New Bounds on the Multiplicative Complexity of Boolean Functions

Meltem Sönmez Turan

National Institute of Standards and Technology

Abstract

Multiplicative Complexity (MC) is defined as the minimum number of `AND` gates required to implement a function with a circuit over the basis {`AND, XOR, NOT`}. This complexity measure is relevant for many advanced cryptographic protocols such as fully homomorphic encryption, multi-party computation, and zero-knowledge proofs, where processing `AND` gates is more expensive than processing `XOR` gates. Although there is no known asymptotically efficient technique to compute the MC of a random Boolean function, bounds on the MC of Boolean functions are successfully used to to show existence of Boolean functions with a particular MC. In 2000, Boyar et al. [1] showed that, for all $n \geq 0$, at most $2^{k^2+2k+2kn+n+1}$ n-variable Boolean functions can be computed with k AND gates. This bound is used to prove the existence of a 8-variable Boolean functions with MC greater than 7. In this paper, we improve the Boyar et al. bound.

1 Introduction

Multiplicative Complexity (MC) is defined as the minimum number of `AND` gates required to implement a function with a circuit over the basis {`AND, XOR, NOT`}. This complexity measure is relevant for many advanced cryptographic protocols (e.g., [2]), fully homomorphic encryption (e.g., [3]), and zero-knowledge proofs (e.g., [4]), where processing nonlinear gates such as `AND`, `NAND`, is more expensive than processing linear gates such as `XOR`. These protocols benefit from new symmetric-key primitives that can be implemented with small number of `AND` gates (e.g., Rasta [5], LowMC [6]).

There is no known asymptotically efficient technique to compute the MC of a random Boolean function. Boyar et al. [1] showed that the MC of an n-variable random Boolean function is at least $2^{n/2} - \mathcal{O}(n)$ with high probability. For arbitrary n, it is known that under standard cryptographic assumptions, it is not possible to compute the MC in polynomial time in the length of the truth table [7]. The *degree bound* states that the MC of a Boolean function having degree d is at least $d-1$ [8]. This bound is commonly used to prove that a given Boolean function implementation is optimal.

For up to 6 variables, the MC of each Boolean function has been established in [9, 10]. There are also known bounds for special classes of Boolean functions. The MC of affine Boolean functions is zero. Mirwald and Schnorr [11] showed that the MC of a quadratic function f is k, iff f is affine equivalent to the canonical form $\bigoplus_{i=1}^{k} x_{2i-1}x_{2i}$. This implies the MC of quadratic functions is at most $\lfloor \frac{n}{2} \rfloor$. Turan and Peralta [12] improved the bounds on MC of cubic Boolean functions. Brandão et al. [13] studied the MC of symmetric Boolean functions and constructed circuits for all such functions with up to 25 variables. In 2017, Find et al. [14] characterized the Boolean functions with MC 2 by using the fact that MC is invariant with respect to affine transformations. In 2020, Çalık et al. extended the result to Boolean functions with MC up to 4 [15]. In 2022, Häner and Soeken [16] showed the MC of interval checking.

A particular value of interest is the number of n-variable Boolean functions with MC k, denoted $\lambda(n,k)$. In [1], it is shown that $\lambda(n,k) \leq 2^{k^2+2k+2kn+n+1}$. Using this bound, it is easy to see that 7 `AND` gates are not enough to compute all 8-variable Boolean functions, i.e., there exists 8-variable Boolean functions with MC at least 8. In 2002, Fischer and Peralta [17] showed that $\lambda(n,1)$ is equal to $2\binom{2^n}{3}$. In 2017, Find et al. [14] showed that

$$\lambda(n,2) = 2^n(2^n-1)(2^n-2)(2^n-4)\left(\frac{2}{21} + \frac{2^n-8}{12} + \frac{2^n-8}{360}\right). \tag{1}$$

Çalık and Turan [15] studied the Boolean functions with MC 3 and 4, and provided a closed formula for $\lambda(n,3)$ and $\lambda(n,4)$, by summing the sizes of all the affine equivalence classes with MC 3 (total of 24 classes) and 4 (total of 1277 classes).

For large values of n and k, the bound $\lambda(n,k) \leq 2^{k^2+2k+2kn+n+1}$ is essentially tight, but it is unclear to what extent this is true for small constant values of k. In this paper, we improve the Boyar et al. bound and provide new bounds on the maximum multiplicative complexity for n-variable Boolean functions.

2 Preliminaries

2.1 Boolean functions

Let $\mathbb{F}_2$ be the finite field with two elements. An n-variable Boolean function f is a mapping from $\mathbb{F}_2^n$ to $\mathbb{F}_2$. Let B_n be the set of n-variable Boolean functions and B_n^c be the set of n-variable cubic Boolean functions.

The *algebraic normal form* (ANF) of f is the multivariate polynomial $f(x_1,\ldots,x_n) = \sum_{u\in\mathbb{F}_2^n} a_u x^u$, where $a_u \in \mathbb{F}_2$ and $x^u = x_1^{u_1}x_2^{u_2}\cdots x_n^{u_n}$ is a *monomial* containing the variables x_i where $u_i = 1$. The degree of the monomial x^u is the number of variables appearing in x^u. The *degree* of a Boolean function, denoted $deg(f)$, is the highest degree among the monomials appearing in its ANF.

Two functions $f, g \in \mathcal{B}_n$ are *affine equivalent* if f can be written as

$$f(\mathbf{x}) = g(A\mathbf{x}+\mathbf{a}) + \mathbf{b}^\top\mathbf{x} + c, \tag{2}$$

where A is a non-singular $n \times n$ matrix over $\mathbb{F}_2$, $\mathbf{a}, \mathbf{b}$ are column vectors in $\mathbb{F}_2^n$, and $c \in \mathbb{F}_2$. We use $[f]$ to denote the affine equivalence class of the function f. Degree and MC are invariant under affine transformations.

2.2 Boolean Circuits

A *Boolean circuit* C with n inputs and m outputs is a directed acyclic graph, where the inputs and the gates are the nodes, and the edges correspond to the Boolean-valued *wires*. The *fanin* and *fanout* of a node is the number of wires going in and out of the node, respectively. The nodes with fanin zero are called the *input nodes* and are labeled with an input variable from $\{x_1,\ldots,x_n\}$. The circuits considered in this study only contain gates from the complete basis {`AND, XOR, NOT`} and have exactly one node with fanout zero (i.e., $m=1$), which is called the *output node*. For our purposes, we assume `AND` gates have fan-in two, but `XOR` gates have arbitrary fan-in > 0.

3 Number of Boolean functions with MC k

3.1 Previous results

Boyar et al. [1] showed that $\lambda(n,k) \leq 2^{k^2+2k+2kn+n+1}$, for all $n \geq 0$. To compute this bound, the authors considered an abstraction of Boolean circuits having binary `AND` gates and `XOR` gates with unbounded inputs. Each `AND` gate is assumed to input a subset of input variables, outputs of `AND` gates (that are topologically located before the gate) and the constant function $\mathbf{1}$, *i.e.*, for the ith AND gate a_i the (right and the left) input is subset of $\{x_1, \dots, x_n, a_1, a_2, \dots, a_{i-1}, \mathbf{1}\}$. Hence, for a_i, there are $2^{n+1+(i-1)} = 2^{n+i}$ possible choices for its left and right inputs. The bound increases by $2(n+k)+3$ for each addition of the new `AND` gate to the circuit.

The exact values of $\lambda(n,k)$ are known for the following values of n and k:

- [17] $\lambda(n,1) = 2\binom{2^n}{3}$
- [7] $\lambda(n,2) = 2^n(2^n-1)(2^n-2)(2^n-4)\left(\frac{2}{21} + \frac{2^n-8}{12} + \frac{2^n-8}{360}\right)$.
- [15] $\lambda(n,3) = \sum_{d=4}^{6}\left(2^{n-d}\prod_{i=0}^{d-1}\frac{2^n-2^i}{2^d-2^i}\beta(d,3)\right)$ where

$$\begin{aligned}\beta(4,3) &= 32\,768,\\ \beta(5,3) &= 775\,728\,128,\\ \beta(6,3) &= 183\,894\,007\,808.\end{aligned}$$

- [15] $\lambda(n,4) = \sum_{d=5}^{8}\left(2^{n-d}\prod_{i=0}^{d-1}\frac{2^n-2^i}{2^d-2^i}\beta(d,4)\right)$ where

$$\begin{aligned}\beta(5,4) &= 3\,515\,396\,096,\\ \beta(6,4) &= 7\,944\,313\,921\,970\,176,\\ \beta(7,4) &= 8\,217\,135\,092\,528\,316\,416,\\ \beta(8,4) &= 5\,502\,415\,308\,673\,798\,144.\end{aligned}$$

Table 1 shows the gap between the Boyar *et al.* bound and the exact number of functions with MC up to 4.

MC	Bound	$n=6$	$n=7$	$n=8$	$n=9$	$n=10$	$n=11$	$n=12$	$n=13$	$n=14$	$n=15$	$n=16$
1	Exact	16.34	19.38	22.38	25.40	28.41	31.41	34.41	37.41	40.41	43.41	46.41
1	Bound	22	25	28	31	34	37	40	43	46	49	52
2	Exact	26.13	31.30	36.38	41.42	46.44	51.45	56.45	61.45	66.46	71.46	76.46
2	Bound	39	44	49	54	59	64	69	74	79	84	89
3	Exact	38.03	45.64	52.92	60.05	67.12	74.15	81.17	88.18	95.18	102.18	109.18
3	Bound	58	65	72	79	86	93	100	107	114	121	128
4	Exact	52.81	63.15	71.94	80.29	88.46	96.56	104.63	112.70	120.82	129.02	137.35
4	Bound	79	88	97	106	115	124	133	142	151	160	169

Table 1: Number of Boolean functions with MC 1, 2, 3, and 4 compared to the Boyar et al. bound [1] on a log scale with base 2

3.2 Improving the Boyar et al. bound

Next, we present two observations on Boolean circuits to improve the bound.

1. *Elimination of equivalent inputs* Let f_1 and f_2 be n-bit Boolean functions representing the left and right inputs to an `AND` gate, respectively, and let f_3 be the Boolean function `XOR`ed to the output to the `AND` gate. The output of the `AND` gate is $f_1 * f_2 + f_3$. It is easy to see that the following inputs also produce the same output as f_1,f_2, and f_3.

$$\begin{aligned}(f_1, f_2, f_3) &\rightarrow f_1 * f_2 + f_3\\ (f_2, f_1, f_3) &\rightarrow f_1 * f_2 + f_3\\ (f_1 + f_2, f_2, f_3 + f_2) &\rightarrow f_1 * f_2 + f_2 + f_2 + f_3 = f_1 * f_2 + f_3\\ (f_2, f_1 + f_2, f_3 + f_2) &\rightarrow f_2 * f_1 + f_2 + f_2 + f_3 = f_1 * f_2 + f_3\\ (f_1, f_2 + f_1, f_3 + f_1) &\rightarrow f_2 * f_1 + f_1 + f_3 + f_1 = f_1 * f_2 + f_3\\ (f_2 + f_1, f_1, f_3 + f_1) &\rightarrow f_2 * f_1 + f_1 + f_3 + f_1 = f_1 * f_2 + f_3\end{aligned}$$

   In the Boyar et al. bound, each of these cases are counted separately.

2. *Elimination of the constant 1 function.* Boolean functions can be partitioned into those f for which $f(0) = 0$ and those f for which $f(0) = 1$. One set can be mapped bijectively into the other by the transformation $g(\mathbf{x}) = f(\mathbf{x})+1$. A function $f(\mathbf{x})$ for which $f(0) = 0$ can be computed by a circuit which is both optimal with respect to multiplicative complexity and has no negations. Thus, considering circuits that do not have the constant `1` as input would produce the same set of Boolean functions. Boyar et al. bound computes the two inputs (f_1, f_2, f_3), and $(f_1, f_2 + 1, f_3 + f_1)$ separately, although they both results in the same output.

$$\begin{aligned}(f_1, f_2, f_3) &\rightarrow f_1 * f_2 + f_3\\ (f_1 + \mathtt{1}, f_2, f_3 + f_2) &\rightarrow f_1 * f_2 + f_2 + f_3 + f_2 = f_1 * f_2 + f_3\\ (f_1, f_2 + \mathtt{1}, f_3 + f_1) &\rightarrow f_1 * f_2 + f_1 + f_3 + f_1 = f_1 * f_2 + f_3\\ (f_1 + \mathtt{1}, f_2 + \mathtt{1}, f_3 + f_1 + f_2) &\rightarrow f_2 * f_1 + f_1 + f_2 + f_3 + f_1 + f_2 = f_1 * f_2 + f_3\end{aligned}$$

Using the observations given above, the bound on the number of n-variable Boolean functions that can be generated using k `AND` gates, over the basis $\{\mathtt{AND}, \mathtt{XOR}, \mathtt{NOT}\}$, can be improved, by a factor of 24 for each `AND` gate. In other words, the function $f_1 * f_2 + f_3$ can be generated using 24 different choices of inputs for each `AND` gate, and instead of trying all 24 inputs (each combination of cases from (1) and (2)), it is possible to try only one of the inputs.

Theorem 3.1 *The number of n-variable Boolean functions that can be generated with k-`AND` gates is at most*

$$\begin{aligned}\lambda(n, k) &\leq 2^{n+k+1} \prod_{i=1}^{k} \frac{1}{24}(2^{n+i+1})^2,\\ &\leq 2^{k^2+2nk+n-k+1}3^{-k}.\end{aligned}$$

4 Discussion

In this note, we improved the Boyar et al. [1] the bound on the number of n-variable Boolean functions that can be generated using k `AND` gates by a factor of $2^{3k}3^k$ (by a factor of 24 for each `AND` gate), which can be used to provide bounds on the maximum MC across all n-variable Boolean functions.

Acknowledgments

The author thanks L. Brandão, and R. Peralta for helpful discussions and suggestions. The author also thanks the anonymous reviewers for their useful comments.

References

[1] Joan Boyar, René Peralta, and Denis Pochuev. On the Multiplicative Complexity of Boolean Functions over the Basis $(\wedge, \oplus, 1)$. *Theor. Comput. Sci.*, 235(1):43–57, 2000.

[2] Vladimir Kolesnikov and Thomas Schneider. Improved Garbled Circuit: Free XOR Gates and Applications. In Luca Aceto, Ivan Damgård, Leslie Ann Goldberg, Magnús M. Halldórsson, Anna Ingólfsdóttir, and Igor Walukiewicz, editors, *Automata, Languages and Programming, 35th International Colloquium, ICALP*, volume 5126 of *Lecture Notes in Computer Science*, pages 486–498. Springer, 2008.

[3] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. (Leveled) fully homomorphic encryption without bootstrapping. In Shafi Goldwasser, editor, *Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012*, pages 309–325. ACM, 2012.

[4] Joan Boyar, Ivan Damgård, and René Peralta. Short Non-Interactive Cryptographic Proofs. *J. Cryptology*, 13(4):449–472, 2000.

[5] Christoph Dobraunig, Maria Eichlseder, Lorenzo Grassi, Virginie Lallemand, Gregor Leander, Eik List, Florian Mendel, and Christian Rechberger. Rasta: A Cipher with Low ANDdepth and Few ANDs per Bit. In *CRYPTO (1)*, volume 10991 of *Lecture Notes in Computer Science*, pages 662–692. Springer, 2018.

[6] Martin R. Albrecht, Christian Rechberger, Thomas Schneider, Tyge Tiessen, and Michael Zohner. Ciphers for MPC and FHE. In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I*, volume 9056 of *Lecture Notes in Computer Science*, pages 430–454. Springer, 2015.

[7] Magnus Gausdal Find. On the Complexity of Computing Two Nonlinearity Measures. In *Computer Science - Theory and Applications - 9th International Computer Science Symposium in Russia, CSR 2014, Moscow, Russia, June 7-11, 2014. Proceedings*, pages 167–175, 2014.

[8] C. P. Schnorr. The Multiplicative Complexity of Boolean Functions. In Teo Mora, editor, *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes (AAECC 1988)*, volume 357 of *LNCS*, pages 45–58, Berlin, Heidelberg, 1989. Springer Berlin Heidelberg.

[9] Meltem Turan Sönmez and René Peralta. *The Multiplicative Complexity of Boolean Functions on Four and Five Variables*, pages 21–33. Springer International Publishing, Cham, 2015.

[10] Çağdaş Çalık, Meltem Sönmez Turan, and René Peralta. The Multiplicative Complexity of 6-variable Boolean Functions. *Cryptogr. Commun.*, 11(1):93–107, 2019.

[11] Roland Mirwald and Claus-Peter Schnorr. The Multiplicative Complexity of Quadratic Boolean Forms. *Theor. Comput. Sci.*, 102(2):307–328, 1992.

[12] Meltem Sonmez Turan and Rene Peralta. On the Multiplicative Complexity of Cubic Boolean Functions. The 6th International Workshop on Boolean Functions and their Applications (BFA), 2021.

[13] Luís T. A. N. Brandão, Çağdaş Çalık, Meltem Sönmez Turan, and René Peralta. Upper Bounds on the Multiplicative Complexity of Symmetric Boolean Functions. *Cryptogr. Commun.*, 11(6):1339–1362, 2019.

[14] Magnus Gausdal Find, Daniel Smith-Tone, and Meltem Sönmez Turan. The Number of Boolean Functions with Multiplicative Complexity 2. *IJICoT*, 4(4):222–236, 2017.

[15] Çağdaş Çalık, Meltem Sönmez Turan, and René Peralta. Boolean Functions with Multiplicative Complexity 3 and 4. *Cryptogr. Commun.*, 12(5):935–946, 2020.

[16] Thomas Häner and Mathias Soeken. The multiplicative complexity of interval checking, 2022.

[17] M. J. Fischer and R. Peralta. Counting Predicates of Conjunctive Complexity One. *Yale Technical Report 1222*, February 2002.