{I}$ we randomly draw vectors $\mathbf{x}$ and check, if they satisfy $\mathbf{y} = f(\mathbf{x})$. If so, they are added to the set $\widetilde{\Psi}_{\mathbf{y}}$. This procedure is repeated for a certain number of samples.

Algorithm 3

Initialize $L_j^{(0)} = 0$ for all nodes.
Set the desired LLRs of the out-nodes, i.e., $L_j^{(0)}$ is either $-\infty$ or $+\infty$, for all out-nodes $j \in \mathbb{O}$.
t=0
repeat
  t=t+1
  for each non-in-node $j \in [n] \setminus \mathbb{I}$ **do**
    for each input variable $i \in \tilde{n}(f_j)$ **do**
      calculate $L_{j\to i}$ using Eq. (5.2) and Eq. (5.4)
    end for
  end for
  for each non-out-node v **do**
    $L_j^{(t)} = L_j^{(t-1)} + \sum_{o\in\mathbb{S}_j} L_{o\to j}^{(t)}$
  end for
until maximum number of iterations reached

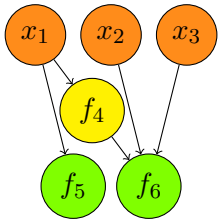


Figure 5.5.: Small Examplary Boolean Network

5.2.3.4. Small Example

Let us consider the exemplary network as shown in Figure 5.5. If we extract the functions into separate nodes and reorder them, we get the factor graph depicted in Figure 5.6 with variable nodes $x_1, x_2, x_3, x_4, x_5, x_6$ and function nodes f_4, f_5, f_6 In this picture we also included the initialized LLRs of the variable nodes. As x_5 and x_6 are the output nodes of the network, they are fixed. Here we choose $x_5 = -1$ and $x_6 = +1$, and consequently, $L_5 = -\infty$ and $L_6 = +\infty$. The other variable nodes are undetermined, hence $L_i = 0$ for $i \in \{1, 2, 3, 4\}$.

Next, let's discuss the update rules. The update of the variable nodes is performed by summing up the LLR values of all incoming messages, except the one to which the update is to be sent back (see last *for*-loop in Algorithm 3). The calculation of the messages from the function nodes to the variable nodes are more complicated. We will exemplary calculate the update messages $L_{f_6\to x_6}^{(t)}$ and $L_{f_6\to x_3}^{(t)}$, i.e., the outgoing messages of function node f_6. Let us assume that the truthtable of f_6 is given as follows:

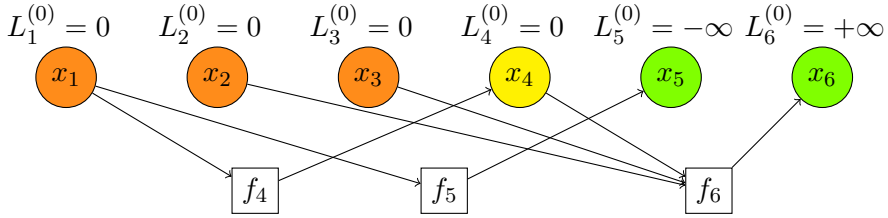


Figure 5.6.: Factor Graph of the BN from Figure 5.5

x_1	x_2	x_3	f_6
$+1$	$+1$	$+1$	$+1$
$+1$	$+1$	-1	-1
$+1$	-1	$+1$	-1
$+1$	-1	-1	-1
-1	$+1$	$+1$	-1
-1	$+1$	-1	-1
-1	-1	$+1$	-1
-1	-1	-1	-1

Let us recall that the message $\mu_{f_i \to x_j}(+1)$ gives the probability that variable node x_j is $+1$. Hence,

$$L^{(t)}_{f_i \to x_j} = \ln \frac{\mu_{f_i \to x_j}(+1)}{1 - \mu_{f_i \to x_j}(+1)},$$

since, obviously, $\mu_{f_i \to x_j}(-1) = 1 - \mu_{f_i \to x_j}(+1)$ and likewise:

$$L^{(t)}_{x_j \to f_i} = \ln \frac{\lambda_{x_j \to f_i}(+1)}{1 - \lambda_{x_j \to f_i}(+1)}.$$

As f_6 is only $+1$ if all three inputs are $x_1 = x_2 = x_3 = +1$, the update rule $\mu_{f_6 \to x_6}(+1)$ calculates as:

$$\mu_{f_6 \to x_6}(+1) = \lambda_{x_1 \to f_6}(+1) \cdot \lambda_{x_2 \to f_6}(+1) \cdot \lambda_{x_3 \to f_6}(+1).$$

Now let's examine the message sent from node f_6 to x_3, which is the probability distribution of x_3 given the distributions of x_1, x_2, x_6 and the BF f_6. If $x_1 = x_2 = +1$, which is the case with probability $\lambda_{x_1 \to f_6}(+1) \cdot \lambda_{x_2 \to f_6}(+1)$, the value of x_3 depends on the function's output, i.e., on its probability distribution given by $\lambda_{x_6 \to f_6}(\cdot)$. For other choices of x_1 and x_2 the function's output is always -1, i.e., x_3 has no influence. Thus, we assume that x_3 can be either $+1$ or -1, each case with probability $\frac{1}{2}$. Hence, the update message can be calculated as:

$$\begin{aligned}
\mu_{4 \to 3}(0) = {} & \lambda_{x_1 \to f_6}(+1) \cdot \lambda_{x_2 \to f_6}(+1) \cdot \lambda_{x_6 \to f_6}(+1) \\
& + \lambda_{x_1 \to f_6}(+1) \cdot \lambda_{x_2 \to f_6}(-1) \cdot \frac{1}{2} \\
& + \lambda_{x_1 \to f_6}(-1) \cdot \lambda_{x_2 \to f_6}(+1) \cdot \frac{1}{2} \\
& + \lambda_{x_1 \to f_6}(-1) \cdot \lambda_{x_2 \to f_6}(-1) \cdot \frac{1}{2}.
\end{aligned}$$

The remaining messages can be obtained in the same fashion.

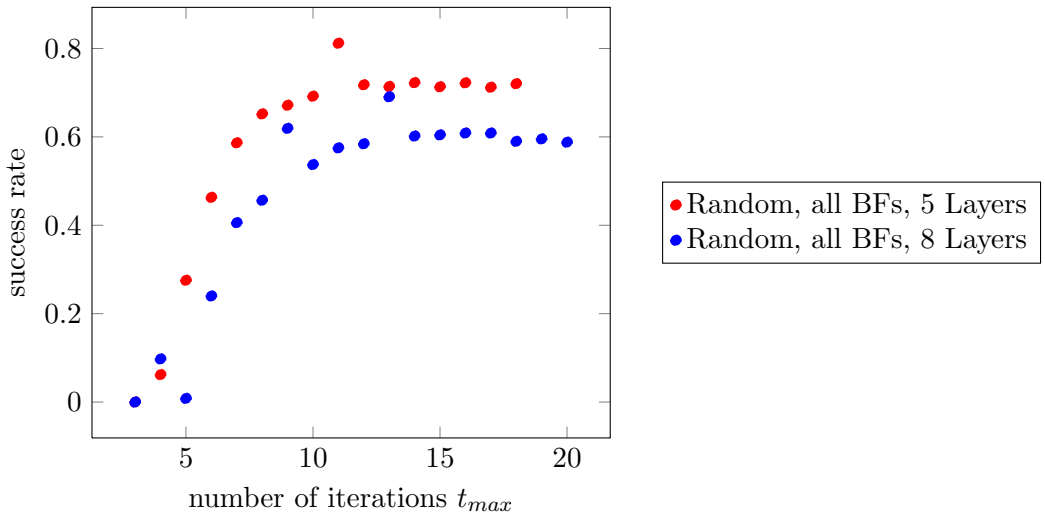


Figure 5.7.: Success rate of Algorithm 3 for Random BNs with 5 and 8 Layers plotted against the maximum number of iterations t_{max}.

5.2.3.5. Finding the Optimal Number of Iterations

Choosing the maximum number of iterations t_{max} is obviously a trade-off between computational complexity and the quality of the algorithm. In Figure 5.7 we plotted the success rate against the number of iterations for two randomly generated networks, with $K = 5$ and $K = 8$, respectively. We define the success rate as the fraction of networks and outputs, for which the algorithm found at least one valid preimage. One can see that the success rate converges at around 8 and 14 iterations, respectively. This corresponds to $2 \cdot (K - 1)$, which is the maximum number of hops in the network needed, to travel from the bottom layer to the top layer and back. As we are only considering networks up to 8 layers, we will choose $t_{max} = 14$ for the remainder of this chapter.

5.2.4. Investigated Networks

In this subsection we will briefly describe the Boolean networks, which we will investigate in this chapter. They basically consist of two variations of the regulatory network of *Escherichia coli* as presented by Covert et al. [CKR+04] and randomly generated networks with a similar topology.

5.2.4.1. The Regulatory Network of Escherichia coli

The regulatory network of *E. coli*, as presented in [CKR+04], provides Boolean formulas leading to an BN that describes how environmental conditions act on gene expression via a transcriptional regulatory network. We slightly modified this network as follows. Some nodes with in-degree zero, i.e., potential input nodes to the network, have a constant function (their outputs are either one or zero) attached. Therefore, we did not treat these nodes as input nodes, but fed their value to their successive nodes. Hence, we could restrict the functions of these successive nodes and reduce their in-degree. After that, the original nodes with constant functions became obsolete and were removed.

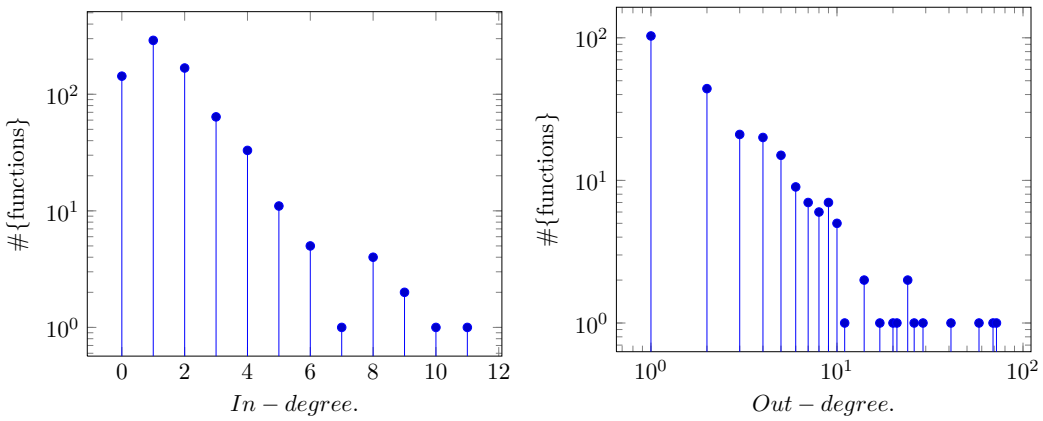


Figure 5.8.: In-degree and out-degree distribution of the regulatory network of *E. coli* (EcoliA).

Original Regulatory Network of Escherichia Coli (EcoliA) The resulting network has a total of 723 nodes on $\tilde{l} = 7$ layers. The first layer (input-layer) represents the environmental conditions. It consists of 143 nodes, describing the presence or absence of certain chemicals or metabolites (e.g. oxygen or glucose) and general states the cell may be in (e.g. heat-shock). The output-layer (472 nodes) contains mainly genes coding for proteins. The remaining nodes are regulatory genes. In Figure 5.8 we plotted the in- and out-degree distribution of this network. The in-degree distribution (without the nodes with zero in-degree) follows an exponential distribution. The average in-degree is 1.91897. Further, the out-degree distribution shows a typical long tail behavior [Ald03]. Please note that all BFs in this network are unate.

Extended Regulatory Network of Escherichia coli (EcoliB) The second network we consider, is an extension of the EcoliA-network introduced above. Feist et al. [FHR+07] published a mapping of the genes coding for proteins to the fluxes in the flux-balance model. Basically, this is a model, in which each flux represents a chemical reaction that can take place in the cell. This means, if certain proteins are present, we consider a flux as *on*, i.e., the reaction may take place. This relations are again described as BFs. We combined these two networks. If a gene occurs only in the latter network, i.e., it is not regulated by the Covert-network, we set it permanently to *on* and reduce the network and the successive nodes accordingly.

Thus, we obtain a network consisting of $\tilde{l} = 8$ layers and 1253 nodes of which 143 are input-nodes and 594 are output-nodes. The in- and out-degree distribution (see Figure 5.9) show the same properties as before. Further, all BFs are again unate.

5.2.4.2. Random Generated Networks

To show that our proposed algorithms work also for more general networks, we will test them also for four different random generated networks. Their basic statistics can be found in the following table:

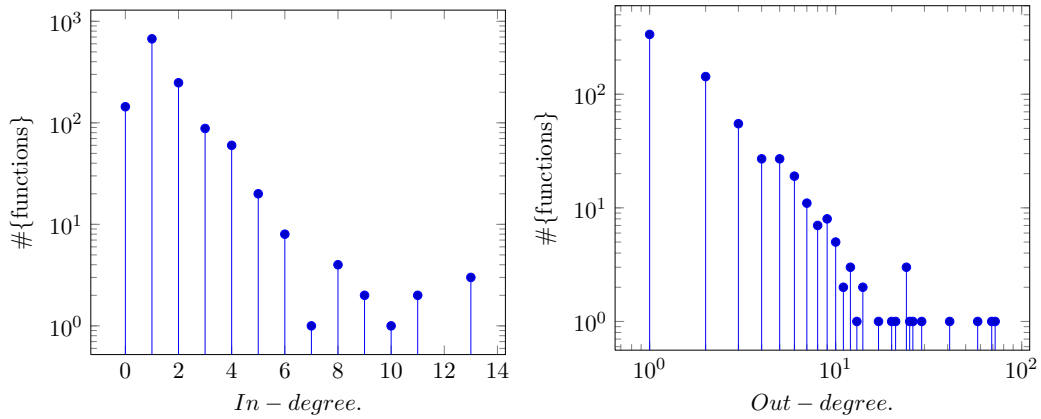


Figure 5.9.: In-degree and out-degree distribution of the extended regulatory network of *E. coli* (EcoliB).

name	#{nodes}	#{input-nodes}	#{output-nodes}	#{layers}	BFs class
RandomA	1550	150	600	8	unate
RandomB	1550	150	600	8	all
RandomC	2600	200	1200	8	unate
RandomD	2600	200	1200	8	all

Further, the in-degree of each function is drawn randomly from an exponential distribution with parameter $\lambda = 0.5$. Algorithm 4 in Appendix B shows the exact generation process. Further, as to our knowledge it is not possible to generate unate functions directly, one has to draw BFs randomly and check, if they are unate. However, the probability of drawing a unate function with $k > 5$ relevant variables is very small. In order to generate unate functions with a higher in-degree, we combine several unate functions, as a composition of unate functions is again unate.

5.2.5. Evaluation of the Proposed Algorithms

For each network type we evaluated 1000 networks and inputs, respectively. The results are listed in the following table. We first state the success rate of Algorithm 2 and the average size of set of preimages $\Psi_{\mathbf{y}}$. Then we give the success rate of Algorithm 3, followed by the average number of preimages found by Algorithm 3 after 1000 samples.

network	success rate Alg. 2	avg. size of $\Psi_{\mathbf{y}}$	success rate Alg. 3	avg. size of $\widetilde{\Psi}_{\mathbf{y}}$
EcoliA	100%	$5.38 \cdot 10^{16}$	99.1%	216.687
EcoliB	100%	$3.71 \cdot 10^{17}$	96.8%	122.27
RandomA	99.2%	5248	61.7%	24.51
RandomB	100%	10.75	52.6%	3.99
RandomC	100%	1189	91.07%	52.96
RandomD	100%	4.92	81.0%	3.62

The results show that Algorithm 2 is almost always successful, hence the graph can be split in feasible subsets. Further one can see that the average size of the set of preimages of the *E. coli* networks is rather high. One possible explanation for this effect may be that a large

number of input-nodes have only small influence on the output (see [Hec10]). However, these numbers are only a small fraction of the total number of possible input vectors ($\approx 2^{140}$).

Algorithm 3 also performs quite well. One can see from the results that in general for most networks, except the smaller random networks, at least one preimage can be found. If we increased the number of samples, we would get a larger $\widetilde{\Psi}_{\mathbf{y}}$ for the *E. coli* networks. This is also due to the fact that there exist a few inputs, which completely determine the output [Hec10]. The other input variables then have no influence and, hence, a marginal distribution of 0.5. Further, the results for the unate networks are much better than for non-unate ones. It seems that the marginal distributions for unate functions produce a better estimation of the actual distribution than the marginal distributions for non-unate functions.

As already mentioned, the fraction of preimages found for the *E. coli* networks is rather small. However, for the random networks we find large parts of $\Psi_{\mathbf{y}}$. This is particularly of interest, since Algorithm 3 performs much faster.

By replacing the fixed output values of the network with probabilities, one can simply apply Algorithm 3 to networks, whose designated output is described by probability distributions. Further, the algorithm may be easily adjusted to work on stochastic, e.g., Bayesian, networks, where the nodes only contain transition probabilities instead of Boolean functions. Therefore, the update rules have to be adapted accordingly.

It remains an open question, which influence topographical properties such as the number of layers and the number of nodes in these layers, have on the performance of the proposed algorithms, since we only investigated networks which are similar to the regulatory network of *E-coli*.

Chapter 6

Concluding Remarks

Within this thesis, we discussed important properties of canalizing and nested canalizing Boolean functions as well as the preimage problem of Boolean networks. First, we proved important spectral properties of those classes of Boolean functions, for which we applied Fourier analysis of Boolean functions, where functions are represented as multivariate, multilinear real polynomials. To this end, we stated spectral relationships between so-called restricted Boolean functions and their unrestricted counter part and gave a recursive expression for the zero coefficient of nested canalizing functions.

Next, we investigated the noise sensitivity, a measure for the error tolerance of Boolean functions. We gave an expression for this entity for the case of restricted Boolean functions and for the case of nested canalizing functions, we found a recursive description. Based on these findings, we derived several upper bounds for canalizing and nested canalizing functions, which showed that these functions have in general a very low noise sensitivity in comparison to other Boolean functions.

The derivative of the noise sensitivity yields into the so-called average sensitivity, which is also a measure for the error tolerance with a different channel model. Using the recursive expression, we provided a tight upper and lower bound on the average sensitivity for nested canalizing functions. We showed that the lower bound is achieved by functions, whose input variables are all most dominant and which maximize the absolute zero coefficient. The upper bound is reached by functions, whose canalized values are alternating. Using this bound, we proved an upper bound conjectured in literature on the average sensitivity of nested canalizing functions, namely $\mathtt{as}(f) < \frac{4}{3}$.

We also investigated the mutual information of Boolean functions in two scenarios. In the first scenario we were interested in the mutual information between one input variable and the function's output. We proved that canalizing functions maximize this mutual information, if the bias of the function is fixed. We also extended this result to the mutual information between a set of input variables and the output, which is maximized by *joint canalizing* functions. The second scenario covers, like the noise sensitivity, the mutual information between all inputs, each of which is disturbed by a binary symmetric channel, and the output. We stated two relations between the noise sensitivity and this mutual information. Based on this and our findings for the noise sensitivity, we established upper and lower bounds on the mutual information. These bounds indicate a high information processing capability of canalizing and nested canalizing functions.

Finally, we addressed the preimage problem of Boolean networks. This is of interest when designing experiments, in which certain regulators are supposed to be in a specific state.

It has been shown in literature that this problem is NP-hard. Hence, it is infeasible for large Boolean networks, like the regulatory network of *Escherichia coli*. We proposed two algorithms to solve this problem. The first algorithm reduces the size of the network and splits the network into several independent subgraphs, using the property that canalizing functions can be inverted under certain conditions. The known Wuensche algorithm for finding the predecessors can now be applied to these reduced subgraphs, since the problem has now become feasible. Secondly, we introduced a probabilistic method based on the well-known sum-product algorithm, which does not find the whole set of predecessors, but at least parts of it. We evaluated our algorithms by applying them to random Boolean networks and to the regulatory network of *E. coli*.

Appendix A

Proof of Lemmas from Chapter 4

A.1. Lemma 1

Lemma 1

If $-1 \leq \hat{f}(\emptyset) \leq -|\mu_i|$, *then*

$$q(\hat{f}(\emptyset)) < r(\hat{f}(\emptyset)) \text{ if } \mu_i > 0$$
$$q(\hat{f}(\emptyset)) > r(\hat{f}(\emptyset)) \text{ if } \mu_i < 0.$$

Proof: First, let's recall that

$$q(\hat{f}(\emptyset)) = \frac{1-\mu_i}{2} h\left(\frac{\hat{f}(\emptyset)+1}{1-\mu_i}\right)$$

and

$$r(\hat{f}(\emptyset)) = \frac{1+\mu_i}{2} h\left(\frac{\hat{f}(\emptyset)+1}{1+\mu_i}\right).$$

Let's assume that $\mu_i > 0$. Due to the concavity and the parabolic form of $q(\hat{f}(\emptyset))$ and $r(\hat{f}(\emptyset))$, they can intersect at most two times. Obviously, $q(-1) = r(-1) = 0$ and $q(-\mu_i) = 0 < r(-\mu_i)$. Hence, if the slope of r at $\hat{f}(\emptyset) = -1$ is larger than the slope of s, then $q(\hat{f}(\emptyset)) < r(\hat{f}(\emptyset))$ in the interval $-1 \leq \hat{f}(\emptyset) \leq -\mu_i$.

Building the derivative of $q(\hat{f}(\emptyset))$ leads us to

$$\begin{aligned} q'(\hat{f}(\emptyset)) &= \mathrm{P}_{X_i}(-1) \log\left(\frac{1-\frac{\hat{f}(\emptyset)+1}{1-\mu_i}}{\frac{\hat{f}(\emptyset)+1}{1-\mu_i}}\right) \\ &= \mathrm{P}_{X_i}(-1) \log_2\left(\frac{-\mu_i - \hat{f}(\emptyset)}{\hat{f}(\emptyset)+1}\right), \end{aligned}$$

and similarly to

$$r'(\hat{f}(\emptyset)) = \mathrm{P}_{X_i}(+1) \log_2\left(\frac{+\mu_i - \hat{f}(\emptyset)}{\hat{f}(\emptyset)+1}\right).$$

One can see that

$$\lim_{\hat{f}(\emptyset)\to -1} \Big(r'(\hat{f}(\emptyset)) - q'(\hat{f}(\emptyset)) \Big) = +\infty,$$

which concludes the proof for $\mu_i > 0$. The proof for $\mu_i < 0$ goes along the lines as for $\mu_i > 0$. ■

A.2. Lemma 2

Lemma 2

If $|\mu_i| \leq \hat{f}(\emptyset) \leq 1$, *then*

$$s(\hat{f}(\emptyset)) < t(\hat{f}(\emptyset)) \text{ if } \mu_i > 0$$
$$s(\hat{f}(\emptyset)) > t(\hat{f}(\emptyset)) \text{ if } \mu_i < 0.$$

Proof: First, let's recall that

$$s(\hat{f}(\emptyset)) = \frac{1-\mu_i}{2} h\left(\frac{\hat{f}(\emptyset) - \mu_i}{1-\mu_i} \right)$$

and

$$t(\hat{f}(\emptyset)) = \frac{1+\mu_i}{2} h\left(\frac{\hat{f}(\emptyset) + \mu_i}{1+\mu_i} \right).$$

Now we assume that $\mu_i > 0$. Due to the concavity and the parabolic form of $s(\hat{f}(\emptyset))$ and $t(\hat{f}(\emptyset))$, they can intersect at most two times. Obviously, $s(+1) = t(+1) = 0$ and $s(\mu_i) = 0 < t(\mu_i)$. Hence, if the slope of t at $\hat{f}(\emptyset) = -1$ is larger than the slope of s, then $s(\hat{f}(\emptyset)) < t(\hat{f}(\emptyset))$ in the interval $\mu_i \leq \hat{f}(\emptyset) \leq 1$.

Building the derivative of $s(\hat{f}(\emptyset))$ leads us to

$$\begin{aligned} s'(\hat{f}(\emptyset)) &= \mathrm{P}_{X_i}(-1) \log_2 \left(\frac{1 - \frac{\hat{f}(\emptyset)-\mu_i}{1-\mu_i}}{\frac{\hat{f}(\emptyset)-\mu_i}{1-\mu_i}} \right) \\ &= \mathrm{P}_{X_i}(-1) \log_2 \left(\frac{1-\hat{f}(\emptyset)}{\hat{f}(\emptyset)-\mu_i} \right) \end{aligned}$$

and similarly to

$$t'(\hat{f}(\emptyset)) = \mathrm{P}_{X_i}(+1) \log_2 \left(\frac{1-\hat{f}(\emptyset)}{\hat{f}(\emptyset)+\mu_i} \right).$$

One can see that

$$\lim_{\hat{f}(\emptyset)\to 1} \Big(t'(\hat{f}(\emptyset)) - s'(\hat{f}(\emptyset)) \Big) = -\infty,$$

which concludes the proof for $\mu_i > 0$. The proof for $\mu_i < 0$ goes along the lines as for $\mu_i > 0$. ■

A.3. Lemma 3

Lemma 3

If $\mu_i > 0$ and $|\hat{f}(\emptyset)| \leq |\mu_i|$, then

$$t(\hat{f}(\emptyset)) < r(\hat{f}(\emptyset)) \text{ if } \hat{f}(\emptyset) < 0$$
$$t(\hat{f}(\emptyset)) > r(\hat{f}(\emptyset)) \text{ if } \hat{f}(\emptyset) > 0.$$

If $\mu_i < 0$ and $|\hat{f}(\emptyset)| \leq |\mu_i|$, then

$$s(\hat{f}(\emptyset)) < q(\hat{f}(\emptyset)) \text{ if } \hat{f}(\emptyset) < 0$$
$$s(\hat{f}(\emptyset)) > q(\hat{f}(\emptyset)) \text{ if } \hat{f}(\emptyset) > 0.$$

Proof: Let's first assume $\mu_i > 0$. One can easily see that

$$\begin{aligned} t(-\mu_i) &= \mathrm{P}_{X_i}(+1)\, h\left(\frac{-\mu_i + \mu_i}{1+\mu_i}\right) = 0 \\ &< r(-\mu_i) = \mathrm{P}_{X_i}(+1)\, h\left(\frac{-\mu_i + 1}{1+\mu_i}\right), \end{aligned}$$

and

$$\begin{aligned} t(\mu_i) &= \mathrm{P}_{X_i}(+1)\, h\left(\frac{\mu_i + \mu_i}{1+\mu_i}\right) \\ &> r(\mu_i) = \mathrm{P}_{X_i}(+1)\, h\left(\frac{\mu_i + 1}{1+\mu_i}\right) = 0. \end{aligned}$$

Further,

$$\begin{aligned} t(0) &= \mathrm{P}_{X_i}(+1)\, h\left(\frac{\mu_i}{1+\mu_i}\right) \\ &= \mathrm{P}_{X_i}(+1)\, h\left(1 - \frac{\mu_i}{1+\mu_i}\right) \\ &= \mathrm{P}_{X_i}(+1)\, h\left(\frac{1}{1+\mu_i}\right) = r(0), \end{aligned}$$

which due to concavity of t and r proves the first part of the Lemma. The proof of the second part goes along the lines. ■

Appendix B

Algorithm to randomly generate Boolean Networks

Algorithm 4

Given: number of layers l.
Given: total number of nodes K.
Given: number of nodes on the first layer K_1.
Given: number of nodes on the last layer K_l.
Draw number of nodes on layers $2, \ldots, l-1$ uniformly, so that $\sum_{i=2}^{l-1} K_i = K - K_1 - K_l$.
Add nodes 1 to K_1 to the network.
for each layer i,$1 < i \leq l$ **do**
  for $0 < j < K_i$ **do**
    draw k from exponential distribution with $\lambda = 0.5$;
    randomly draw function with k relevant variables;
    add new node to network;
    draw k input nodes from all nodes with layer $< i$;
    add corresponding edges;
  end for
end for

Bibliography

References

[ACK03] Maximino Aldana, Susan Coppersmith, and Leo P. Kadanoff. Boolean dynamics with random couplings. In Ehud Kaplan, Jerrold E. Marsden, and Katepalli R. Sreenivasan, editors, *Perspectives and Problems in Nolinear Science*, pages 23–89. Springer New York, 2003.

[AHZ+09] Tatsuya Akutsu, Morihiro Hayashida, Shu-Qin Zhang, Wai-Ki Ching, and Michael K. Ng. Analyses and algorithms for predecessor and control problems for Boolean networks of bounded indegree. *Information and Media Technologies*, 4(2):338–349, 2009.

[AJL+04] Bruce Alberts, Alexander Johnson, Julian Lewis, Martin Raff, Keith Roberts, and James D. Watson. *Molekularbiologie der Zelle*. Wiley-VCH, 4th edition, 2004.

[Ald03] Maximino Aldana. Boolean dynamics of networks with scale-free topology. *Physica D: Nonlinear Phenomena*, 185(1):45–66, October 2003.

[Bah61] Raghu R. Bahadur. A representation of the joint distribution of responses to n dichotomous items. In Herbert Solomon, editor, *Studies on Item Analysis and Prediction*, number 6 in Stanford mathematical studies in the social sciences, pages 158–176. Stanford University Press, Stanford CA, 1961.

[Ber98] Anna Bernasconi. *Mathematical Techniques for the Analysis of Boolean Functions*. PhD thesis, University of Pisa, Italy, 1998.

[BKS99] Itai Benjamini, Gil Kalai, and Oded Schramm. Noise sensitivity of Boolean functions and applications to percolation. *Publications mathematiques de l'IHES*, 90:5–43, 1999.

[BOL85] Michael Ben-Or and Nathan Linial. Collective coin flipping, robust voting schemes and minima of Banzhaf values. In *FOCS*, pages 408–416. IEEE Computer Society, 1985.

[Boo47] George Boole. *The mathematical analysis of logic*. Philosophical Library, 1847.

[Bos13] Martin Bossert. *Kanalcodierung*. Oldenbourg Verlag, München, third edition, 2013.

[BSM05] Jon T Butler, Tsutomu Sasao, and Munehiro Matsuura. Average path length of binary decision diagrams. *IEEE Transactions on Computers*, 54(9):1041–1053, 2005.

[BT96] Nader H. Bshouty and Christino Tamon. On the Fourier spectrum of monotone functions. *Journal of the ACM*, 43(4):747–770, 1996.

[BV04] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, March 2004.

[CDR86] Stephen Cook, Cynthia Dwork, and Rüdiger Reischuk. Upper and lower time bounds for parallel random access machines without simultaneous writes. *SIAM J. Comput.*, 15(1):87–97, February 1986.

[CKR+04] Markus W. Covert, Eric M. Knight, Jennifer L. Reed, Markus J. Herrgard, and Bernhard O. Palsson. Integrating high-throughput and computational data elucidates bacterial networks. *Nature*, 429(6987):92–96, May 2004.

[CY90] James P. Crutchfield and Karl Young. Computation at the onset of chaos. *Complexity, Entropy, and the Physics of Information*, 1990.

[DB08] Maria I. Davidich and Stefan Bornholdt. Boolean network model predicts cell cycle sequence of fission yeast. *PLoS ONE*, 3(2), 02 2008.

[FET+07] Ronny Feuer, Michael Ederer, Natalie Trachtmann, Thomas Sauter, Ernst Dieter Gilles, Georg Sprenger, and Oliver Sawodny. Methods for analysis of evolutive adaptation of E. coli. In *SICE, 2007 Annual Conference*, pages 1359–1365, Sept. 2007.

[FHR+07] Adam M. Feist, Christopher S. Henry, Jennifer L. Reed, Markus Krummenacker, Andrew R. Joyce, Peter D. Karp, Linda J. Broadbelt, Vassily Hatzimanikatis, and Bernhard O. Palsson. A genome-scale metabolic reconstruction for Escherichia coli K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information. *Molecular Systems Biology*, 3:121, Jun 2007.

[FJS91] Merrick L. Furst, Jeffrey C. Jackson, and Sean W. Smith. Improved learning of AC0 functions. In *Proceedings of the Fourth Annual Workshop on Computational Learning Theory*, pages 317–325, Santa Cruz, California, United States, 1991. Morgan Kaufmann Publishers Inc.

[Fri98] Ehud Friedgut. Boolean functions with low average sensitivity depend on few coordinates. *Combinatorica*, 18(1):27–35, January 1998.

[Gal63] Robert G. Gallager. *Low-Density Parity-Check Codes*. M.I.T. Press, Cambridge, 1963.

[GKK06] John Grefenstette, Sohyoung Kim, and Stuart Kauffman. An analysis of the class of gene regulatory functions implied by a biochemical model. *Biosystems*, 84(2):81–90, May 2006.

[Gol59] Solomon W. Golomb. On the classification of boolean functions. *IRE Transactions on Circuit Theory*, 6(5):176 – 186, may 1959.

[GYC92] Moncef Gabbouj, Pao-Ta Yu, and Edward J. Coyle. Convergence behavior and root signal sets of stack filters. *Circuits, Systems, and Signal Processing*, 11(1):171–193, March 1992.

[Hec10] Reinhard Heckel. Analysis of binary polynomial systems using the Foruier transform. Diploma Thesis, Ulm University, 2010.

[HM91] Thomas Hancock and Yishay Mansour. Learning monotone $k\mu$ DNF formulas on product distributions. In *Proceedings of the fourth annual workshop on Computational learning theory*, COLT '91, pages 179–183, San Francisco, CA, USA, 1991. Morgan Kaufmann Publishers Inc.

[HOP96] J. Hagenauer, E. Offer, and L. Papke. Iterative decoding of binary block and convolutional codes. *IEEE Transactions on Information Theory*, 42(2):429–445, 1996.

[HSB13] Reinhard Heckel, Steffen Schober, and Martin Bossert. Harmonic analysis of Boolean networks: Determinative power and perturbations. *EURASIP Journal on Bioinformatics and Systems Biology*, 2013(6), 2013.

[HSWK02] Stephen E. Harris, Bruce K. Sawhill, Andrew Wuensche, and Stuart A. Kauffman. A model of transcriptional regulatory networks based on biases in the observed regulation rules. *Complexity*, 7(4):23–40, 2002.

[HV86] Hyman Hartman and Grard Y. Vichniac. Inhomogeneous cellular automata (inca). In E. Bienenstock, F.Fogelman Soulié, and G. Weisbuch, editors, *Disordered Systems and Biological Organization*, volume 20 of *NATO ASI Series*, pages 53–57. Springer Berlin Heidelberg, 1986.

[JK08] Wilfried Janning and Elisabeth Knust. *Genetik: allgemeine Genetik, molekulare Genetik, Entwicklungsgenetik*. Georg Thieme Verlag, 2nd edition, 2008.

[JM62] Francois Jacob and Jacques Monod. Genetic repression, allosteric inhibition and cellular differentiation. *Cytodifferentiation and Macromolecular Synthesis: 21st Symposium of the Society for the Study of Development and Growth*, 21:30–64, 1962.

[JRL07] Abdul Salam Jarrah, Blessilda Raposa, and Reinhard Laubenbacher. Nested canalyzing, unate cascade, and polynomial functions. *Physica D*, 233(2):167–174, 2007.

[JSK04] Winfried Just, Ilya Shmulevich, and John Konvalina. The number and probability of canalizing functions. *Physica D: Nonlinear Phenomena*, 197(3-4):211–221, October 2004.

[Kau67] Stuart A. Kauffman. Sequential DNA replication and the control of differences in gene activity between sister chromatids—a possible factor in cell differentiation. *Journal of Theoretical Biology*, 17(3):483 – 497, 1967.

[Kau69a] Stuart A. Kauffman. Homeostasis and Differentiation in Random Genetic Control Networks. *Nature*, 224(5215):177–178, October 1969.

[Kau69b] Stuart A. Kauffman. Metabolic stability and epigenesis in randomly constructed nets. *Journal of Theoretical Biology*, 22:437–467, 1969.

[Kau74] Stuart A. Kauffman. The large scale structure and dynamics of gene control circuits: An ensemble approach. *Journal of Theoretical Biology*, 44:167–190, 1974.

[Kau93] Stuart A. Kauffman. *The Origins of Order: Self Organization and Selection in Evolution*. Oxford University Press, 1993.

[KC13] Gowtham R. Kumar and Thomas A. Courtade. Which Boolean functions are most informative? In *Proceedings of the IEEE International Symposium on Information Theory*, pages 226–230, Istanbul, Turkey, July 2013.

[KFL01] Frank Kschischang, Brendan J. Frey, and Hans-Andrea Loeliger. Factor Graphs and the Sum-Product Algorithm. *IEEE Transactions on Information Theory*, 47:498–519, 2001.

[KKL88] Jeff Kahn, Gill Kalai, and Nathan Linial. The influence of variables on Boolean functions. In *Proceedings of the 29th Annual Symposium on Foundations of Computer Science*, pages 68–80, White Plains, New York, USA., October 1988.

[KLW+09] Edda Klipp, Wolfram Liebermeister, Christoph Wierling, Axel Kowald, Hans Lehrach, and Ralf Herwig. *Systems Biology*. Wiley, 2009.

[KPST03] Stuart A. Kauffman, Carsten Peterson, Bjorn Samuelsson, and Carl Troein. Random Boolean network models and the yeast transcriptional network. *Proceedings of the National Academy of Sciences of the United States of America*, 100(25):14796–14799, December 2003.

[KPST04] Stuart A. Kauffman, Carsten Peterson, Björn Samuelsson, and Carl Troein. Genetic networks with canalyzing Boolean rules are always stable. *Proceedings of the National Academy of Sciences of the United States of America*, 101(49):17102–17107, 2004.

[KRYH05] Juha Kesseli, Pauli Rämö, and Olli Yli-Harja. Analyzing dynamics of boolean networks with canalyzing functions using spectral methods. In *Proceedings of the 2005 International TICSP Workshop on Spectral Methods and Multirate Signal Processing (SMMSP 2005)*, pages 151–158, Riga, Latvia, June 2005.

[Lö08] Leopold Löwenheim. Über das Auflösungsproblem im logischen Klassenkalkül. *Sitzungsberichte der Berliner Mathematischen Gesellschaft*, 7:89–94, 1908.

[LAM+13] Yuan Li, John O. Adeyeye, David Murrugarra, Boris Aguilar, and Reinhard Laubenbacher. Boolean nested canalizing functions: A comprehensive analysis. *Theoretical Computer Science*, 481:24–36, April 2013.

[Lan90] Chris G. Langton. Computation at the edge of chaos: phase transitions and emergent computation. *Physica D*, 42(1-3):12–37, June 1990.

[LF00] Bartolo Luque and Antonio Ferrera. Measuring mutual information in random Boolean networks. *Complex systems*, 12, 2000.

[LLL+04] Fangting Li, Tao Long, Ying Lu, Qi Ouyang, and Chao Tang. The yeast cell-cycle network is robustly designed. *Proceedings of the National Academy of Sciences of the United States of America*, 101(14):4781–4786, April 2004. PMC387325.

[Lyn07] James F. Lynch. Dynamics of random Boolean networks. In R. Culshaw K. Mahdavi and J. Boucher, editors, *Current Developments in Mathematical Biology: Proceedings of the Conference on Mathematical Biology and Dynamical Systems*, pages 15 – 38. World Scientific Publishing Co., 2007.

[Mai62] K. K. Maitra. Cascaded switching networks of two-input flexible cells. *IRE Transactions on Electronic Computers*, (2):136–143, 1962.

[McN61] Robert McNaughton. Unate truth functions. *IRE Transactions on Electronic Computers*, EC-10(1):1–6, 1961.

[MO03] Elchanan Mossel and Ryan O'Donnell. On the noise sensitivity of monotone functions. *Random Structures & Algorithms*, 23(3):333–350, October 2003.

[Muk69] Asish Mukhopadhyay. Unate cellular logic. *IEEE Transactions on Computers*, 18(2):114–121, 1969.

[Mun00] Katharina Munk, editor. *Grundstudium Biologie: Biochemie, Zellbiologie, Ökologie, Evolution.* Grundstudium Biologie. Spektrum, Akad. Verlag, 2000.

[O'D03] Ryan O'Donnell. *Computational applications of noise sensitivity.* PhD thesis, Massachusetts Institute of Technology, 2003.

[O'D08] Ryan O'Donnell. Some topics in analysis of boolean functions. In *Proceedings of the 40th annual ACM symposium on Theory of computing*, pages 569–578, Victoria, British Columbia, Canada, 2008. ACM.

[Por84] Platon S. Poretski. On the method for solving logical equations and on the inverse method for mathematical logic (in russian). *Sobrante protokolov zasedanit fis. mat. Kasan*, 2:161–330, 1884.

[RKLP+08] Andre S. Ribeiro, Stuart A. Kauffman, Jason Lloyd-Price, Björn Samuelsson, and Joshua E. S. Socolar. Mutual information in random boolean models of regulatory networks. *Phys. Rev. E*, 77:011901, Jan 2008.

[Rud74] Sergiu Rudeanu. *Boolean functions and equations.* North-Holland, 1974.

[SB07] Steffen Schober and Martin Bossert. Analysis of random Boolean networks using the average sensitivity. 2007. `arXiv:0704.0197`.

[Sch] Steffen Schober. Personal communication. Ulm, January 2013.

[Sch90] Ernst Schröder. *Vorlesungen über die Algebra der Logik.* B.G. Teubner, 1890.

[Sch13] Steffen Schober. *Lecture notes on Information Theory and Biology.* Ulm University, 2013.

[Sha48] Claude E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 1948.

[Sha49] Claude E. Shannon. The synthesis of two terminal switching circuits. *Bell System Technical Journal*, 28:59–98, 1949.

[Sha69] John L. Shanks. Computation of the fast Walsh-Fourier transform. *IEEE Transactions on Computers*, C-18(5):457–459, 1969.

[Shm05] Ilya Shmulevich. Average sensitivity of typical monotone Boolean functionss, 2005. `arXiv:math/0507030`.

[Sim83] Hans-Ulrich Simon. A tight $\Omega(\log\log n)$-bound on the time for parallel RAM's to compute nondegenerated boolean functions. In Marek Karpinski, editor, *Foundations of Computation Theory*, volume 158 of *Lecture Notes in Computer Science*, pages 439–444. Springer Berlin Heidelberg, 1983.

[SK04] Ilya Shmulevich and Stuart A. Kauffman. Activities and sensitivities in Boolean network models. *Physical Review Letters*, 93(4):048701, July 2004.

[SLE04] Ilya Shmulevich, Harri Lähdesmäki, and Karen Egiazarian. Spectral methods for testing membership in certain post classes and the class of forcing functions. *Signal Processing Letters, IEEE*, 11(2):289 – 292, February 2004.

[SML+96] Ricard V. Solé, Susanna C. Manrubia, Bartolo Luque, Jordi Delgado, and Jordi Bascompte. Phase transitions and complex systems. *Complexity*, 1(4):13–26, 1996.

[Sta87] Dietrich Stauffer. On forcing functions in Kauffman's random Boolean networks. *Journal of Statistical Physics*, 46(3-4):789–794, 1987.

[Sto35] Marshall H Stone. Subsumption of the theory of Boolean algebras under the theory of rings. *Proceedings of the National Academy of Sciences of the United States of America*, 21(2):103, 1935.

[Sto36] Marshall H Stone. The theory of representation for Boolean algebras. *Transactions of the American Mathematical Society*, 40(1):37–111, 1936.

[Tit62] Robert .C. Titsworth. *Correlation Properties of Cyclic Sequences*. PhD thesis, California Institute of Technology, 1962.

[Top01] Flemming Topsøe. Bounds for entropy and divergence for distributions over a two-element set. *JIPAM. Journal of Inequalities in Pure & Applied Mathematics [electronic only]*, 2(2):null, 2001.

[Wad42] Conrad H. Waddington. Canalization of development and the inheritance of acquired characters. *Nature*, 150:563–565, November 1942.

[WCG86] Peter Wendt, Edward J. Coyle, and Neal C. Jr. Gallagher. Stack filters. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 34(4):898 – 911, aug 1986.

[Weg87] Inge Wegener. *The Complexity of Boolean Functions*. John Wiley & Sons, Inc., New York, NY, USA, 1987.

[Wue94] Andrew Wuensche. The ghost in the machine: Basins of attraction of random Boolean networks. In *Artificial Life III Proceedings, Santa Fe Institute Studies in the Sciences of Complexity*. Addison-Wesley, 1994.

[YC90] Pao-Ta Yu and Edward J. Coyle. Convergence behavior and n-roots of stack filters. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 38(9):1529 –1544, sep 1990.

Publications Containing Parts of this Thesis

[KBS13a] Johannes Georg Klotz, Martin Bossert, and Steffen Schober. Computing preimages of boolean networks. *BMC Bioinformatics*, 14(Suppl 10):S4, August 2013.

[KBS13b] Johannes Georg Klotz, Martin Bossert, and Steffen Schober. On the noise sensitivity and mutual information of (nested-) canalizing Boolean functions. In *IEEE Information Theory Workshop, ITW 2013*, 2013.

[KFG+11] Johannes Georg Klotz, Ronny Feuer, Katrin Gottlieb, Oliver Sawodny, Georg Sprenger, Martin Bossert, Michael Ederer, and Steffen Schober. Properties of a Boolean network model of *Escherichia coli*. In *Proc. of the 8th International Workshop on Computational Systems Biology (WCSB)*, pages 108–111, Zuerich, Switzerland, June 2011.

[KFS+13] Johannes Georg Klotz, Ronny Feuer, Oliver Sawodny, Martin Bossert, Michael Ederer, and Steffen Schober. Properties of Boolean networks and methods for their tests. *EURASIP Journal on Bioinformatics and Systems Biology*, 2013(1):1, 2013.

[KHS13] Johannes Georg Klotz, Reinhard Heckel, and Steffen Schober. Bounds on the average sensitivity of nested canalizing functions. *PLoS ONE*, 8 (5), 2013.

[KKBS13] Johannes Georg Klotz, David Kracht, Martin Bossert, and Steffen Schober. Canalizing Boolean functions maximize the mutual information. In *9th International ITG Conference on Systems, Communications and Coding (SCC)*, Munich, Germany,, January 2013.

[KKBS14] Johannes Georg Klotz, David Kracht, Martin Bossert, and Steffen. Schober. Canalizing Boolean functions maximize mutual information. *Information Theory, IEEE Transactions on*, 60(4):2139–2147, April 2014.

[KSB11] Johannes Georg Klotz, Steffen Schober, and Martin Bossert. On the predecessor problem in Boolean network models of regulatory networks. In *Proc. of the 3rd International Conference on Bioinformatics and Computational Biology (BICoB)*, pages 67–73, New Orleans, Louisiana, USA, March 2011.

[KSB12] Johannes Georg Klotz, Steffen Schober, and Martin Bossert. On the predecessor problem in Boolean network models of regulatory networks. *International Journal of Computers and Their Applications*, 19(2):93–100, June 2012.

List of Symbols and Acronyms

χ ..... Basis for Fourier transform in the uniform case

$\hat{f}$ ..... Fourier representation of BF f

ϕ ..... Basis for Fourier transform in the product distributed case

LLR ..... Log-Likelihood Ratio

`as` ..... Average Sensitivity

BF ..... Boolean function

BN ..... Boolean Network

BSC ..... Binary Symmetric Channel

CF ..... Canalizing Boolean Function

CNS ..... Conditional Noise Sensitivity

$\mathbf{E}[X]$ ..... Expectation value of random variable X

K ..... Number of nodes in a Boolean Network

k ..... Number of relevant variables of a BF

MI ..... Mutual Information

n ..... Number of input variables of a BF

NCF ..... Nested Canalizing Boolean Function

NCF-ACV ..... NCF with alternating canalized values

NCF-MDV ..... NCF with only most dominant variables

NCF-SMD ..... NCF with second layer most dominant variables

NS ..... Noise Sensitivity

SPA ..... Sum-Product Algorithm

UBF ..... Unate Boolean function

$\mathbf{Var}[X]$ ..... Variance of random variable X

Johannes Georg Klotz

Der Lebenslauf wurde aus Gründen des Datenschutzes entfernt.

✉ *Johannes.Klotz@uni-ulm.de*

List of Publications

Book Chapters

Martin Bossert, Carolin Huppert, and Johannes Georg Klotz. *OFDM Concepts for Future Communication Systems*, chapter Resource Allocation Using Broadcast Techniques, pages 122–127. Springer Verlag, 2011.

Scientific Journals

Carolin Huppert, Frederic Knabe, and Johannes Georg Klotz. User assignment for minimum rate requirements in OFDM-MIMO broadcast systems. *Electronics Letters*, 45(12):621–623, 2009.

Johannes Georg Klotz, Frederic Knabe, and Carolin Huppert. Resource allocation algorithms for minimum rates scheduling in MIMO-OFDM systems. *European Transactions on Telecommunications*, 21(5):449–457, August 2010.

Carolin Huppert and Johannes Georg Klotz. Required transmit power applying Tomlinson-Harashima-Precoding in scalar and MIMO broadcast systems. *IEEE Transactions on Communications*, 58(10):3011–3020, 2010.

Johannes Georg Klotz, Steffen Schober, and Martin Bossert. On the predecessor problem in Boolean network models of regulatory networks. *International Journal of Computers and Their Applications*, 19(2):93–100, June 2012.

Ronny Feuer, Katrin Gottlieb, Gero Viertel, Johannes Klotz, Steffen Schober, Martin Bossert, Oliver Sawodny, Georg Sprenger, and Michael Ederer. Model-based analysis of an adaptive evolution experiment with Escherichia coli in a pyruvate limited continuous culture with glycerol. *EURASIP Journal on Bioinformatics and Systems Biology*, 2012(1):14, October 2012.

Johannes Georg Klotz, Ronny Feuer, Oliver Sawodny, Martin Bossert, Michael Ederer, and Steffen Schober. Properties of Boolean networks and methods for their tests. *EURASIP Journal on Bioinformatics and Systems Biology*, 2013(1):1, 2013.

Johannes Georg Klotz, Reinhard Heckel, and Steffen Schober. Bounds on the average sensitivity of nested canalizing functions. *PLoS ONE*, 8 (5), 2013.

Johannes Georg Klotz, David Kracht, Martin Bossert, and Steffen. Schober. Canalizing Boolean functions maximize mutual information. *Information Theory, IEEE Transactions on*, 60(4):2139–2147, April 2014.

Conference Contributions

Stephan Stiglmayr, Johannes Georg Klotz, and Martin Bossert. Adaptive coding and modulation in MIMO OFDMA systems. In *Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on*, pages 1–5, 2008.

Stephan Stiglmayr, Johannes Georg Klotz, and Martin Bossert. Mutual information of V-BLAST transmission. In *Wireless Communication Systems. 2008. ISWCS '08. IEEE International Symposium on*, pages 468–472, Reykjavik, Iceland, October 2008.

✉ *Johannes.Klotz@uni-ulm.de*

Johannes Georg Klotz, Carolin Huppert, and Martin Bossert. Heuristic resource allocation for sum rate optimization in MIMO-OFDM systems using eigenvalue updates. In *Wireless Communication Systems. 2008. ISWCS '08. IEEE International Symposium on*, Reykjavik, Iceland, October 2008.

Johannes Georg Klotz, Frederic Knabe, and Carolin Huppert. Resource allocation algorithms for minimum rates scheduling in MIMO-OFDM systems. In Simon Plass, Armin Dammann, Stefan Kaiser, and Khaled Fazel, editors, *Multi-Carrier Systems & Solutions 2009*, volume 41 of *Lecture Notes in Electrical Engineering*, pages 79–88. Springer Netherlands, 2009.

Frederic Knabe, Moriz Wiese, Carolin Huppert, and Johannes Georg Klotz. The wideband slope region of BPSK and QPSK for broadcast channels in the low-power regime. In *Information Theory Workshop, 2009. ITW 2009. IEEE*, pages 440–444, Taormina, Sicily, Italy, October 2009.

Johannes Georg Klotz and Aydin Sezgin. Antenna selection criteria for interference alignment. In *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Istanbul, Turkey, September 2010.

Johannes Georg Klotz, Frederic Knabe, and Martin Bossert. Combined resource allocation and antenna selection for the MIMO-OFDM downlink. In *Proceedings of 15th International OFDM Workshop*, Hamburg, Germany, September 2010.

Moriz Wiese, Frederic Knabe, Johannes Georg Klotz, and Aydin Sezgin. The performance of QPSK in low-SNR interference channels. In *Information Theory and its Applications (ISITA), 2010 International Symposium on*, pages 708–713, Taichung, Taiwan, October 2010.

Johannes Georg Klotz, Steffen Schober, and Martin Bossert. On the predecessor problem in Boolean network models of regulatory networks. In *Proc. of the 3rd International Conference on Bioinformatics and Computational Biology (BICoB)*, pages 67–73, New Orleans, Louisiana, USA, March 2011.

Ronny Feuer, Katrin Gottlieb, Johannes Georg Klotz, Steffen Schober, Martin Bossert, Oliver Sawodny, Georg Sprenger, and Michael Ederer. Model-based analysis of adaptive evolution. In *Proceedings of the 8th International Workshop on Computational Systems Biology (WCSB)*, Zuerich, Switzerland, June 2011.

Johannes Georg Klotz, Ronny Feuer, Katrin Gottlieb, Oliver Sawodny, Georg Sprenger, Martin Bossert, Michael Ederer, and Steffen Schober. Properties of a Boolean network model of *Escherichia coli*. In *Proc. of the 8th International Workshop on Computational Systems Biology (WCSB)*, pages 108–111, Zuerich, Switzerland, June 2011. **best paper award**.

Johannes Georg Klotz, Gero Viertel, Ronny Feuer, Katrin Gottlieb, Oliver Sawodny, Georg Sprenger, Martin Bossert, Michael Ederer, and Steffen Schober. Prediction of regulatory impact under defined environmental conditions based on a combination of metabolic and transcriptional networks. In *Ninth International Workshop on Computational Systems Biology, WCSB 2012, June 4-6, Ulm, Germany*, page 100.

✉ *Johannes.Klotz@uni-ulm.de*

Carolin Huppert and Johannes Georg Klotz. On the performance of Tomlinson-Harashima-Precoding with tilted constellation in two user Gaussian broadcast channels. In *9th International ITG Conference on Systems, Communications and Coding (SCC)*, Munich, Germany,, January 2013.

Johannes Georg Klotz, David Kracht, Martin Bossert, and Steffen Schober. Canalizing Boolean functions maximize the mutual information. In *9th International ITG Conference on Systems, Communications and Coding (SCC)*, Munich, Germany, January 2013.

Johannes Georg Klotz, Martin Bossert, and Steffen Schober. Computing preimages of Boolean networks. In *Proc. of the 10th International Workshop on Computational Systems Biology (WCSB)*, Tampere, Finland, June 2013.

Johannes Georg Klotz, Martin Bossert, and Steffen Schober. On the noise sensitivity and mutual information of (nested-) canalizing Boolean functions. In *IEEE Information Theory Workshop, ITW 2013*, 2013.

✉ *Johannes.Klotz@uni-ulm.de*