Theoretical Computer Science 720 (2018) 36–46

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Multiplicative complexity of vector valued Boolean functions

Joan Boyar *, Magnus Gausdal Find

Department of Mathematics and Computer Science, University of Southern Denmark, Campusvej 55, DK-5230 Odense M, Denmark

ARTICLE INFO

Article history:
Received 16 February 2017
Received in revised form 15 February 2018
Accepted 21 February 2018
Available online 1 March 2018
Communicated by M.J. Golin

Keywords:
Multiplicative complexity
Nonlinearity
Circuits
Error correcting codes

ABSTRACT

We consider the multiplicative complexity of Boolean functions with multiple bits of output, studying how large a multiplicative complexity is necessary and sufficient to provide a desired nonlinearity. For so-called $\Sigma\Pi\Sigma$ circuits, we show that there is a tight connection between error correcting codes and circuits computing functions with high nonlinearity. Combining this with known coding theory results, we show that functions with n inputs and n outputs with the highest possible nonlinearity must have at least $2.32n$ AND gates. We further show that one cannot prove stronger lower bounds by only appealing to the nonlinearity of a function; we show a bilinear circuit computing a function with almost optimal nonlinearity with the number of AND gates being exactly the length of such a shortest code.

Additionally we provide a function which, for general circuits, has multiplicative complexity at least $2n - 3$.

Finally we study the multiplicative complexity of "almost all" functions. We show that every function with n bits of input and m bits of output can be computed using at most $2.5(1 + o(1))\sqrt{m2^n}$ AND gates.

1. Introduction

Cryptographic functions such as encryption functions should have high nonlinearity to be resistant against linear and differential attacks (see again [10] and the references therein). This is an explicit design criteria for modern cryptographic systems, such as AES, [12], which has been used as a benchmark for several implementations of homomorphic encryption.

In several settings, such as homomorphic encryption and secure multiparty computation (see e.g. [30] and [17]), for practicality/efficiency, the number of AND gates is significantly more important than the number of XOR gates, hence one is interested in functions with as few AND gates as possible. For many such protocols, it is not just the *number* of AND gates that matters, but also the *AND depth*. For example, in several protocols for secure multiparty computation the number of AND gates is proportional to the number of bits sent, and the AND depth corresponds to the number of rounds in the protocol (see e.g. [20]), and in typical protocols for homomorphic encryption, the norm of the noise after an AND gate is the product of the norms of the noise from the inputs, so the AND depth greatly affects the number of expensive bootstrappings, relinearizations, and/or modulus reductions which are necessary (see e.g. [13]). For more examples we refer to [1] and the references therein.

A natural question to ask is how the nonlinearity of a function and its multiplicative complexity (the number of AND gates necessary to compute it when only AND, NOT and XOR gates are used) are related to each other: how large does

* Corresponding author.
E-mail addresses: joan@imada.sdu.dk (J. Boyar), magnus@gausdalfind.dk (M.G. Find).

https://doi.org/10.1016/j.tcs.2018.02.023
0304-3975/

one measure need to be in order for the other to have at least a certain value? As stated in Section 1.1, for every desired nonlinearity, it is known exactly how many AND gates are necessary and sufficient for functions with only one output to achieve this. We study this same question for functions with multiple bits of output.

1.1. Definitions and preliminaries

Let $\mathbb{F}_2$ be the finite field of order 2 and $\mathbb{F}_2^n$ the n-dimensional vector space over $\mathbb{F}_2$.

We denote by $[n]$ the set $\{1, \ldots, n\}$. An (n, m)-function is a mapping from $\mathbb{F}_2^n$ to $\mathbb{F}_2^m$ and we refer to these as the *Boolean functions*. When $m > 1$ we say that the function is *vector valued*.

It is well known that every $(n, 1)$-function f can be written uniquely as a multilinear polynomial over $\mathbb{F}_2$

$$f(x_1, \ldots, x_n) = \sum_{X \subseteq [n]} \alpha_X \prod_{i \in X} x_i,$$

where $\alpha_X \in \{0, 1\}$ for subsets of indices. This polynomial is called the *Zhegalkin polynomial* or the *algebraic normal form* (ANF) of f. For the rest of this paper most, but not all, arithmetic will be in $\mathbb{F}_2$. We trust that the reader will find it clear whether arithmetic is in $\mathbb{F}_2$, $\mathbb{F}_{2^n}$, or $\mathbb{R}$ when not explicitly stated, and will not address it further.

The *degree* of an $(n, 1)$-function f is the largest $|X|$ such that $\alpha_X = 1$. For an (n, m)-function f, we let f_i be the $(n, 1)$-function defined by the ith output bit of f, and say that the degree of f is the largest degree of f_i for $i \in [m]$. A function is *affine* if it has degree 1, and *quadratic* if it has degree 2. For $T \subseteq [m]$ we let

$$f_T = \sum_{i \in T} f_i,$$

and for $\mathbf{v} \in \mathbb{F}_2^n$ we let $|\mathbf{v}|$ denote the *Hamming weight* of $\mathbf{v}$, that is, the number of nonzero entries in $\mathbf{v}$, and let $|\mathbf{u} + \mathbf{v}|$ be the *Hamming distance* between the two vectors $\mathbf{u}$ and $\mathbf{v}$.

Nonlinearity of Boolean Functions. We will use several facts on the nonlinearity of Boolean functions. We refer to the two chapters in [9,10] for proofs and references.

The *nonlinearity* of an $(n, 1)$-function f is the Hamming distance to the closest affine function, more precisely

$$NL(f) = 2^n - \max_{\mathbf{a} \in \mathbb{F}_2^n, b \in \mathbb{F}_2} |\{\mathbf{x} \in \mathbb{F}_2^n | \langle \mathbf{a}, \mathbf{x} \rangle + b = f(\mathbf{x})\}|,$$

where $\langle \mathbf{a}, \mathbf{x} \rangle = \sum_{i=1}^n a_i x_i$. For an (n, m)-function f, the nonlinearity is defined as

$$NL(f) = \min_{T \subseteq [m], T \neq \emptyset} \{NL(f_T)\}.$$

The nonlinearity of an (n, m)-function is always between 0 and $2^{n-1} - 2^{\frac{n}{2}-1}$. The (n, m)-functions which meet the upper bound are called *bent functions*. Bent $(n, 1)$ functions exist if and only if n is even. A standard example of a bent $(n, 1)$-function is the *inner product*, on $n = 2k$ variables, defined as:

$$IP_{2k}(x_1, \ldots, x_k, y_1, \ldots, y_k) = \langle \mathbf{x}, \mathbf{y} \rangle .$$

This function is clearly quadratic. If we identify $\mathbb{F}_2^n$ with $\mathbb{F}_{2^n}$, a standard example of a bent $(2n, n)$-function is the *finite field multiplication* function:

$$f(\mathbf{x}, \mathbf{y}) = \mathbf{x} \cdot \mathbf{y} \tag{1}$$

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

If $n = m$, $NL(f)$ is at most $2^{n-1} - 2^{\frac{n-1}{2}}$, see [11]. Functions meeting this bound are called *almost bent*. These exist only for odd n. As remarked by Carlet, this name is a bit misleading since the name indicates that they are suboptimal, which they are not. Again, if we identify $\mathbb{F}_2^n$ and $\mathbb{F}_{2^n}$, for $1 \leq i \leq \frac{n-1}{2}$ and $gcd(i, n) = 1$, the so called *Gold functions*, $G(x)$, defined as

$$G(\mathbf{x}) = \mathbf{x}^{2^i+1} = \mathbf{x} \cdot \left(\mathbf{x}^{2^i}\right) \tag{2}$$

are almost bent. This function is quadratic since the $\mathbb{F}_{2^n}$-operator $\mathbf{x} \mapsto \mathbf{x}^2$ is affine in when considered as an operator on $\mathbb{F}_2^n$, and each output bit of finite field multiplication is quadratic in the inputs, see also [10].

Multiplicative Complexity and Circuit Classes. In this paper we consider multiple classes of circuits:

- An *XOR-AND circuit* is a Boolean circuit where each of the gates is either $\oplus$ (XOR, addition in $\mathbb{F}_2$), $\wedge$ (AND, multiplication in $\mathbb{F}_2$) or the constant 1. The XOR gates may have unbounded fanin, and the AND gates have fanin 2;