Slicing Boolean Functions with Inner Products

Pierrick Méaux and Tim Seuré

University of Luxembourg, Esch-sur-Alzette, Luxembourg
pierrick.meaux@uni.lu, tim.seure@uni.lu

Abstract. Boolean functions with additional structure play an important role in symmetric cryptography, both for achieving strong cryptographic properties and for enabling efficient implementations. Recent works on homomorphic-friendly symmetric primitives, especially in the context of Hybrid Homomorphic Encryption, highlighted the interest of Boolean functions whose evaluation can be decomposed according to structured partitions of the Boolean cube. A classical example is given by Hamming-weight decompositions, which underlie symmetric and weightwise degree-d Boolean functions. In this work, we generalize this viewpoint by replacing the Hamming weight with a general integer linear form. Given a vector $v \in \mathbb{Z}^n$, we partition the Boolean cube $\mathbb{F}_2^n$ by grouping the Boolean vectors $x \in \mathbb{F}_2^n$ according to the value of $\langle v, x \rangle$ into so-called v-slices, and study functions that have bounded degree on each of them. We study how many such slices are needed to describe a given function, and provide bounds and structural properties for the partitions induced by integer vectors. We also show how this representation leads to a homomorphic evaluation strategy in a GSW-like setting, together with noise estimates for the resulting ciphertexts. Finally, we generalize several symmetric and weightwise degree-d Boolean functions using different vector families, and experimentally evaluate their algebraic degree, algebraic immunity, and nonlinearity. The results show that direct generalizations of symmetric functions often lose cryptographic strength, while generalized weightwise degree-d constructions lead to richer and more promising behavior.

Keywords: Boolean Functions · Fully Homomorphic Encryption · Slice Complexity · Symmetric Cryptography

1 Introduction

Boolean functions are central building blocks in symmetric cryptography. Their cryptographic strength is measured through parameters linked to known attacks, such as algebraic immunity [41] against algebraic attacks [17], resilience and nonlinearity against correlation attacks [50]. These parameters, however, are only part of the story: in many cryptographic protocols, the function must also admit efficient evaluation.

This is particularly relevant in Hybrid Homomorphic Encryption (HHE) [44], where a symmetric primitive is evaluated homomorphically to convert symmetric

ciphertexts into fully homomorphic encryption ciphertexts [24]. Several HHE-oriented designs thus aim to make the symmetric encryption scheme homomorphic-friendly. In stream-cipher-based constructions such as FLIP [37] and FiLIP [36], and more recently in constructions based on filtered Linear Feedback Shift Registers (LFSR) [10, 12], the main homomorphic cost is the evaluation of one filtering Boolean function, while the remaining operations are linear. In the LFSR case, the noise introduced by the linear recurrence remains controlled, typically growing by at most a factor proportional to the state size; see, for instance, [2, 12].

Sparse and Hamming-Weight-Based Structures. Early homomorphic-friendly constructions exploited sparse algebraic representations. In FLIP, the filtering function is expressed as a direct sum of monomials, allowing functions of relatively large algebraic degree while keeping homomorphic evaluation practical: each monomial is evaluated by a multiplication chain, and the results are then summed. The first FiLIP constructions used a similar direct-sum approach [36], whose cryptographic properties were further investigated in [9].

Another important direction relies on symmetry and Hamming-weight decompositions. Symmetric Boolean functions, whose value depends only on the Hamming weight of the input, admit dedicated homomorphic evaluation techniques based on MUX circuits [14, 29]. Threshold functions can be evaluated even more efficiently using specialized methods [16,39]. These techniques connect naturally with XOR-threshold functions, used in several low-complexity constructions, including variants of Goldreich's pseudorandom generator [1, 3, 28] and FiLIP. They also motivated the study of direct sums of symmetric functions for FiLIP [26].

Symmetric functions suggest a broader Hamming-weight-slice viewpoint, formalized by weightwise degree-d functions [27]. There, the Boolean cube is partitioned by Hamming weight, and a low-degree Boolean function is applied on each slice. This family includes the hidden-weight-bit function (HWBF) [5], studied cryptographically in [35, 55], as well as recent classes of weightwise degree-1 and degree-2 functions [11, 38, 40]. Such structures also appear in the HHE-oriented cipher Nostalgia [12].

Generalized Slice Decompositions. The above suggests that the useful structure is not symmetry itself, but rather a partition of the Boolean cube on which the function has a simple local description. For symmetric and weightwise degree-d functions, the slices are given by the Hamming weight, equivalently by the integer inner product $\langle 1_n, x\rangle$ for $x \in \mathbb{F}_2^n$. In this paper, we replace 1_n by an arbitrary vector $v \in \mathbb{Z}^n$. This partitions the Boolean cube into v-slices, defined by the integer values of $\langle v, x\rangle$, and we study Boolean functions whose restrictions to these slices have bounded degree.

This leads to three questions. Combinatorially, how many slices can an integer vector induce, and how can this number be bounded? From the Boolean-function viewpoint, how many slices are needed to represent a given function with bounded local degree? And cryptographically, can such generalized slices yield structured Boolean functions with strong parameters and efficient homomorphic evaluation?

Structured Boolean Functions in Cryptography. Our work fits into the broader study of structured Boolean functions in cryptography, where additional algebraic or combinatorial structure is used to combine strong cryptographic parameters with efficient implementation or analysis. Examples include rotation-symmetric Boolean functions [18, 23, 42, 51] and Maiorana–McFarland-type constructions and variants [30, 52, 53, 54]. Here, the structure comes from partitions of the Boolean cube induced by integer inner products.

Our Contributions. We generalize the Hamming-weight viewpoint behind symmetric and weightwise degree-d Boolean functions by grouping each $x \in \mathbb{F}_2^n$ according to the integer value of $\langle v, x\rangle$, for a vector $v \in \mathbb{Z}^n$. This gives a partition of the Boolean cube into v-slices, and we study functions whose restrictions to these slices have degree at most d. Our contributions are organized as follows:

- Section 3 formalizes degree-d slice decompositions and gives a homomorphic evaluation strategy in GSW-like schemes, together with a noise bound.
- Section 4 studies the partitions induced by integer vectors: how many slices a vector can create, how this number can be bounded, and which reductions preserve it.
- Section 5 studies how many slices are needed for a degree-d slice description of a given Boolean function. A main result is a threshold around degree $n/2$: for every fixed $\varepsilon > 0$, almost all Boolean functions require exponentially many slices when $d \le (1/2 - \varepsilon)n$, while linearly many slices suffice for every function once $d \ge \lfloor n/2 \rfloor$.
- Finally, Section 6 builds generalized symmetric and weightwise degree-d functions from several vector families. Our experiments on algebraic degree, algebraic immunity, and nonlinearity suggest that direct generalizations of symmetric functions tend to lose cryptographic strength, whereas generalized weightwise degree-d constructions show richer and more promising behavior.

2 Preliminaries

Throughout the paper, we denote by $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$, and $\mathbb{F}_2$ the natural numbers (including zero), the ring of integers, the field of rational numbers, the field of real numbers, and the field with two elements, respectively. Unless otherwise stated, $n \ge 1$ will denote a fixed positive integer. For two integers $a \le b$, we write $[a, b] = \{a, a+1, \ldots, b\}$. If R is a commutative ring, then we write $R_{\neq 0}$ for the set of nonzero elements of R. We denote by R^n the set of length-n vectors with entries in R, and for $v \in R^n$ we implicitly denote its entries as $v = (v_1, \ldots, v_n)$. The all-ones vector and the zero vector are denoted by 1_n and 0_n, respectively. Given two vectors $v, w \in R^n$, we write $\langle v, w\rangle = \sum_{i=1}^{n} v_i w_i \in R$. When writing $\langle v, x\rangle$ with $v \in \mathbb{Z}^n$ and $x \in \mathbb{F}_2^n$, we identify x with its representative in $\{0, 1\}^n \subseteq \mathbb{Z}^n$.

We recall the notation and terminology for Boolean functions used throughout the paper; for further background, see [8]. We denote by $e_1, \ldots, e_n \in \mathbb{F}_2^n$ the standard basis vectors of $\mathbb{F}_2^n$. For a set $A \subseteq \mathbb{F}_2^n$, we write $\mathbb{1}_A : \mathbb{F}_2^n \to \mathbb{F}_2$ for its

indicator function, defined by $\mathbb{1}_A(x) = 1$ if and only if $x \in A$. If $A = \{a\}$ is a singleton, we write $\mathbb{1}_a$ instead of $\mathbb{1}_{\{a\}}$.

Definition 1 (Boolean function, support). *A* Boolean function *in n variables is a function $f : \mathbb{F}_2^n \to \mathbb{F}_2$. The set of Boolean functions in n variables is denoted by $\mathcal{B}_n$. The* support *of a Boolean function $f \in \mathcal{B}_n$ is the set* $\mathsf{supp}(f) = \{x \in \mathbb{F}_2^n : f(x) = 1\}$.

Definition 2 (Hamming weight, slices). *The* Hamming weight *of a Boolean vector $x \in \mathbb{F}_2^n$, denoted $\mathsf{w_H}(x)$, is the number of coordinates of x equal to 1, that is, $\mathsf{w_H}(x) = |\{i \in [1, n] : x_i = 1\}|$. For an integer $k \in [0, n]$, the k-*th slice *of the Boolean cube $\mathbb{F}_2^n$ is the set $\mathsf{E}_{k,n} = \{x \in \mathbb{F}_2^n : \mathsf{w_H}(x) = k\}$.*

Definition 3 (Algebraic normal form, algebraic degree). *The* algebraic normal form (ANF) *of a Boolean function $f \in \mathcal{B}_n$ is the unique representation of f as an element of the quotient ring $\mathbb{F}_2[x_1, \ldots, x_n]/\left(x_1^2 - x_1, \ldots, x_n^2 - x_n\right)$:*

$$f(x) = \sum_{I \subseteq [1,n]} a_I \prod_{i \in I} x_i,$$

with coefficients $a_I \in \mathbb{F}_2$, where the equality is meant for every $x \in \mathbb{F}_2^n$. The (algebraic) degree *is then defined as $\mathsf{deg}(f) = \max\{|I| : I \subseteq [1, n] \text{ and } a_I \neq 0\}$ for $f \neq 0$ and $\mathsf{deg}(0) = 0$. For a subset $A \subseteq \mathbb{F}_2^n$, the* (algebraic) degree *of a map $g : A \to \mathbb{F}_2$ is defined as $\mathsf{deg}(g) = \min\{\mathsf{deg}(f) : f \in \mathcal{B}_n \text{ and } f(x) = g(x) \text{ for every } x \in A\}$.*

Definition 4 (Numerical normal form, numerical degree). *The* numerical normal form (NNF) *of a Boolean function $f \in \mathcal{B}_n$ is the unique representation of f as an element of the quotient ring $\mathbb{R}[x_1, \ldots, x_n]/\left(x_1^2 - x_1, \ldots, x_n^2 - x_n\right)$:*

$$f(x) = \sum_{I \subseteq [1,n]} \alpha_I \prod_{i \in I} x_i,$$

with coefficients $\alpha_I \in \mathbb{R}$, where the equality is meant for every $x \in \{0, 1\}^n$ and $f(x)$ is identified with its value in $\{0, 1\} \subseteq \mathbb{R}$. The numerical degree *of f is defined as $\deg_{\mathsf{num}}(f) = \max\{|I| : I \subseteq [1, n] \text{ and } \alpha_I \neq 0\}$ for $f \neq 0$ and $\deg_{\mathsf{num}}(0) = 0$.*

Property 1 ([8]). Every Boolean function $f \in \mathcal{B}_n$ satisfies $\deg_{\mathsf{num}}(f) \geq \mathsf{deg}(f)$.

Definition 5 (Annihilator, algebraic immunity [41]). *Let $f \in \mathcal{B}_n$ be a Boolean function. A nonzero Boolean function $0 \neq g \in \mathcal{B}_n$ is called an* annihilator *of f if $f(x)g(x) = 0$ for every $x \in \mathbb{F}_2^n$. The* algebraic immunity (AI) *of f is defined as:*

$$\mathsf{AI}(f) = \min\{\mathsf{deg}(g) : 0 \neq g \in \mathcal{B}_n \text{ is an annihilator of } f \text{ or of } f + 1\}.$$

Definition 6 (Nonlinearity). *The* nonlinearity (NL) *of a Boolean function $f \in \mathcal{B}_n$, denoted $\mathsf{NL}(f)$, is the Hamming distance between f and the space of affine functions in $\mathcal{B}_n$:*

$$\mathsf{NL}(f) = \min\{|\mathsf{supp}(f + g)| : g \in \mathcal{B}_n \text{ and } \mathsf{deg}(g) \leq 1\}.$$

3 Slice Complexities and Homomorphic Evaluation

This section introduces the central notions of the paper. Symmetric Boolean functions, and more generally weightwise degree-d functions [27,38], are naturally described by partitioning the Boolean cube according to the Hamming weight. We generalize this viewpoint by replacing the Hamming weight with the integer inner product against an arbitrary vector $v \in \mathbb{Z}^n$. This gives rise to a hierarchy of slice complexities $\kappa_d(f)$, which measure how many vector-induced slices are needed to describe f when the restriction of f to each slice is allowed to have degree at most d. We also explain why such decompositions are relevant for the homomorphic evaluation of Boolean functions. General theoretical properties of these complexities are studied later in Section 5.

3.1 Constant on Vector Slices

Let us start by recalling the definition of symmetric Boolean functions, a class of functions which has been thoroughly studied in the context of cryptography; see, *e.g.*, [4, 6, 7, 9, 13, 22, 31, 32, 33, 34, 46, 47, 48, 49].

Definition 7 (Symmetric Boolean functions, *e.g.*, [4]). *A Boolean function $f \in \mathcal{B}_n$ is called* symmetric *if for all $x, x' \in \mathbb{F}_2^n$ with $\mathsf{w_H}(x) = \mathsf{w_H}(x')$, we have $f(x) = f(x')$.*

Since the Hamming weight of a Boolean vector $x \in \mathbb{F}_2^n$ can be expressed as the inner product $\langle 1_n, x \rangle$ (over the integers), symmetric functions are precisely those Boolean functions which are constant on the Hamming-weight slices $\mathsf{E}_{k,n} = \{x \in \mathbb{F}_2^n : \langle 1_n, x \rangle = k\}$ for every $k \in [0, n]$. A natural generalization of symmetric functions is obtained by replacing the all-1 vector 1_n with an arbitrary vector $v \in \mathbb{Z}^n$.

Definition 8 (Slices induced by a vector). *We define the* image *of a vector $v \in \mathbb{Z}^n$ as the image of the map $\langle v, \cdot \rangle : \mathbb{F}_2^n \to \mathbb{Z}$, that is, $\mathrm{Im}(v) = \{\langle v, x \rangle : x \in \mathbb{F}_2^n\}$, where the inner product is taken over the integers. For an integer $k \in \mathbb{Z}$, we define the k-*th *v-*slice *as the set $\mathsf{E}_k^v = \{x \in \mathbb{F}_2^n : \langle v, x \rangle = k\}$. The number of nonempty v-slices is denoted by $\kappa(v) = |\mathrm{Im}(v)|$.*

Example 1. For instance, for the all-1 vector $v = 1_n \in \mathbb{Z}^n$, we have $\mathrm{Im}(1_n) = [0, n]$, and therefore $\kappa(1_n) = n + 1$. For $k \in [0, n]$, the k-th v-slice is the set of Boolean vectors of Hamming weight k.

We provide in Figure 1 and Figure 2 some illustrations of vector-induced slices.

Definition 9 (Constant on v-slices). *Let $v \in \mathbb{Z}^n$. A Boolean function $f \in \mathcal{B}_n$ is called* constant on v-slices *if for all $k \in \mathrm{Im}(v)$ and all $x, x' \in \mathsf{E}_k^v$, we have $f(x) = f(x')$.*

Example 2. The symmetric Boolean functions are precisely those Boolean functions which are constant on 1_n-slices.

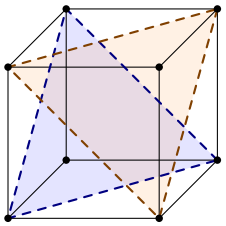

Fig. 1. Slices induced by $v = 1_3$ containing more than one element.

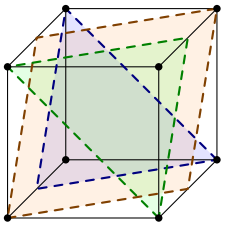

Fig. 2. Slices induced by $v = (1, 1, 2)$ containing more than one element.

If $f \in \mathcal{B}_n$ is constant on v-slices, then we find a constant $c_k \in \mathbb{F}_2$ for every $k \in \mathrm{Im}(v)$ such that $f(x) = c_k$ for every $x \in \mathsf{E}_k^v$. Consequently, we can evaluate f at any input $x \in \mathbb{F}_2^n$ by computing $f(x) = \sum_{k \in \mathrm{Im}(v)} \mathbb{1}_{\mathsf{E}_k^v}(x) \cdot c_k$. Contrary to the case of symmetric functions, the question here is not whether f can be expressed in the above form, but rather how efficiently this can be done, that is, how small the number $\kappa(v)$ of v-slices can be made.

Definition 10 (Constant slice complexity). *For a Boolean function $f \in \mathcal{B}_n$, the* constant slice complexity *is defined as follows:*

$$\kappa_0(f) = \min\{\kappa(v) : v \in \mathbb{Z}^n \text{ and } f \text{ is constant on } v\text{-slices}\}.$$

Remark 1. Note that for a Boolean function $f \in \mathcal{B}_n$, the constant slice complexity $\kappa_0(f)$ is well-defined since there always exists a vector $v \in \mathbb{Z}^n$ such that f is constant on v-slices. For instance, if we take $v = \left(1, 2, 4, \ldots, 2^{n-1}\right)$, then each v-slice contains exactly one Boolean vector, and therefore f is constant on it. In particular, this shows that $\kappa_0(f) \leq 2^n$ for every $f \in \mathcal{B}_n$ since $\mathrm{Im}(v) = [0, 2^n - 1]$.

Remark 2. Assume that a Boolean function $f \in \mathcal{B}_n$ is constant on v-slices for some $v \in \mathbb{Z}^n$. If $v_i = 0$ for some $i \in [1, n]$, then the v-slices do not distinguish between inputs that differ only in the i-th coordinate: for every $x \in \mathbb{F}_2^n$, we have $\langle v, x\rangle = \langle v, x + e_i\rangle$. Hence $f(x) = f(x + e_i)$ for all $x \in \mathbb{F}_2^n$, so f does not depend on the i-th variable. Thus, in the constant-slice setting, it is natural to restrict to vectors with no zero entries.

Example 3. Let $f \in \mathcal{B}_n$ be given by $f(x) = x_1x_2$. We claim that $\kappa_0(f) = 3$. First, consider the vector $v = (1, 1, 0, \ldots, 0) \in \mathbb{Z}^n$. Its nonempty slices are $\mathsf{E}_0^v = \{x \in \mathbb{F}_2^n : x_1 = 0 = x_2\}$, $\mathsf{E}_1^v = \{x \in \mathbb{F}_2^n : x_1 + x_2 = 1\}$, and $\mathsf{E}_2^v = \{x \in \mathbb{F}_2^n : x_1 = 1 = x_2\}$. Thus $\kappa(v) = 3$. Moreover, f is constant on these three slices, with values 0, 0, and 1, respectively. Hence $\kappa_0(f) \leq \kappa(v) = 3$. Conversely, suppose that f is constant on w-slices for some $w \in \mathbb{Z}^n$ with $\kappa(w) \leq 2$. Since $f(0_n) = f(e_1) = f(e_2) = 0$ and $f(e_1 + e_2) = 1$, the vector $e_1 + e_2$ cannot lie in the same w-slice as any of 0_n, e_1, or e_2. Since there are at most two nonempty w-slices, the vectors 0_n, e_1, and e_2 must all lie in the other slice. Thus, $0 = \langle w, 0_n\rangle = \langle w, e_1\rangle = \langle w, e_2\rangle$, so $w_1 = 0 = w_2$. But then $\langle w, e_1 + e_2\rangle = w_1 + w_2 = 0$, so $e_1 + e_2$ lies in the same w-slice as 0_n, a contradiction. It follows that $\kappa_0(f) = 3$.

3.2 Degree-d on Vector Slices

The notion of weightwise degree-d functions, introduced in [27] and further analyzed in [38], generalizes symmetry by allowing bounded-degree behavior on each Hamming-weight slice. We recall this definition and then extend it to general vector-induced slices.

Definition 11 (Weightwise degree-d functions, *e.g.*, [27, 38]). *Let $d \geq 0$ be an integer. A Boolean function $f \in \mathcal{B}_n$ is called* weightwise degree-d *if there exist Boolean functions $f_k \in \mathcal{B}_n$ ($k \in [0, n]$) of degree at most d such that for every $x \in \mathbb{F}_2^n$, we have:*

$$f(x) = \sum_{k=0}^{n} \mathbb{1}_{\mathsf{E}_{k,n}}(x) \cdot f_k(x).$$

Definition 12 (Degree-d on v-slices). *Let $v \in \mathbb{Z}^n$ be an integer vector and $d \geq 0$ be an integer. A Boolean function $f \in \mathcal{B}_n$ is called* degree-d on v-slices *if there exist Boolean functions $f_k \in \mathcal{B}_n$ ($k \in \mathrm{Im}(v)$) of degree at most d such that for every $x \in \mathbb{F}_2^n$, we have:*

$$f(x) = \sum_{k \in \mathrm{Im}(v)} \mathbb{1}_{\mathsf{E}_k^v}(x) \cdot f_k(x).$$

We are again interested in minimizing the number of slices needed for such a decomposition.

Definition 13 (Degree-d slice complexity). *Let $d \geq 0$ be an integer. For a Boolean function $f \in \mathcal{B}_n$, the* degree-d slice complexity *is defined as follows:*

$$\kappa_d(f) = \min\{\kappa(v) : v \in \mathbb{Z}^n \text{ and } f \text{ is degree-}d \text{ on } v\text{-slices}\}.$$

Remark 3. As in the constant-slice setting, the degree-d slice complexity $\kappa_d(f)$ is well-defined for every Boolean function $f \in \mathcal{B}_n$ and every integer $d \geq 0$. Indeed, again taking $v = \left(1, 2, 4, \ldots, 2^{n-1}\right)$, f is constant (and hence of degree at most d) on each v-slice. In particular, this shows that $\kappa_d(f) \leq 2^n$.

Remark 4. In contrast to the constant-slice setting, zero entries in the vector v are not necessarily degenerate for degree-d slice decompositions. Indeed, if $v_i = 0$, then the v-slices do not distinguish between inputs that differ only in the i-th coordinate, but the degree-d functions f_k may still depend on the i-th coordinate. Thus, for $d \geq 1$, vectors with zero entries can still be useful.

3.3 Allowing Real Vectors

Throughout the paper, vector slices and the quantities derived from them, such as $\kappa(v)$ and the slice complexities $\kappa_d(f)$, are defined using integer vectors $v \in \mathbb{Z}^n$. One could *a priori* formulate the same notions using real vectors $v \in \mathbb{R}^n$. The following proposition shows that this would not change the induced partitions of the Boolean cube. Thus, working over $\mathbb{Z}^n$ is not an additional restriction.

Proposition 1. *Let $v \in \mathbb{R}^n$ be a real vector. Then there exists an integer vector $w \in \mathbb{Z}^n$ such that for all $x, x' \in \mathbb{F}_2^n$, we have:*

$$\langle v, x\rangle = \langle v, x'\rangle \iff \langle w, x\rangle = \langle w, x'\rangle.$$

Proof. A proof is provided in the full version of the paper.[1] □

3.4 Homomorphic Evaluation

Homomorphic Encryption (HE) allows one to evaluate algorithms directly on encrypted data; we say that such algorithms are evaluated *homomorphically*. The evaluation is necessarily input-oblivious: the sequence of homomorphic operations may depend on the public description of the algorithm, but not on the encrypted input values. We explain how Boolean functions given by v-slice decompositions can be evaluated homomorphically in a GSW-like HE setting [14, 15, 20, 25].

In this setting, ciphertexts carry noise, modeled here by a variance parameter. Homomorphic operations increase this variance, so bounding the variance of the final ciphertext is necessary to guarantee correct decryption. Our analysis uses standard noise-propagation rules for homomorphic addition, multiplication, and multiplexers (MUX). Recall that $\mathsf{MUX}(x, a, b) = a$ if $x = 0$ and $\mathsf{MUX}(x, a, b) = b$ if $x = 1$. Given ciphertexts C_1, C_2, C_3 encrypting bits $x_1, x_2, x_3 \in \mathbb{F}_2$, we write $C_1 + C_2$, $C_1 \times C_2$, and $\mathsf{MUX}(C_1, C_2, C_3)$ for ciphertexts obtained homomorphically from C_1, C_2, C_3 and encrypting $x_1 + x_2$, $x_1 x_2$, and $\mathsf{MUX}(x_1, x_2, x_3)$, respectively.

We use the following assumption, adapted from Proposition 2 of [29].

Assumption 1. *For $i \in [1, 3]$, let C_i be an encryption of $x_i \in \mathbb{F}_2$ with noise variance V_i. Denote by V_+, $V_\times$, and V_{MUX} the variances of $C_1 + C_2$, $C_1 \times C_2$, and $\mathsf{MUX}(C_1, C_2, C_3)$, respectively. Then there are constants $\alpha > 1$, $\beta > 0$ (depending on the HE scheme) such that:*

$$V_+ \le V_1 + V_2, \quad V_\times \le \alpha V_1 + V_2 + \beta, \quad V_{\mathsf{MUX}} \le \max(V_2, V_3) + \alpha V_1 + \beta.$$

We now show how to compute the value of $\langle v, x\rangle$ in a HE-friendly way in Algorithm 1. It is illustrated in Figure 3, and generalizes the standard Hamming-weight circuit of [29]. Correctness is proven in Proposition 2, while the noise variance of the resulting ciphertexts after homomorphic evaluation is bounded in Proposition 3.

Proposition 2. *For the circuit of Algorithm 1, we have $L_n = \mathrm{Im}(v)$. Moreover, for every $\ell \in \mathrm{Im}(v)$, the output value $M_{n,\ell}$ satisfies $M_{n,\ell} = 1$ if and only if $\langle v, x\rangle = \ell$.*

Proof. The proof follows the same inductive structure as in [29]. We prove by induction on $k \in [0, n]$ that $L_k = \left\{\sum_{i\in I} v_i : I \subseteq [1, k]\right\}$ and that, for every $\ell \in L_k$, we have $M_{k,\ell} = 1$ if and only if $\sum_{i=1}^{k} v_i x_i = \ell$. The case $k = 0$ follows from $L_0 = \{0\}$ and $M_{0,0} = 1$. Assume the claim holds at level $k - 1$ for $k \in [1, n]$.

[1] `https://eprint.iacr.org/2026/1392`

Algorithm 1: HE-friendly scalar-product circuit

Input: An integer vector $v \in \mathbb{Z}^n$ and bits $x_1, \ldots, x_n \in \mathbb{F}_2$ (the entries of $x \in \mathbb{F}_2^n$).
Output: Bits $(M_{n,\ell})_{\ell \in \mathrm{Im}(v)}$ with $M_{n,\ell} = 1$ if and only if $\langle v, x \rangle = \ell$.

$L_0 \leftarrow \{0\}$ and $M_{0,0} \leftarrow 1$
for $k \in [1, n]$ **do**
  $L_k \leftarrow L_{k-1} \cup (L_{k-1} + v_k)$
  for $\ell \in L_k$ **do**
    $M_{k,\ell} \leftarrow \mathsf{MUX}(x_k, M_{k-1,\ell}, M_{k-1,\ell-v_k})$ ▷ $M_{k-1,j}$ interpreted as 0 if undefined
  end for
end for
return $(M_{n,\ell})_{\ell \in L_n}$

Then $L_k = \left\{\sum_{i \in I} v_i : I \subseteq [1, k-1]\right\} \cup \left\{v_k + \sum_{i \in I} v_i : I \subseteq [1, k-1]\right\}$, so $L_k = \left\{\sum_{i \in I} v_i : I \subseteq [1, k]\right\}$. Next, let $\ell \in L_k$. If $x_k = 0$, then $M_{k,\ell} = M_{k-1,\ell}$, which is equal to 1 by the induction hypothesis exactly when $\sum_{i=1}^{k-1} v_i x_i = \ell$, equivalently when $\sum_{i=1}^{k} v_i x_i = \ell$; if $x_k = 1$, then $M_{k,\ell} = M_{k-1,\ell-v_k}$, which is equal to 1 exactly when $\sum_{i=1}^{k-1} v_i x_i = \ell - v_k$, again equivalently when $\sum_{i=1}^{k} v_i x_i = \ell$. □

Proposition 3. *Under Assumption 1, the circuit of Algorithm 1 can be evaluated homomorphically so that the output ciphertexts encrypting $M_{n,\ell}$ for $\ell \in \mathrm{Im}(v)$ have variance at most $n(\alpha V + \beta)$, where V is an upper bound on the variance of the input ciphertexts encrypting $x_1, \ldots, x_n \in \mathbb{F}_2$, and $\alpha > 1$, $\beta > 0$ are the constants from Assumption 1.*

Proof. We prove by induction on $k \in [0, n]$ that every ciphertext encrypting some $M_{k,\ell}$, with $\ell \in L_k$, has variance at most $k(\alpha V + \beta)$. For the case $k = 0$ the value of $M_{0,0} = 1$ is known and therefore the corresponding ciphertext has variance 0. Next, assume the claim holds at level $k-1$ for some $k \in [1, n]$. For every $\ell \in L_k$, we homomorphically compute $M_{k,\ell} = \mathsf{MUX}(x_k, M_{k-1,\ell}, M_{k-1,\ell-v_k})$, where undefined values are interpreted as constants equal to 0. By the induction hypothesis, the defined level-$(k-1)$ ciphertexts have variance at most $(k-1)(\alpha V + \beta)$, while the undefined values have variance 0. Hence, by Assumption 1 the ciphertext encrypting $M_{k,\ell}$ has variance at most $(k-1)(\alpha V + \beta) + \alpha V + \beta = k(\alpha V + \beta)$. □

We can now combine the scalar-product circuit with the local degree-d descriptions on the slices to obtain a general noise bound for the homomorphic evaluation of Boolean functions with small degree-d slice complexity.

Theorem 1. *Let $f \in \mathcal{B}_n$ be a Boolean function and $d \geq 0$ an integer. Assume we are given n ciphertexts encrypting the entries $x_1, \ldots, x_n$ of some $x \in \mathbb{F}_2^n$ with variance at most V. Then, under Assumption 1, one can homomorphically compute $f(x)$ with output variance bounded by:*

$$\kappa_d(f) \cdot \left((\alpha V + \beta) \left(\alpha n + d \sum_{i=1}^{d} \binom{n}{i} \right) + \beta \right),$$

where $\alpha > 1$ and $\beta > 0$ are the constants from Assumption 1.

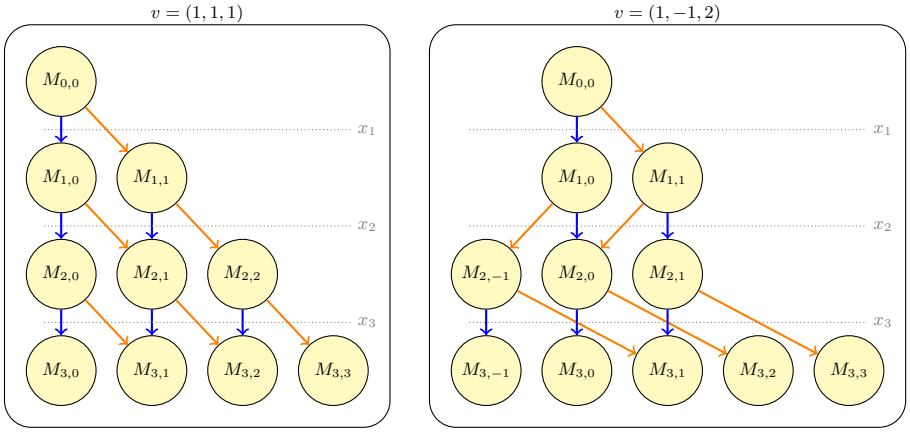

Fig. 3. Layered Boolean circuit computing the value of $\langle v, x\rangle$ for $v = (1, 1, 1)$ (left) and $v = (1, -1, 2)$ (right), adapted from [29]. Each node $M_{k,\ell}$ indicates whether the partial sum $\sum_{i=1}^{k} v_i x_i$ equals ℓ. Blue edges correspond to $x_i = 0$ (no change), while orange edges correspond to $x_i = 1$ (adding v_i). Recursively, we initialize $M_{0,0} \leftarrow 1$ and $M_{0,\ell} \leftarrow 0$ for $\ell \neq 0$, and compute $M_{k,\ell} \leftarrow \mathsf{MUX}(x_k, M_{k-1,\ell}, M_{k-1,\ell-v_k})$, where undefined entries are interpreted as 0.

Proof. We know that f is degree-d on v-slices for some $v \in \mathbb{Z}^n$ with $\kappa(v) = \kappa_d(f)$. Write $f = \sum_{k \in \mathrm{Im}(v)} \mathbb{1}_{\mathsf{E}_k^v} \cdot f_k$, where each $f_k \in \mathcal{B}_n$ has degree at most d. By Proposition 2 and Proposition 3, the homomorphic application of Algorithm 1 gives ciphertexts encrypting $M_{n,k} = \mathbb{1}_{\mathsf{E}_k^v}(x)$, each with variance at most $n(\alpha V+\beta)$.

An encryption of any monomial of degree at most d in the x_i can be computed with variance at most $d(\alpha V + \beta)$. Indeed, for a monomial $x_{i_1} \cdots x_{i_t}$ with $t \leq d$, we can apply Algorithm 1 to the vector $1_t \in \mathbb{Z}^t$ and input bits $x_{i_1}, \ldots, x_{i_t}$, yielding a family $(M_{t,\ell})_{\ell \in [0,t]}$, where $M_{t,t} = x_{i_1} \cdots x_{i_t}$. By Proposition 3, the homomorphic version of the circuit yields an encryption of $M_{t,t}$ with variance at most $d(\alpha V + \beta)$.

We now homomorphically compute $f_k(x)$ from its ANF. If present, its constant term is known and has variance 0, while its nonconstant part contains at most $B = \sum_{i=1}^{d} \binom{n}{i}$ monomials in the x_i. Each such monomial has variance at most $d(\alpha V + \beta)$; summing these monomial ciphertexts, together with the possible constant term, gives a ciphertext encrypting $f_k(x)$ with variance at most $dB(\alpha V + \beta)$ by Assumption 1. Left-multiplying the ciphertext encrypting $f_k(x)$ by the ciphertext encrypting $M_{n,k} = \mathbb{1}_{\mathsf{E}_k^v}(x)$ gives a ciphertext encrypting $\mathbb{1}_{\mathsf{E}_k^v}(x) \cdot f_k(x)$ with variance at most $(\alpha V + \beta)(\alpha n + dB) + \beta$. Finally, summing over $k \in \mathrm{Im}(v)$ gives a ciphertext encrypting $f(x)$ with variance at most the claimed value. □

Remark 5. Therefore, for fixed V, α, β, and d, the variance bound of Theorem 1 scales as $\mathcal{O}\big(\kappa_d(f) \cdot n^d\big)$. In other words, the dependence is polynomial in n and linear in the slice complexity $\kappa_d(f)$.

4 Combinatorics of Vector Slices

We study the map $\kappa : \mathbb{Z}^n \to \mathbb{N}$, which counts the number of slices induced by an integer vector. This quantity controls both slice complexity and the variance bound of Theorem 1.

4.1 Reduction to Positive Non-Decreasing Vectors

We first argue that, when computing $\kappa(v)$, we can always assume that $1 \le v_1 \le v_2 \le \cdots \le v_n$. Given an integer vector $v \in \mathbb{Z}^n$ and a Boolean vector $a \in \mathbb{F}_2^n$, we denote by $\mathsf{flip}_a(v)$ the vector w obtained from v by flipping the signs according to a, that is, $w_i = (-1)^{a_i} v_i$ for every $i \in [1, n]$.

Proposition 4. *For every integer vector $v \in \mathbb{Z}^n$ and every Boolean vector $a \in \mathbb{F}_2^n$, there exists an integer $c \in \mathbb{Z}$ such that $\langle \mathsf{flip}_a(v), x + a\rangle = \langle v, x\rangle + c$ for every $x \in \mathbb{F}_2^n$. In particular, $\mathrm{Im}(\mathsf{flip}_a(v)) = \mathrm{Im}(v) + c$ and $\kappa(\mathsf{flip}_a(v)) = \kappa(v)$.*

Proof. Let $I = \{i \in [1, n] : a_i = 1\}$ and $\overline{I} = [1, n] \setminus I$. Put $c = -\sum_{i \in I} v_i$. Then, for every $x \in \mathbb{F}_2^n$, we have:

$$\langle v, x\rangle + c = \sum_{i \in \overline{I}} v_i x_i + \sum_{i \in I} (-v_i)(1 - x_i) = \langle \mathsf{flip}_a(v), x + a\rangle. \qquad \square$$

For a permutation $\pi : [1, n] \to [1, n]$ and a vector $v \in \mathbb{Z}^n$, let us denote by $\pi(v)$ the vector $w \in \mathbb{Z}^n$ satisfying $w_i = v_{\pi(i)}$ for every $i \in [1, n]$.

Proposition 5. *Let $v \in \mathbb{Z}^n$ be an integer vector and $\pi : [1, n] \to [1, n]$ a permutation. Then for every $x \in \mathbb{F}_2^n$, we have $\langle \pi(v), \pi(x)\rangle = \langle v, x\rangle$. In particular, $\mathrm{Im}(\pi(v)) = \mathrm{Im}(v)$ and $\kappa(\pi(v)) = \kappa(v)$.*

Proof. We have $\langle \pi(v), \pi(x)\rangle = \sum_{i=1}^n v_{\pi(i)} x_{\pi(i)} = \sum_{i=1}^n v_i x_i = \langle v, x\rangle$. $\square$

Corollary 1. *Let $v \in \mathbb{Z}^n$ be an integer vector, and let $w \in \mathbb{N}^m$ with $1 \le w_1 \le w_2 \le \cdots \le w_m$ be the integer vector obtained from v by replacing its entries by their absolute values, deleting all zero entries, and then sorting the remaining entries in non-decreasing order. Then $\kappa(w) = \kappa(v)$.*

Proof. This follows from Proposition 4 and Proposition 5, together with the fact that deleting zero entries does not affect the image set. $\square$

Therefore, to compute $\kappa(v)$, we can always just apply the reduction from Corollary 1 and assume that $v \in \mathbb{Z}^n$ is a positive non-decreasing vector, that is, v satisfies $1 \le v_1 \le v_2 \le \cdots \le v_n$.

4.2 Attainable Number of Slices

We now analyze the image of the map $\kappa : \mathbb{Z}^n \to \mathbb{N}$ and its restriction $\kappa : \mathbb{Z}^n_{\neq 0} \to \mathbb{N}$.

Proposition 6. *For every integer vector $v \in \mathbb{Z}^n$, we have $\kappa(v) \in [1, 2^n]$. Conversely, for every $k \in [1, 2^n]$, there exists an integer vector $v \in \mathbb{Z}^n$ such that $\kappa(v) = k$.*

Proof. For $v \in \mathbb{Z}^n$, the map $\langle v, \cdot \rangle : \{0,1\}^n \to \mathbb{Z}$ has a domain size 2^n, and thus $\kappa(v) = |\mathrm{Im}(v)| \in [1, 2^n]$. For the converse, if $k = 1$, we take $v = 0_n$. Otherwise, if $k \in [2, 2^n)$, then the vector $v = \big(1, 2, \ldots, 2^{j-1}, k - 2^j, 0, \ldots, 0\big) \in \mathbb{N}^n$ with $j = \lfloor \log_2(k) \rfloor$ satisfies $\mathrm{Im}(v) = [0, k-1]$. If $k = 2^n$, we take $v = \big(1, 2, \ldots, 2^{n-1}\big) \in \mathbb{N}^n$, which satisfies $\mathrm{Im}(v) = [0, 2^n - 1]$. □

Proposition 7. *For every integer vector $v \in \mathbb{Z}^n_{\neq 0}$, we have $\kappa(v) \in [n+1, 2^n]$. Conversely, for every $k \in [n+1, 2^n]$, there exists a vector $v \in \mathbb{Z}^n_{\neq 0}$ such that $\kappa(v) = k$.*

Proof. A proof is provided in the full version of the paper.[2] □

In fact, the argument above admits a natural generalization, leading to the so-called *no-gap condition* stated in Proposition 8.

4.3 Bounds on the Number of Slices

In this subsection, we provide upper and lower bounds on $\kappa(v)$ in terms of the entries of v. We first focus on finding upper bounds, and provide a simple criterion under which the image set $\mathrm{Im}(v)$ is the full interval between 0 and the sum of the entries of v.

Proposition 8. *Let $v \in \mathbb{N}^n$ be an integer vector with $1 \le v_1 \le v_2 \le \cdots \le v_n$. Then $\mathrm{Im}(v) \subseteq [0, \sum_{i=1}^n v_i]$, with equality if and only if v satisfies the following* no-gap condition*: $v_i \le 1 + \sum_{j=1}^{i-1} v_j$ for every $i \in [1, n]$. Therefore, $\kappa(v)$ satisfies the following upper bound:*

$$\kappa(v) \le 1 + \sum_{i=1}^{n} v_i,$$

with equality if and only if v satisfies the no-gap condition.

Proof. For every $i \in [0, n]$, let $S_i = \sum_{j=1}^{i} v_j$. Then by definition, $\mathrm{Im}(v) \subseteq [0, S_n]$. We prove that $\mathrm{Im}(v) = [0, S_n]$ if and only if $v_i \le 1 + S_{i-1}$ for every $i \in [1, n]$. Assume first that $v_i > 1 + S_{i-1}$ for some $i \in [1, n]$. Then $1 + S_{i-1} \notin \mathrm{Im}(v)$ because $\langle v, x \rangle \le S_{i-1}$ for every $x \in \mathbb{F}_2^n$ with $x_j = 0$ for all $j \in [i, n]$, while $\langle v, x \rangle \ge v_i > 1 + S_{i-1}$ for every $x \in \mathbb{F}_2^n$ with $x_j = 1$ for some $j \in [i, n]$. Therefore, the value $1 + S_{i-1}$ lies in $[0, S_n]$ but not in $\mathrm{Im}(v)$. Conversely, assume that $v_i \le 1 + S_{i-1}$ for every $i \in [1, n]$. We prove by induction on $j \in [0, n]$ that

[2] `https://eprint.iacr.org/2026/1392`

the set $T_j = \left\{\sum_{i\in I} v_i : I \subseteq [1,j]\right\}$ equals $[0, S_j]$; choosing $j = n$ then yields $\mathrm{Im}(v) = T_n = [0, S_n]$. The base case $j = 0$ is direct: $T_0 = \{0\} = [0, S_0]$. For the induction step, assume that $T_{j-1} = [0, S_{j-1}]$ for some $j \in [1, n]$. Then $T_j = T_{j-1} \cup (T_{j-1} + v_j) = [0, S_{j-1}] \cup [v_j, S_{j-1} + v_j] = [0, S_j]$ because $v_j \leq 1 + S_{j-1}$, as required. □

In what follows, vectors $v \in \mathbb{Z}^n$ which satisfy the no-gap condition will be assumed to be of the form $1 \leq v_1 \leq v_2 \leq \cdots \leq v_n$.

Corollary 2. *Let $B \geq 0$. Let $v \in \mathbb{Z}^n$ be an integer vector satisfying $|v_i| \leq B$ for every $i \in [1, n]$. Then $\kappa(v)$ satisfies the following upper bound:*

$$\kappa(v) \leq 1 + nB.$$

Proof. Let $w \in \mathbb{N}^m$ be the vector obtained from v as described in Corollary 1. Then $\kappa(v) = \kappa(w) \leq 1 + \sum_{i=1}^m w_i \leq 1 + mB \leq 1 + nB$ by Proposition 8. □

A lower bound on $\kappa(v)$ can be obtained by counting the number of distinct values among the first i entries of v for every $i \in [1, n]$.

Proposition 9. *Let $v \in \mathbb{N}^n$ satisfy $1 \leq v_1 \leq v_2 \leq \cdots \leq v_n$. For every $i \in [1, n]$, let $d_i = |\{v_1, \ldots, v_i\}|$ be the number of distinct values among the first i entries of v. Then $\kappa(v)$ satisfies the following lower bound:*

$$\kappa(v) \geq 1 + \sum_{i=1}^{n} d_i.$$

Proof. For every $i \in [0, n]$, let $S_i = \sum_{j=1}^{i} v_j$. Fixing $i \in [1, n]$, we have $S_i - u \in \mathrm{Im}(v)$ for every $u \in \{v_1, \ldots, v_i\}$, but also $S_i - u \in [S_{i-1}, S_i - 1]$ because $u \leq v_i$. Therefore, for a fixed i, we obtain d_i distinct elements of $\mathrm{Im}(v)$ contained in the interval $[S_{i-1}, S_i - 1]$. These intervals are disjoint as i varies. Together with the additional value $S_n \in \mathrm{Im}(v)$, this gives $\kappa(v) \geq 1 + \sum_{i=1}^n d_i$. □

Corollary 3. *Let $v \in \mathbb{N}^n$ satisfy $1 \leq v_1 \leq v_2 \leq \cdots \leq v_n$. Assume that no value appears more than m times among the entries of v. Then $\kappa(v)$ satisfies the following lower bound:*

$$\kappa(v) \geq 1 + \frac{n(n+1)}{2m}.$$

Proof. We have $d_i \geq i/m$ in Proposition 9, so $\kappa(v) \geq 1 + \sum_{i=1}^n i/m = 1 + n(n+1)/(2m)$. □

4.4 Lowest Possible Numbers of Slices

We now characterize the vectors $v \in \mathbb{Z}^n_{\neq 0}$ that attain the smallest possible number of slices.

Proposition 10. *Let $v \in \mathbb{N}^n$ be an integer vector with $1 \leq v_1 \leq v_2 \leq \cdots \leq v_n$. Then $\kappa(v) \geq n + 1$, with equality if and only if $v = k \cdot 1_n$ for some $k \geq 1$.*

Proof. The inequality has already been established in Proposition 7. Next, if $v = k \cdot 1_n$, then $\mathrm{Im}(v) = \{0, k, 2k, \ldots, nk\}$, and hence $\kappa(v) = n+1$. Conversely, assume that $\kappa(v) = n+1$. For every $i \in [0, n]$, let $S_i = \sum_{j=1}^{i} v_j$. If $v_{i+1} > v_i$ for some $i \in [1, n-1]$, then $S_{i+1} - v_i \in (S_i, S_{i+1})$ would give another element of $\mathrm{Im}(v)$ not among $S_0 < \cdots < S_n$, contradicting $\kappa(v) = n+1$. Hence $v_1 = \cdots = v_n$, as required. □

This result admits a natural generalization to other small values of $\kappa(v)$.

Proposition 11. *Let $v \in \mathbb{N}^n$ satisfy $1 \le v_1 \le \cdots \le v_n$. Let $m \in [1, n/3+1]$ be an integer. Then $\kappa(v) = n+m$ if and only if $v = k \cdot (1_{n-r}, \lambda_1+1, \ldots, \lambda_r+1)$ for some $k \ge 1$, where $1 \le \lambda_1 \le \cdots \le \lambda_r$ partition $m-1$, that is, they satisfy $\lambda_1 + \cdots + \lambda_r = m-1$.*

Proof. Assume first that $v = k \cdot (1_{n-r}, \lambda_1+1, \ldots, \lambda_r+1)$ with k and the λ_i as above. To show that $\kappa(v) = n+m$, we can assume that $k = 1$ (since $\mathrm{Im}(kv) = k \cdot \mathrm{Im}(v)$). Notice that v satisfies the no-gap condition. Indeed, for an entry $\lambda_i + 1$, the sum of the previous entries is at least equal to the sum of the $n-r$ initial 1's, which we can bound as $n-r \ge n-(m-1) \ge 2(m-1) \ge \lambda_i$. It follows by Proposition 8 that $\kappa(v) = 1 + n + \sum_{i=1}^{r} \lambda_i = 1 + n + (m-1) = n+m$.

We now prove the converse. Assume that $v \in \mathbb{N}^n$ with $1 \le v_1 \le v_2 \le \cdots \le v_n$ satisfies $\kappa(v) = n+m$. For every $i \in [1, n]$, let $d_i = |\{v_1, \ldots, v_i\}|$. By Proposition 9, we have $n+m \ge 1 + \sum_{i=1}^{n} d_i$, that is, $m-1 \ge \sum_{i=1}^{n}(d_i - 1)$. Let $k = v_1$ and let r be the number of entries of v that are larger than k. Then $\sum_{i=1}^{n}(d_i - 1) \ge r$ and thus $m-1 \ge r$.

The first $n-r$ entries of v equal k; let us show that the remaining ones are divisible by k. If v_i was not divisible by k for some $i > n-r$, then $\{0, k, 2k, \ldots, (n-r)k\}$ and $\{v_i, v_i+k, v_i+2k, \ldots, v_i+(n-r)k\}$ would be disjoint subsets of $\mathrm{Im}(v)$, and thus $n+m = \kappa(v) \ge 2(n-r+1)$, which is a contradiction if we can show that $2(n-r+1) > n+m$. To this end, we have $n-r \ge (3m-3)-(m-1) = 2m-2 > m+r-2$. Therefore, $2(n-r+1) = (n-r+2)+(n-r) > (n-r+2)+(m+r-2) = n+m$, as claimed.

Thus, since all entries of v are divisible by k, we can assume that $k = 1$, and thus that $v = (1_{n-r}, \lambda_1+1, \ldots, \lambda_r+1)$ for some integers $1 \le \lambda_1 \le \cdots \le \lambda_r$. It remains to show that $\sum_{i=1}^{r} \lambda_i = m-1$. Let $w = (\lambda_1+1, \ldots, \lambda_r+1)$ and write $\mathrm{Im}(w) = \{\ell_0, \ldots, \ell_q\}$ with $\ell_0 < \ell_1 < \cdots < \ell_q$. Notice that $\mathrm{Im}(v) = \bigcup_{j=0}^{q} [\ell_j, \ell_j + n - r]$. This must be a set of consecutive integers, otherwise $\mathrm{Im}(v)$ would contain at least two disjoint integer intervals of length $n-r+1$, and thus $\kappa(v) \ge 2(n-r+1) > n+m$, a contradiction. It follows that $\mathrm{Im}(v) = [0, \ell_q+n-r] = [0, \sum_{i=1}^{r}(1+\lambda_i) + n - r]$, hence that $n+m = \kappa(v) = 1 + \sum_{i=1}^{r}(1+\lambda_i) + n - r$. The required equality $\sum_{i=1}^{r} \lambda_i = m-1$ follows. □

Example 4. Let $v \in \mathbb{N}^n$ be an integer vector with $1 \le v_1 \le v_2 \le \cdots \le v_n$. If $n \ge 3$, then by Proposition 11, we have $\kappa(v) = n+2$ if and only if $v = k \cdot (1_{n-1}, 2)$ for some $k \ge 1$. Similarly, if $n \ge 6$, then $\kappa(v) = n+3$ if and only if $v = k \cdot (1_{n-2}, 2, 2)$ or $v = k \cdot (1_{n-1}, 3)$ for some $k \ge 1$.

4.5 Generating Function for Slice Cardinalities

We note that $\kappa(v)$ also admits a generating-function interpretation. We will not develop this viewpoint further here, but it provides a compact way to recover the slice cardinalities and to derive some of their basic identities.

Proposition 12. *Let $v \in \mathbb{Z}^n$. Then the slice cardinalities of v are given by the following Laurent-polynomial identity:*

$$\sum_{k\in\mathbb{Z}} |\mathsf{E}_k^v| X^k = \prod_{i=1}^{n} (1 + X^{v_i}).$$

Proof. Expanding the product gives $\prod_{i=1}^{n}(1 + X^{v_i}) = \sum_{I\subseteq[1,n]} X^{\sum_{i\in I} v_i}$. Thus, for every $k \in \mathbb{Z}$, the coefficient of X^k counts the subsets $I \subseteq [1, n]$ such that $\sum_{i\in I} v_i = k$, which are, via the map $I \mapsto (\mathbb{1}_I(i))_{i\in[1,n]}$, in bijection with the Boolean vectors $x \in \mathbb{F}_2^n$ satisfying $\langle v, x\rangle = k$. □

Corollary 4. *Let $v = (w, v_n) \in \mathbb{Z}^n$ with $w \in \mathbb{Z}^{n-1}$. Let $S = \sum_{i=1}^{n} v_i$. Then, for every $k \in \mathbb{Z}$, we have:*

$$|\mathsf{E}_k^w| + |\mathsf{E}_{k-v_n}^w| = |\mathsf{E}_k^v| = |\mathsf{E}_{S-k}^v|.$$

Proof. Let $P(X) = \sum_{k\in\mathbb{Z}} |\mathsf{E}_k^v| X^k$. Comparing coefficients in the equality $P(X) = (1 + X^{v_n}) \prod_{i=1}^{n-1}(1 + X^{w_i})$ from Proposition 12 gives the first equality. Similarly, comparing coefficients in $P(X) = \prod_{i=1}^{n} X^{v_i}(1 + X^{-v_i}) = X^S P\big(X^{-1}\big)$ gives the second equality. □

5 Properties of Slice Complexities

We now collect general properties of the slice complexities $\kappa_d(f)$. We first record their monotonicity and invariance properties, then derive universal bounds, specialize to the constant-slice case $d = 0$, and finally show that generic Boolean functions have exponential slice complexity for degrees d below $n/2$.

5.1 Basic Properties

We begin with two elementary consequences of the definition: the case where the slice complexity is equal to 1, and the monotonicity of $\kappa_d(f)$ in the allowed degree.

Proposition 13. *Let $d \geq 0$ be an integer and let $f \in \mathcal{B}_n$ be a Boolean function. Then $\kappa_d(f) = 1$ if and only if $\mathsf{deg}(f) \leq d$.*

Proof. The only vector $v \in \mathbb{Z}^n$ with $\kappa(v) = 1$ is $v = 0_n$, whose unique slice is $\mathbb{F}_2^n$. Thus, $\kappa_d(f) = 1$ is equivalent to saying that f is of degree at most d on $\mathbb{F}_2^n$. □

Proposition 14. *Let $d \geq 0$ be an integer and let $f \in \mathcal{B}_n$ be a Boolean function. Then we have:*

$$2^n \geq \kappa_0(f) \geq \kappa_1(f) \geq \cdots \geq \kappa_n(f) = \kappa_{n+1}(f) = \cdots = 1.$$

Proof. The upper bound $\kappa_0(f) \leq 2^n$ follows by taking $v = \big(1, 2, 4, \ldots, 2^{n-1}\big) \in \mathbb{N}^n$, for which $\mathrm{Im}(v) = [0, 2^n - 1]$ and thus $\kappa(v) = 2^n$. Next, the sequence $(\kappa_d(f))_{d \geq 0}$ is non-increasing, since being degree-d on v-slices implies being degree-$(d+1)$ on v-slices. Finally, since every Boolean function has degree at most n, we have $\kappa_d(f) = 1$ by Proposition 13 for every $d \geq n$. □

Next, recall that the symmetries of the n-dimensional Boolean cube $\mathbb{F}_2^n$ are obtained by flipping and permuting coordinates. They are thus described by the maps of the form $x \mapsto \pi(x) + a$ for some $a \in \mathbb{F}_2^n$ and some permutation $\pi : [1, n] \to [1, n]$.

Definition 14 (Cube equivalence). *Two Boolean functions $f, g \in \mathcal{B}_n$ are called* cube-equivalent *if there exist a Boolean vector $a \in \mathbb{F}_2^n$ and a permutation $\pi : [1, n] \to [1, n]$ such that for every $x \in \mathbb{F}_2^n$, we have $f(x) = g(\pi(x) + a)$.*

Cube equivalence is an equivalence relation on $\mathcal{B}_n$. It refines the classical affine-equivalence relation, since cube-equivalent functions are necessarily affine-equivalent. We now show that degree-d slice complexities are invariant under cube equivalence.

Proposition 15. *Let $f, g \in \mathcal{B}_n$ be cube-equivalent Boolean functions, that is, there exist a Boolean vector $a \in \mathbb{F}_2^n$ and a permutation $\pi : [1, n] \to [1, n]$ such that for every $x \in \mathbb{F}_2^n$, we have $f(x) = g(\pi(x) + a)$. If f is degree-d on v-slices for some $v \in \mathbb{Z}^n$, then g is degree-d on w-slices for $w = \mathsf{flip}_a(\pi(v))$. In particular, $\kappa_d(f) = \kappa_d(g)$ for every integer $d \geq 0$.*

Proof. Assume that f is degree-d on v-slices, so that $f = \sum_{k \in \mathrm{Im}(v)} \mathbb{1}_{\mathsf{E}_k^v} \cdot f_k$ for some Boolean functions $f_k \in \mathcal{B}_n$ of degrees at most d. By Proposition 4 and Proposition 5, there exists an integer $c \in \mathbb{Z}$ such that for every $x \in \mathbb{F}_2^n$, we have $\langle w, \pi(x) + a \rangle = \langle \mathsf{flip}_a(\pi(v)), \pi(x) + a \rangle = \langle \pi(v), \pi(x) \rangle + c = \langle v, x \rangle + c$; in particular, $\mathrm{Im}(w) = \mathrm{Im}(v) + c$. For every $k \in \mathrm{Im}(w)$, define $g_k \in \mathcal{B}_n$ by $g_k(x) = f_{k-c}\big(\pi^{-1}(x + a)\big)$, so that $\mathsf{deg}(g_k) \leq d$. If $x \in \mathsf{E}_k^w$, then the above implies that $\pi^{-1}(x + a) \in \mathsf{E}_{k-c}^v$, and therefore $g(x) = f\big(\pi^{-1}(x + a)\big) = f_{k-c}\big(\pi^{-1}(x + a)\big) = g_k(x)$. Hence $g = \sum_{k \in \mathrm{Im}(w)} \mathbb{1}_{\mathsf{E}_k^w} \cdot g_k$, showing that g is degree-d on w-slices. Taking v such that $\kappa(v) = \kappa_d(f)$ gives $\kappa_d(g) \leq \kappa(w) = \kappa(v) = \kappa_d(f)$. Since cube equivalence is symmetric, it follows that $\kappa_d(f) = \kappa_d(g)$. □

5.2 Universal Bounds

We now establish universal bounds for $\kappa_d(f)$, which apply to arbitrary Boolean functions without imposing any additional structure.

Proposition 16. *Let $d \geq 0$ be an integer. Then for every Boolean function $f \in \mathcal{B}_n$, we have:*

$$\kappa_d(f) \geq \mathsf{deg}(f) - d + 1.$$

Proof. Let $v \in \mathbb{Z}^n$ be such that f is degree-d on v-slices and $\kappa(v) = \kappa_d(f)$. Then there exist Boolean functions $f_k \in \mathcal{B}_n$ ($k \in \mathrm{Im}(v)$) of degrees at most d such that $f = \sum_{k \in \mathrm{Im}(v)} \mathbb{1}_{\mathsf{E}_k^v} \cdot f_k$. For each $k \in \mathrm{Im}(v)$, the indicator function $x \mapsto \mathbb{1}_{\mathsf{E}_k^v}(x)$ can be written as a univariate polynomial (over $\mathbb{Q}$) in the variable $\langle v, x \rangle \in \mathbb{Z}$ of degree at most $\kappa_d(f) - 1$, and thus as a rational polynomial in the variables $x_1, \ldots, x_n$ of degree at most $\kappa_d(f) - 1$. Replacing in this expression x_i^r with x_i for all $i \in [1, n]$ and $r \geq 2$ gives us the NNF of $\mathbb{1}_{\mathsf{E}_k^v}$. Thus, the numerical degree, and hence also the algebraic degree, of $\mathbb{1}_{\mathsf{E}_k^v}$ is at most $\kappa_d(f) - 1$. The equality $f = \sum_{k \in \mathrm{Im}(v)} \mathbb{1}_{\mathsf{E}_k^v} \cdot f_k$ then implies $\mathsf{deg}(f) \leq \kappa_d(f) - 1 + d$. □

Remark 6. Therefore, by Proposition 16, if $f \in \mathcal{B}_n$ has degree d, then:

$$\kappa_d(f) = 1, \quad \kappa_{d-1}(f) \geq 2, \quad \kappa_{d-2}(f) \geq 3, \quad \ldots, \quad \kappa_0(f) \geq d + 1.$$

Proposition 17. *Let $d \geq 0$ be an integer and let $f \in \mathcal{B}_n$ be a Boolean function. Let $T \subseteq [1, n]$ be such that for every monomial $\prod_{i \in I} x_i$ appearing in the ANF of f, we have $|I \setminus T| \leq d$. Then:*

$$\kappa_d(f) \leq 2^{|T|}.$$

Proof. Write $T = \{t_1, \ldots, t_{|T|}\}$ and let $v \in \mathbb{Z}^n$ be given by $v_{t_i} = 2^{i-1}$ for $i \in [1, |T|]$ and $v_i = 0$ for $i \notin T$. We show that each monomial $\prod_{i \in I} x_i$ appearing in the ANF of f is degree-d on v-slices, which implies that f itself is degree-d on v-slices and thus that $\kappa_d(f) \leq \kappa(v) = 2^{|T|}$. Towards this, observe that on a fixed v-slice, the variables indexed by T are known by definition of v, and thus the monomial $\prod_{i \in I} x_i$ is either zero or equal to $\prod_{i \in I \setminus T} x_i$. By assumption, we have $|I \setminus T| \leq d$, so $\prod_{i \in I} x_i$ is degree-d on this v-slice, as required. □

Corollary 5. *For every integer $d \geq 0$ and every Boolean function $f \in \mathcal{B}_n$, we have:*

$$\kappa_d(f) \leq 2^{n-d}.$$

Proof. The condition of Proposition 17 is satisfied by $T = [d + 1, n]$. □

Corollary 6. *Let $f \in \mathcal{B}_n$ be a Boolean function of degree n. Then $\kappa_{n-1}(f) = 2$.*

Proof. By Proposition 16, we have $\kappa_{n-1}(f) \geq 2$. On the other hand, by Corollary 5, we have $\kappa_{n-1}(f) \leq 2^{n-(n-1)} = 2$. Hence $\kappa_{n-1}(f) = 2$. □

Next, we give another source of upper bounds, based on the geometry of the slices induced by a vector. For a fixed vector v, we measure the degree needed to represent arbitrary Boolean functions on each v-slice. Once this degree is known, the same vector v gives a degree-d slice decomposition for every Boolean function.

Definition 15 (Interpolation degree of a vector). *The* interpolation degree *of a vector $v \in \mathbb{Z}^n$, denoted $\delta(v)$, is the smallest integer $d \geq 0$ such that every Boolean function $f \in \mathcal{B}_n$ is degree-d on all v-slices.*

Example 5. Let $v = \left(1, 2, \ldots, 2^{n-1}\right) \in \mathbb{N}^n$ be an integer vector. Then E_k^v is a singleton for every $k \in \mathrm{Im}(v)$, and thus $\delta(v) = 0$.

Example 6. Consider the vector $v = (1_m, 0_{n-m}) \in \mathbb{N}^n$ for some $m \in [0, n]$. We claim that $\delta(v) \leq n - \lceil m/2 \rceil$ (actually, equality holds, but we will not need it here). To see this, notice that $\mathrm{Im}(v) = [0, m]$ and that for every $k \in [0, m]$, the k-th v-slice is $\mathsf{E}_k^v = \mathsf{E}_{k,m} \times \mathbb{F}_2^{n-m}$. We show that every $f : \mathsf{E}_{k,m} \times \mathbb{F}_2^{n-m} \to \mathbb{F}_2$ has degree at most $n - \lceil m/2 \rceil$ for every $k \in [0, m]$. Therefore, we fix such k and f. For every $(x, y) \in \mathsf{E}_{k,m} \times \mathbb{F}_2^{n-m}$, we can write $f(x, y) = \sum_a \sum_b f(a, b) \cdot \mathbb{1}_a(x)\mathbb{1}_b(y)$, where a ranges over $\mathsf{E}_{k,m}$ and b ranges over $\mathbb{F}_2^{n-m}$. The map $\mathbb{1}_a$ has degree at most $\min(k, m-k) \leq \lfloor m/2 \rfloor$ on $\mathsf{E}_{k,m}$ because $\mathbb{1}_a(x) = \prod_{i:a_i=1} x_i = \prod_{i:a_i=0}(1 + x_i)$ on $\mathsf{E}_{k,m}$, and the map $\mathbb{1}_b$ has degree at most $n - m$ on $\mathbb{F}_2^{n-m}$. Consequently, f has degree at most $\lfloor m/2 \rfloor + (n - m) = n - \lceil m/2 \rceil$.

Proposition 18. *Let $v \in \mathbb{Z}^n$ be an integer vector. Then, for every Boolean function $f \in \mathcal{B}_n$ and every integer $d \geq \delta(v)$, we have:*

$$\kappa_d(f) \leq \kappa(v).$$

Proof. By definition of $\delta(v)$, every Boolean function $f \in \mathcal{B}_n$ is degree-$\delta(v)$ on v-slices. Hence, if $d \geq \delta(v)$, then f is degree-d on v-slices. Therefore, $\kappa_d(f) \leq \kappa(v)$. □

Corollary 7. *For every Boolean function $f \in \mathcal{B}_n$ and every integer $d \in [\lfloor n/2 \rfloor, n-1]$, we have:*

$$\kappa_d(f) \leq 2(n - d).$$

Proof. Let $m = 2(n - d) - 1$. Consider the vector $v = (1_m, 0_{n-m}) \in \mathbb{N}^n$. We have $\delta(v) \leq n - \lceil m/2 \rceil = d$ by Example 6 and $\kappa(v) = m + 1 = 2(n - d)$. Hence, $\kappa_d(f) \leq \kappa(v) = 2(n - d)$ by Proposition 18. □

Thus, for $d \geq \lfloor n/2 \rfloor$, the degree-$d$ slice complexity of every Boolean function $f \in \mathcal{B}_n$ satisfies $\kappa_d(f) = \mathcal{O}(n)$. We will see in Section 5.4 that the situation is very different for $d < \lfloor n/2 \rfloor$, where generic Boolean functions have exponential slice complexity.

5.3 Constant Slice Complexity

In this subsection, we study the constant slice complexity $\kappa_0(f)$ in more detail. We consider only Boolean functions that depend on all variables.

Definition 16. *A Boolean function $f \in \mathcal{B}_n$ is said to* depend on all variables *if for every $i \in [1, n]$, there exists $x \in \mathbb{F}_2^n$ such that $f(x) \neq f(x + e_i)$.*

Remark 7. If $f \in \mathcal{B}_n$ does not depend on all variables, then there exists $i \in [1,n]$ such that $f(x) = f(x+e_i)$ for every $x \in \mathbb{F}_2^n$. Let $v \in \mathbb{Z}^n$ be such that $\kappa(v) = \kappa_0(f)$ and such that f is constant on v-slices. We can assume that $v_i = 0$ because f does not depend on the i-th variable. Letting $w \in \mathbb{Z}^{n-1}$ be the vector obtained from v by removing the i-th entry, we have $\kappa(w) = \kappa(v)$ and $\kappa(w) \le 2^{n-1}$ by Proposition 6. Thus, $\kappa_0(f) = \kappa(v) = \kappa(w) \le 2^{n-1}$.

Proposition 19. *Let $f \in \mathcal{B}_n$ depend on all variables. Then $\kappa_0(f) \ge n+1$, with equality if and only if f is cube-equivalent to a symmetric Boolean function.*

Proof. Let $v \in \mathbb{Z}^n$ be such that $\kappa(v) = \kappa_0(f)$ and such that f is constant on v-slices. Since f depends on all variables, no entry of v is zero. Let $w \in \mathbb{N}^n$ be given by Corollary 1, so that $1 \le w_1 \le \cdots \le w_n$ and $\kappa(w) = \kappa(v)$. Then Proposition 10 gives $\kappa_0(f) = \kappa(v) = \kappa(w) \ge n+1$.

Now assume that $\kappa_0(f) = n+1$. Then $\kappa(w) = n+1$, so $w = k \cdot 1_n$ for some $k \ge 1$ by Proposition 10. By construction, $w = \mathsf{flip}_a(\pi(v))$ for some $a \in \mathbb{F}_2^n$ and some $\pi : [1,n] \to [1,n]$. Hence, by Proposition 15, $g \in \mathcal{B}_n$ defined by $g(x) = f\big(\pi^{-1}(x+a)\big)$ is constant on w-slices, cube-equivalent to f, and symmetric since $w = k \cdot 1_n$.

Conversely, assume f is cube-equivalent to a symmetric Boolean function g. Since f depends on all variables, so does g. As g is constant on 1_n-slices, we have $\kappa_0(g) \le \kappa(1_n) = n+1$. The lower bound already proved gives $\kappa_0(g) \ge n+1$, and hence $\kappa_0(g) = n+1$. Finally, Proposition 15 gives $\kappa_0(f) = \kappa_0(g) = n+1$. □

Example 7. Consider the Dirac function $f = \mathbb{1}_{0_n} \in \mathcal{B}_n$. Then f is symmetric and depends on all variables, and thus $\kappa_0(f) = n+1$ by Proposition 19. More generally, consider the Dirac function $f = \mathbb{1}_a \in \mathcal{B}_n$ for some fixed $a \in \mathbb{F}_2^n$. Then $f(x) = \mathbb{1}_{0_n}(x+a)$ for every $x \in \mathbb{F}_2^n$, and thus f is cube-equivalent to $\mathbb{1}_{0_n}$, implying $\kappa_0(f) = \kappa_0(\mathbb{1}_{0_n}) = n+1$ by Proposition 15.

The same approach also yields a characterization of Boolean functions whose constant slice complexity is only slightly larger than the minimum value $n+1$.

Theorem 2. *Let $f \in \mathcal{B}_n$ depend on all variables, and assume that $\kappa_0(f) \ge n+m$ for some $m \in [1, n/3+1]$. Then $\kappa_0(f) = n+m$ if and only if f is cube-equivalent to a Boolean function $g \in \mathcal{B}_n$ of the form $g(x) = G(\hat{x})$ for some symmetric function $G \in \mathcal{B}_{n+m-1}$, where:*

$$\hat{x} = \big(x_1, \ldots, x_{n-r}, \underbrace{x_{n-r+1}, \ldots, x_{n-r+1}}_{\lambda_1+1}, \ldots, \underbrace{x_n, \ldots, x_n}_{\lambda_r+1}\big)$$

with $1 \le \lambda_1 \le \cdots \le \lambda_r$ partitioning $m-1$, that is, satisfying $\lambda_1 + \cdots + \lambda_r = m-1$.

Proof. Assume first that f is cube-equivalent to a Boolean function $g \in \mathcal{B}_n$ of the displayed form. Let $v = (1_{n-r}, \lambda_1+1, \ldots, \lambda_r+1) \in \mathbb{N}^n$. Since G is symmetric, the value of $g(x)$ only depends on the Hamming weight of $\hat{x}$, which is equal to $\langle v, x \rangle$. Thus, g is constant on v-slices, and so $\kappa_0(g) \le \kappa(v) = n+m$, where the equality follows from Proposition 11. Since f and g are cube-equivalent,

Proposition 15 gives $\kappa_0(f) = \kappa_0(g)$, so $\kappa_0(f) \leq n+m$. Together with the assumption $\kappa_0(f) \geq n+m$, this gives $\kappa_0(f) = n+m$.

Conversely, assume that $\kappa_0(f) = n+m$. Let $v \in \mathbb{Z}^n$ be such that $\kappa(v) = \kappa_0(f)$ and such that f is constant on v-slices. Since f depends on all variables, no entry of v is zero. Let $w \in \mathbb{N}^n$ be given by Corollary 1, so that $1 \leq w_1 \leq \cdots \leq w_n$ and $\kappa(w) = \kappa(v) = n+m$. Then, by Proposition 15, f is cube-equivalent to a Boolean function $g \in \mathcal{B}_n$ that is constant on w-slices. By Proposition 11, we have $w = k \cdot (1_{n-r}, \lambda_1+1, \ldots, \lambda_r+1)$ for some $k \geq 1$, where $1 \leq \lambda_1 \leq \cdots \leq \lambda_r$ satisfy $\lambda_1 + \cdots + \lambda_r = m-1$. Without loss of generality, we can assume that $k = 1$.

We now define the symmetric $G \in \mathcal{B}_{n+m-1}$. For every $\ell \in [0, n+m-1] = \mathrm{Im}(w)$, choose some $x \in \mathsf{E}_\ell^w$, and set $G(y) = g(x)$ for all $y \in \mathbb{F}_2^{n+m-1}$ with $\mathsf{w_H}(y) = \ell$. This does not depend on the choice of x because g is constant on w-slices. This fully defines G, and G is symmetric by construction. For every $x \in \mathbb{F}_2^n$, we have $\mathsf{w_H}(\hat{x}) = \langle w, x\rangle$, and thus $g(x) = G(\hat{x})$, as required. □

Example 8. Let $f \in \mathcal{B}_n$ depend on all variables. By Theorem 2, if $n \geq 3$ and $\kappa_0(f) \geq n+2$, then $\kappa_0(f) = n+2$ if and only if f is cube-equivalent to a Boolean function g of the form $g(x) = G(x_1, \ldots, x_{n-1}, x_n, x_n)$ for some symmetric function $G \in \mathcal{B}_{n+1}$. If $n \geq 6$ and $\kappa_0(f) \geq n+3$, then $\kappa_0(f) = n+3$ if and only if f is cube-equivalent to a Boolean function g of the form $g(x) = G(x_1, \ldots, x_{n-2}, x_{n-1}, x_{n-1}, x_n, x_n)$ or $g(x) = G(x_1, \ldots, x_{n-1}, x_n, x_n, x_n)$ for some symmetric function $G \in \mathcal{B}_{n+2}$.

We next turn from lower bounds to a general upper bound for $\kappa_0(f)$. Although some vectors induce 2^n distinct slices, this many slices are never necessary for constant slice decompositions: every Boolean function satisfies $\kappa_0(f) \leq 2^n - 1$, with a further improvement for self-dual functions.

Proposition 20. *Let $n \geq 2$. Then $\kappa_0(f) \leq 2^n - 1$ for every $f \in \mathcal{B}_n$. Further, if f is self-dual (that is, $f(x + 1_n) = f(x) + 1$ for every $x \in \mathbb{F}_2^n$), then $\kappa_0(f) \leq 2^n - 2$.*

Proof. Assume first that f is not self-dual. We may then assume that $f(0_{n-1}, 1) = f(1_{n-1}, 0)$ by cube equivalence. Consider $v = \big(1, 2, \ldots, 2^{n-2}, 2^{n-1} - 1\big)$. Then $\kappa(v) = 2^n - 1$ by Proposition 8, and the only two vectors identified by $x \mapsto \langle v, x\rangle$ are $(0_{n-1}, 1)$ and $(1_{n-1}, 0)$. Hence, f is constant on v-slices, so $\kappa_0(f) \leq 2^n - 1$.

It remains to prove the improved bound for self-dual functions. If $f(x, 0) = f(x, 1)$ for every $x \in \mathbb{F}_2^{n-1}$, then f does not depend on all variables, and hence $\kappa_0(f) \leq 2^{n-1} \leq 2^n - 2$, see Remark 7. Otherwise, choose $a \in \mathbb{F}_2^{n-1}$ such that $f(a, 1) = f(a, 0) + 1$, and define $b = (a_1, \ldots, a_{n-2}, a_{n-1} + 1, 0) \in \mathbb{F}_2^n$ and $g \in \mathcal{B}_n$ by $g(x) = f(x + b)$. By self-duality and the choice of a, we have $g(0_{n-2}, 1, y) = g(1_{n-2}, 0, y)$ for both $y \in \mathbb{F}_2$. Now let $v = \big(1, 2, \ldots, 2^{n-3}, 2^{n-2} - 1, 2^{n-1} - 1\big) \in \mathbb{N}^n$. Then $\kappa(v) = 2^n - 2$ by Proposition 8, and the only two pairs of Boolean vectors that are identified by $x \mapsto \langle v, x\rangle$ are $(0_{n-2}, 1, y)$ and $(1_{n-2}, 0, y)$ for both $y \in \mathbb{F}_2$. Hence, g is constant on v-slices, so $\kappa_0(g) \leq 2^n - 2$. Since f and g are cube-equivalent, Proposition 15 gives $\kappa_0(f) = \kappa_0(g) \leq 2^n - 2$. □

We end with a lower bound which relates constant slice complexity to partial symmetry. If every function cube-equivalent to f is symmetric in at most m

variables, then $\kappa_0(f)$ is bounded below by $\Omega(n^2/m)$. In particular, if m is bounded independently of n, then $\kappa_0(f)$ is at least quadratic in n.

Definition 17. *Let $m \in [1,n]$. A Boolean function $f \in \mathcal{B}_n$ is called* symmetric in m variables *if there exists a subset $I \subseteq [1,n]$ with $|I| = m$ such that for every permutation $\pi : [1,n] \to [1,n]$ satisfying $\pi(i) = i$ for every $i \in [1,n] \setminus I$, we have $f(x) = f(\pi(x))$ for every $x \in \mathbb{F}_2^n$.*

Proposition 21. *Let $f \in \mathcal{B}_n$ depend on all variables, and let $m \geq 1$. Assume that no Boolean function cube-equivalent to f is symmetric in more than m variables. Then:*

$$\kappa_0(f) \geq 1 + \frac{n(n+1)}{2m}.$$

Proof. Let $v \in \mathbb{Z}^n$ be such that $\kappa(v) = \kappa_0(f)$ and such that f is constant on v-slices. Since f depends on all variables, no entry of v is zero. By Corollary 1 and Proposition 15, we may replace v by a positive non-decreasing vector $w \in \mathbb{N}^n$ with $\kappa(w) = \kappa(v)$, up to replacing f by a cube-equivalent function $g \in \mathcal{B}_n$ which is constant on w-slices. The map g is not symmetric in more than m variables by assumption, and thus w cannot have more than m identical entries. Applying Corollary 3, we obtain $\kappa_0(f) = \kappa(v) = \kappa(w) \geq 1 + n(n+1)/(2m)$. □

5.4 Generic Exponential Slice Complexity

We know by Corollary 7 that once $d \geq \lfloor n/2 \rfloor$, every Boolean function $f \in \mathcal{B}_n$ satisfies $\kappa_d(f) = \mathcal{O}(n)$. The next result shows that this behavior is essentially sharp for generic Boolean functions: if d stays bounded away from $n/2$ on the left, then $\kappa_d(f)$ is almost surely exponential in n.

Theorem 3. *Let $(d_n)_{n\geq 1}$ be a sequence of integers with $d_n \in [0, \lfloor (1/2 - \varepsilon) \cdot n \rfloor]$ for some $\varepsilon > 0$. Then, as $n \to \infty$, almost all Boolean functions $f \in \mathcal{B}_n$ satisfy the following:*

$$\kappa_{d_n}(f) = 2^{\Omega(n)}.$$

Proof. A proof is provided in the full version of the paper.[3] □

We can summarize the behavior of $\kappa_d(f)$ as a function of $d \in [0,n]$ for a generic Boolean function $f \in \mathcal{B}_n$ as shown in Figure 4. For every fixed $\varepsilon > 0$ and every $d \leq (1/2 - \varepsilon)n$, almost all Boolean functions $f \in \mathcal{B}_n$ satisfy $\kappa_d(f) = 2^{\Omega(n)}$ by Theorem 3. On the other hand, for every $d \geq \lfloor n/2 \rfloor$, every Boolean function $f \in \mathcal{B}_n$ satisfies $\kappa_d(f) = \mathcal{O}(n)$ by Corollary 7. Thus, the transition from exponential to linear behavior occurs in the window between $(1/2 - \varepsilon)n$ and $\lfloor n/2 \rfloor$. Since $\varepsilon > 0$ can be chosen arbitrarily small, this window can be made arbitrarily narrow relative to n.

[3] `https://eprint.iacr.org/2026/1392`

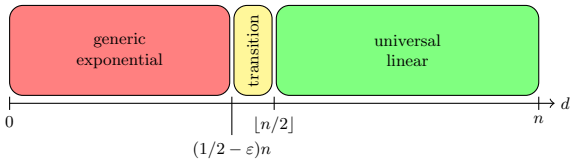

Fig. 4. Schematic behavior of $\kappa_d(f)$ as a function of $d \in [0, n]$. Below $(1/2 - \varepsilon)n$ (red), almost all Boolean functions have degree-d slice complexity which is exponential in n. From $\lfloor n/2 \rfloor$ onward (green), every Boolean function has degree-d slice complexity linear in n. A transition window (yellow) exists between these two regimes.

6 New Families of Boolean Functions

So far, slice decompositions have been used to analyze arbitrary Boolean functions by searching for a vector $v \in \mathbb{Z}^n$ such that f is degree-d on v-slices and $\kappa(v)$ is small. However, Theorem 3 shows that this generalization from Hamming-weight slices to v-slices cannot be expected to help for generic Boolean functions: for $d \leq (1/2 - \varepsilon)n$, almost all Boolean functions have exponential degree-d slice complexity, while for $d \geq \lfloor n/2 \rfloor$ the allowed local degree is already comparable to the degree of a general Boolean function, so the decomposition offers little advantage.

We therefore reverse the viewpoint and use vectors $v \in \mathbb{Z}^n$ with small $\kappa(v)$ to explicitly construct functions that are degree-d on v-slices. These functions inherit the efficient homomorphic evaluations from Section 3.4. We extend Boolean-function families originally defined on Hamming-weight slices to the vector slices listed in Table 1. These vectors are not meant to be canonical; they were chosen as simple examples with different growth behaviors. Here, $(F_i)_{i \geq 1}$ denotes the Fibonacci sequence with $F_1 = 1 = F_2$ and $F_{i+2} = F_i + F_{i+1}$. All listed vectors v satisfy the no-gap condition, so $\mathrm{Im}(v) = [0, \sum_{i=1}^n v_i]$ and $\kappa(v) = 1 + \sum_{i=1}^n v_i$ by Proposition 8.

Table 1. Vectors used for our constructions.

v	$\kappa(v)$
1_n	$n+1$
$(1_{n-1}, 2)$	$n+2$
$(1, 2, \ldots, n)$	$1 + n(n+1)/2$
$(F_1, F_2, \ldots, F_n)$	F_{n+2}

6.1 Variants of Classical Symmetric Boolean Functions

Recall that symmetric Boolean functions $f \in \mathcal{B}_n$ are those whose value $f(x)$ depends only on the Hamming weight $\mathsf{wH}(x)$, or equivalently, those that are constant on each slice $\mathsf{E}_{k,n}$. In our generalized setting, the analogous objects are Boolean functions whose value depends only on $\langle v, x \rangle$ for some vector $v \in \mathbb{Z}^n$,

that is, functions that are constant on each v-slice E_k^v. In what follows, we consider generalizations of the XOR function, the threshold functions, and the $\mathsf{0MOD3}$ function [21].

Definition 18 (XOR function). *The* n-variable XOR function *is the Boolean function* $\mathsf{XOR}_n : \mathbb{F}_2^n \to \mathbb{F}_2$ *defined by* $\mathsf{XOR}_n(x) = \sum_{i=1}^n x_i$.

An equivalent way to define it is as $\mathsf{XOR}_n(x) = \sum_{k=0}^n \mathbb{1}_{\mathsf{E}_{k,n}}(x) \cdot c_k$, where $c_k = (k \bmod 2) \in \mathbb{F}_2$. This point of view allows for a generalization to v-slices.

Definition 19 (XOR function on v-slices). *Let* $v \in \mathbb{Z}^n$. *The* n-variable XOR function on v-slices *is the Boolean function* $\mathsf{XOR}_n^v : \mathbb{F}_2^n \to \mathbb{F}_2$ *defined by* $\mathsf{XOR}_n^v(x) = \sum_{k \in \mathrm{Im}(v)} \mathbb{1}_{\mathsf{E}_k^v}(x) \cdot c_k$, *where* $c_k = (k \bmod 2) \in \mathbb{F}_2$.

By definition, XOR_n^v is constant on v-slices, and hence $\kappa_0(\mathsf{XOR}_n^v) \leq \kappa(v)$. In this case, however, the bound can be improved: the function XOR_n^v corresponds to a XOR function on the variables whose entries in v are odd.

Proposition 22. *Let* $v \in \mathbb{Z}^n$ *and* $m = |\{i \in [1,n] : v_i \equiv 1 \bmod 2\}|$. *Then* XOR_n^v *is cube-equivalent to the* n*-variable Boolean function defined by* $(x_1, \ldots, x_n) \mapsto \mathsf{XOR}_m(x_1, \ldots, x_m)$. *In particular,* XOR_n^v *depends on exactly* m *variables and* $\kappa_0(\mathsf{XOR}_n^v) = m + 1$.

Proof. By definition, we have $\mathsf{XOR}_n^v(x) = (\langle v, x \rangle \bmod 2)$ for every $x \in \mathbb{F}_2^n$. Let $\pi : [1,n] \to [1,n]$ be a permutation such that $\pi(\{1, \ldots, m\}) = \{i \in [1,n] : v_i \equiv 1 \bmod 2\}$. Consider the map $f : \mathbb{F}_2^n \to \mathbb{F}_2$ defined by $f(x) = \mathsf{XOR}_n^v(\pi(x))$. Then f is cube-equivalent to XOR_n^v, and for every $x \in \mathbb{F}_2^n$, we have $f(x) = (\langle \pi(x), v \rangle \bmod 2) = (\langle x, (1_m, 0_{n-m}) \rangle \bmod 2) = \mathsf{XOR}_m(x_1, \ldots, x_m)$. □

Table 2 confirms the behavior predicted by Proposition 22 through the resilience order of XOR_n^v.

Table 2. Resilience order of XOR_n^v for the vector families from Table 1.

n	4	5	6	7	8	9	10	11	12	13	14	15	16
1_n	3	4	5	6	7	8	9	10	11	12	13	14	15
$(1_{n-1}, 2)$	2	3	4	5	6	7	8	9	10	11	12	13	14
$(1, 2, \ldots, n)$	1	2	2	3	3	4	4	5	5	6	6	7	7
$(F_1, F_2, \ldots, F_n)$	2	3	3	4	5	5	6	7	7	8	9	9	10

We next generalize threshold functions.

Definition 20 (Threshold functions). *Let* $\theta \in \mathbb{Z}$. *The* θ-th n-variable threshold function *is the Boolean function* $\mathsf{T}_{\theta,n} : \mathbb{F}_2^n \to \mathbb{F}_2$ *defined by* $\mathsf{T}_{\theta,n}(x) = 1$ *if and only if* $\mathsf{w_H}(x) \geq \theta$.

Threshold functions form a classical and extensively studied family of Boolean functions. They include the majority function $\mathsf{MAJ}_n : \mathbb{F}_2^n \to \mathbb{F}_2$, which is defined by $\mathsf{MAJ}_n(x) = 1$ if and only if $\mathsf{w_H}(x) \geq n/2$, or equivalently $\mathsf{MAJ}_n = \mathsf{T}_{\lceil n/2 \rceil, n}$.

The majority function is one of the first known examples of a Boolean function with optimal AI; see, for instance, [19]. As for the XOR function, we can write $\mathsf{T}_{\theta,n}(x) = \sum_{k=0}^{n} \mathbb{1}_{\mathsf{E}_{k,n}}(x) \cdot c_k$, where $c_k = 1$ if $k \geq \theta$ and $c_k = 0$ otherwise. This slicewise description naturally extends to v-slices.

Definition 21 (Threshold functions on v-slices). *Let $v \in \mathbb{Z}^n$ and $\theta \in \mathbb{Z}$. The θ-*th *n-variable* threshold function on *v-slices is the Boolean function $\mathsf{T}^v_{\theta,n} : \mathbb{F}_2^n \to \mathbb{F}_2$ defined by $\mathsf{T}^v_{\theta,n}(x) = \sum_{k \in \mathrm{Im}(v)} \mathbb{1}_{\mathsf{E}^v_k}(x) \cdot c_k$, where $c_k = 1$ if $k \geq \theta$ and $c_k = 0$ otherwise.*

By definition, $\mathsf{T}^v_{\theta,n}$ is constant on v-slices, and hence $\kappa_0(\mathsf{T}^v_{\theta,n}) \leq \kappa(v)$. This generalization is known as a *linear threshold function* and appears in threshold logic, computational complexity, learning theory, and Fourier analysis of Boolean functions; see, for instance, [43, 45]. When $v \in \mathbb{N}^n$ satisfies the no-gap condition from Proposition 8, then $\mathrm{Im}(v) = [0, \kappa(v) - 1]$, and a natural analogue of the majority function is obtained by taking $\theta = \lfloor \kappa(v)/2 \rfloor$, yielding the generalized majority function MAJ^v_n. For $v = 1_n$, this recovers MAJ_n.

We now investigate the AI of the generalized threshold function. For simplicity, we restrict to the case where $v \in \mathbb{N}^n$. We further assume that $\theta \in [1, \sum_{i=1}^n v_i]$ to avoid cases where $\mathsf{T}^v_{\theta,n}$ is constant. We first require a result on the AI of ordinary threshold functions.

Property 2 (e.g., [9]). Let $\theta \in [0, n+1]$. Then the minimum degree of an annihilator of $\mathsf{T}_{\theta,n}$ has degree $n - \theta + 1$, while that of $\mathsf{T}_{\theta,n} + 1$ has degree θ. Therefore, $\mathsf{AI}(\mathsf{T}_{\theta,n}) = \min(n - \theta + 1, \theta)$.

Proposition 23. *Let $v \in \mathbb{N}^n$ and let $\theta \in [1, \sum_{i=1}^n v_i]$. Define r as the minimum Hamming weight of a Boolean vector x with $\langle v, x \rangle \geq \theta$, and define t as the maximum Hamming weight of a Boolean vector x with $\langle v, x \rangle < \theta$. Then:*

$$\min(r, n - t) \leq \mathsf{AI}(\mathsf{T}^v_{\theta,n}) \leq r.$$

Proof. Write $f = \mathsf{T}^v_{\theta,n}$. We first prove the upper bound. By definition of r, there exists $a \in \mathbb{F}_2^n$ with $\mathsf{w_H}(a) = r$ and $\langle v, a \rangle \geq \theta$. Let $I = \{i \in [1, n] : a_i = 1\}$ and $g = \prod_{i \in I} x_i \in \mathcal{B}_n$. For every $x \in \mathbb{F}_2^n$, if $g(x) = 1$, then $x_i = 1$ for every $i \in I$, and since $v \in \mathbb{N}^n$, we get $\langle v, x \rangle \geq \langle v, a \rangle \geq \theta$. Hence $f(x) = 1$ whenever $g(x) = 1$, so g is an annihilator of $f + 1$ of degree r. Therefore $\mathsf{AI}(f) \leq r$.

We now prove the lower bound. Let $0 \neq g \in \mathcal{B}_n$ have degree strictly smaller than r; then there exists $x \in \mathbb{F}_2^n$ with $\mathsf{w_H}(x) < r$ and $g(x) = 1$ (standard fact). By definition of r, this implies $\langle v, x \rangle < \theta$, and hence $f(x) = 0$. Thus, g cannot be an annihilator of $f + 1$. Similarly, let $0 \neq g \in \mathcal{B}_n$ have degree strictly smaller than $n - t$; then there exists $x \in \mathbb{F}_2^n$ with $\mathsf{w_H}(x) > t$ and $g(x) = 1$. By definition of t, this implies $\langle v, x \rangle \geq \theta$, and hence $f(x) = 1$. Thus, g cannot be an annihilator of f. Therefore every annihilator of $f + 1$ has degree at least r, and every annihilator of f has degree at least $n - t$, which gives $\mathsf{AI}(f) \geq \min(r, n - t)$. □

We further consider the generalization of the $\mathsf{0MOD3}$ function, which has been recently studied in [21] as an alternative to threshold functions in Goldreich's pseudorandom generator [28].

Definition 22 (0MOD3 function [21]). *The* n-variable 0MOD3 function *is the Boolean function* $\mathsf{0MOD3}_n : \mathbb{F}_2^n \to \mathbb{F}_2$ *defined by* $\mathsf{0MOD3}_n(x) = 1$ *if and only if* $\mathsf{w_H}(x) \equiv 0 \bmod 3$.

Writing $\mathsf{0MOD3}_n(x) = \sum_{k=0}^{n} \mathbb{1}_{\mathsf{E}_{k,n}}(x) \cdot c_k$ with $c_k = 1$ if $k \equiv 0 \bmod 3$ and $c_k = 0$ otherwise, we can extend the definition to v-slices.

Definition 23 (0MOD3 function on v-slices). *Let* $v \in \mathbb{Z}^n$. *The* n-variable 0MOD3 function on v-slices *is the Boolean function* $\mathsf{0MOD3}_n^v : \mathbb{F}_2^n \to \mathbb{F}_2$ *defined by* $\mathsf{0MOD3}_n^v(x) = \sum_{k \in \mathrm{Im}(v)} \mathbb{1}_{\mathsf{E}_k^v}(x) \cdot c_k$, *where* $c_k = 1$ *if* $k \equiv 0 \bmod 3$ *and* $c_k = 0$ *otherwise.*

We now report in Table 3 and Table 4 the degree, AI, and NL of the generalized majority and 0MOD3 functions for $n \in [4, 16]$ and for the vectors listed in Table 1. For the majority variants, the degree does not show a uniform trend across the considered vectors: some variants have higher degree than the classical case for most values of n, while others remain comparable or occasionally become smaller. In contrast, both the AI and the NL decrease compared to the classical case. For the 0MOD3 variants, the degree and AI decrease, while the NL remains comparable.

These small-dimensional experiments suggest that requiring the function to be constant on v-slices does not, by itself, preserve the optimal cryptographic parameters of the classical symmetric functions. This motivates the more expressive constructions considered next, where the function is only required to have low degree on each v-slice.

Table 3. Cryptographic parameters for MAJ_n^v for the vector families from Table 1.

Parameter	v	n												
		4	5	6	7	8	9	10	11	12	13	14	15	16
deg	1_n	4	4	4	4	8	8	8	8	8	8	8	8	16
	$(1_{n-1}, 2)$	3	4	4	7	7	8	8	8	8	8	8	15	15
	$(1, 2, \ldots, n)$	4	4	5	6	8	7	8	11	11	11	13	15	16
	$(F_1, F_2, \ldots, F_n)$	3	4	5	5	6	7	7	8	9	9	10	11	11
AI	1_n	2	3	3	4	4	5	5	6	6	7	7	8	8
	$(1_{n-1}, 2)$	2	2	3	3	4	4	5	5	6	6	7	7	8
	$(1, 2, \ldots, n)$	2	2	2	3	3	3	4	4	4	4	5	5	5
	$(F_1, F_2, \ldots, F_n)$	2	2	2	2	2	2	2	2	2	2	2	2	2
NL	1_n	5	10	22	44	93	186	386	772	1 586	3 172	6 476	12 952	26 333
	$(1_{n-1}, 2)$	2	6	12	29	58	130	260	562	1 124	2 380	4 760	9 949	19 898
	$(1, 2, \ldots, n)$	3	6	14	32	67	140	292	605	1 244	2 548	5 206	10 613	21 585
	$(F_1, F_2, \ldots, F_n)$	2	4	8	16	32	64	128	256	512	1 024	2 048	4 096	8 192

6.2 Generalizations of Weightwise Degree-d Functions

Next, we turn to examples inspired by classical weightwise degree-d constructions. We consider the hidden-weight-bit function (HWBF), introduced in [5] as a weightwise degree-1 function and later studied from a cryptographic viewpoint in [35, 55]. We also consider two weightwise degree-2 constructions: the revisited

Table 4. Cryptographic parameters for $\mathsf{0MOD3}_n^v$ for the vector families from Table 1.

Parameter	v	n												
		4	5	6	7	8	9	10	11	12	13	14	15	16
deg	1_n	4	5	5	7	8	8	10	11	11	13	14	14	16
	$(1_{n-1}, 2)$	4	4	6	7	7	9	10	10	12	13	13	15	16
	$(1, 2, \ldots, n)$	3	3	3	5	5	5	7	7	7	9	9	9	11
	$(F_1, F_2, \ldots, F_n)$	3	3	5	5	5	7	8	9	9	9	11	11	11
AI	1_n	2	2	2	3	3	3	4	4	4	5	5	5	6
	$(1_{n-1}, 2)$	2	2	2	3	3	3	4	4	4	5	5	5	6
	$(1, 2, \ldots, n)$	1	1	1	2	2	2	3	3	3	3	3	3	4
	$(F_1, F_2, \ldots, F_n)$	1	1	2	2	2	3	3	3	3	3	4	4	4
NL	1_n	5	7	14	37	85	170	341	683	1 366	2 731	5 461	10 922	21 845
	$(1_{n-1}, 2)$	5	10	21	37	74	171	341	682	1 365	2 731	5 462	10 923	21 845
	$(1, 2, \ldots, n)$	2	4	8	28	56	112	296	592	1 184	2 736	5 472	10 944	21 856
	$(F_1, F_2, \ldots, F_n)$	2	4	14	28	56	148	340	684	1 368	2 736	5 464	10 928	21 856

HWBF (R-HWBF) of [40] and the filtering function of the Nostalgia cipher [12]. As in the symmetric case, we replace Hamming-weight slices by v-slices and define the local functions slice by slice. Taking $v = 1_n$ recovers the original constructions. Throughout this subsection, indices of the entries of x are taken modulo n, with representatives in $[1, n]$. In particular, this means $x_0 = x_n$.

Definition 24 (Hidden-weight-bit function on v-slices). *Let $v \in \mathbb{Z}^n$. The* hidden-weight-bit function (HWBF) on v-slices *is the Boolean function* $\mathsf{HWBF}_n^v : \mathbb{F}_2^n \to \mathbb{F}_2$ *defined by* $\mathsf{HWBF}_n^v(x) = \sum_{k \in \mathrm{Im}(v)} \mathbb{1}_{\mathsf{E}_k^v}(x) \cdot f_k(x)$, *where* $f_k(x) = x_k$.

Definition 25 (Revisited hidden-weight-bit function on v-slices [40]). *Let $v \in \mathbb{Z}^n$. The* revisited hidden-weight-bit function (R-HWBF) on v-slices *is the Boolean function* $\mathsf{R\text{-}HWBF}_n^v : \mathbb{F}_2^n \to \mathbb{F}_2$ *defined by* $\mathsf{R\text{-}HWBF}_n^v(x) = \mathsf{HWBF}_n^v(x) + q(x)$, *where* $q(x) = \sum_{i=1}^{\lfloor n/2 \rfloor} (x_i + 1) x_{i+\lfloor n/2 \rfloor}$.

Definition 26 (Nostalgia filter on v-slices). *Let $v \in \mathbb{Z}^n$. The* Nostalgia filter on v-slices *is the Boolean function* $\mathsf{NF}_n^v : \mathbb{F}_2^n \to \mathbb{F}_2$ *defined by* $\mathsf{NF}_n^v(x) = \sum_{k \in \mathrm{Im}(v)} \mathbb{1}_{\mathsf{E}_k^v}(x) \cdot f_k(x)$, *where* $f_k(x) = x_k + \sum_{i=1}^{\lfloor n/3 \rfloor} x_{k-i} x_{k+2i-1}$.

We report the degree, AI, and NL of the above generalizations for $n \in [4, 16]$ and for the vectors listed in Table 1: the HWBF variants in Table 5, the R-HWBF variants in Table 6, and the Nostalgia filter variants in Table 7.

Across the considered vector families, the degree remains broadly comparable to the classical case, while the AI and NL increase moderately depending on the choice of the vector v. For the R-HWBF, the AI is slightly improved compared to the classical case, while the NL remains comparable. For the Nostalgia filter, some choices of v can slightly improve the already high NL, without significantly degrading the AI. Since the experiments are limited to $n \in [4, 16]$, even modest differences in the reported parameters may be meaningful and suggest that the passage from Hamming-weight slices to v-slices could lead to different trends asymptotically. We further recall that it was shown in [35] that the AI of weightwise degree-d functions is upper bounded by $\mathcal{O}(\sqrt{nd})$; extending this approach to the present setting appears non-trivial, as it relies on the specific algebraic structure

Table 5. Cryptographic parameters for HWBF_n^v for the vector families from Table 1.

Parameter	v	n												
		4	5	6	7	8	9	10	11	12	13	14	15	16
deg	1_n	3	4	5	6	7	8	9	10	11	12	13	14	15
	$(1_{n-1}, 2)$	4	5	6	7	8	9	10	11	12	13	14	15	16
	$(1, 2, \ldots, n)$	3	3	5	6	6	8	8	10	10	11	13	13	12
	$(F_1, F_2, \ldots, F_n)$	4	5	6	7	7	8	9	10	11	12	14	15	14
AI	1_n	2	2	3	3	4	4	4	5	5	5	5	6	6
	$(1_{n-1}, 2)$	2	2	3	3	4	4	4	5	5	5	6	6	6
	$(1, 2, \ldots, n)$	1	1	2	2	3	3	4	4	5	5	6	6	7
	$(F_1, F_2, \ldots, F_n)$	2	2	3	3	4	4	5	5	5	6	7	7	7
NL	1_n	4	10	20	44	88	186	372	772	1 544	3 172	6 344	12 952	25 904
	$(1_{n-1}, 2)$	5	11	23	47	97	195	399	799	1 627	3 255	6 607	13 215	26 761
	$(1, 2, \ldots, n)$	2	4	18	34	92	178	396	874	1 620	3 716	7 486	13 784	30 584
	$(F_1, F_2, \ldots, F_n)$	5	7	23	49	98	208	432	860	1 836	3 642	7 187	14 955	29 428

Table 6. Cryptographic parameters for $\mathsf{R\text{-}HWBF}_n^v$ for the vector families from Table 1.

Parameter	v	n												
		4	5	6	7	8	9	10	11	12	13	14	15	16
deg	1_n	3	4	5	6	7	8	9	10	11	12	13	14	15
	$(1_{n-1}, 2)$	4	5	6	7	8	9	10	11	12	13	14	15	16
	$(1, 2, \ldots, n)$	3	3	5	6	6	8	8	10	10	11	13	13	12
	$(F_1, F_2, \ldots, F_n)$	4	5	6	7	7	8	9	10	11	12	14	15	14
AI	1_n	2	3	3	3	4	4	5	5	6	6	6	7	7
	$(1_{n-1}, 2)$	2	2	3	3	4	4	5	5	6	6	7	7	7
	$(1, 2, \ldots, n)$	2	2	2	3	4	4	4	5	6	6	7	7	7
	$(F_1, F_2, \ldots, F_n)$	2	2	3	3	4	4	5	5	6	6	7	7	8
NL	1_n	4	10	22	46	102	218	458	934	1 902	3 894	7 842	15 804	31 680
	$(1_{n-1}, 2)$	3	9	19	47	99	221	459	937	1 909	3 901	7 785	15 893	31 901
	$(1, 2, \ldots, n)$	4	12	14	46	96	208	404	918	1 844	3 860	7 458	15 892	31 432
	$(F_1, F_2, \ldots, F_n)$	3	7	21	47	108	222	462	932	1 898	3 866	7 901	15 923	32 164

of classical Hamming-weight slices, which does not directly carry over to the partitions induced by a general vector v.

Overall, these experiments show that the structure of v can noticeably influence the cryptographic parameters of the resulting functions. In contrast with the XOR case, the generalization of weightwise degree-d functions does not collapse to a classical construction, but leads to non-trivial variations that can be useful for designing structured Boolean functions.

Table 7. Cryptographic parameters for NF_n^v for the vector families from Table 1.

Parameter	v	n												
		4	5	6	7	8	9	10	11	12	13	14	15	16
deg	1_n	3	4	5	6	6	8	9	10	11	12	13	14	15
	$(1_{n-1}, 2)$	4	5	6	6	7	9	10	10	12	13	13	15	16
	$(1, 2, \ldots, n)$	2	4	5	6	6	8	10	10	10	12	13	14	12
	$(F_1, F_2, \ldots, F_n)$	4	4	6	7	8	9	9	10	12	12	13	15	14
AI	1_n	2	2	3	3	4	5	5	5	6	6	7	7	8
	$(1_{n-1}, 2)$	1	2	3	3	4	4	5	5	6	6	7	7	8
	$(1, 2, \ldots, n)$	2	2	3	3	4	4	5	5	6	6	7	7	8
	$(F_1, F_2, \ldots, F_n)$	2	2	3	3	4	4	5	6	6	6	7	7	8
NL	1_n	4	10	22	48	104	218	456	922	1 892	3 874	7 782	15 740	31 810
	$(1_{n-1}, 2)$	3	11	23	46	104	217	453	940	1 927	3 893	7 836	15 849	31 843
	$(1, 2, \ldots, n)$	6	8	20	46	100	218	447	944	1 928	3 892	7 894	15 952	32 120
	$(F_1, F_2, \ldots, F_n)$	3	8	19	47	105	221	458	928	1 921	3 896	7 910	15 975	32 032

References

1. Applebaum, B., Lovett, S.: Algebraic Attacks against Random Local Functions and Their Countermeasures. In: Wichs, D., Mansour, Y. (eds.) Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016. pp. 1087–1100. ACM (2016). https://doi.org/10.1145/2897518.2897554
2. Baudrin, J., Belaïd, S., Bon, N., Boura, C., Canteaut, A., Leurent, G., Paillier, P., Perrin, L., Rivain, M., Rotella, Y., Tap, S.: Transistor: a TFHE-Friendly Stream Cipher. In: Kalai, Y.T., Kamara, S.F. (eds.) Advances in Cryptology - CRYPTO 2025 - 45th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2025, Proceedings, Part V. Lecture Notes in Computer Science, vol. 16004, pp. 530–565. Springer (2025). https://doi.org/10.1007/978-3-032-01901-1_17
3. Boura, C., Couteau, G., Perrin, L., Rotella, Y.: SoK: On Shallow Weak PRFs: A Common Symmetric Building Block for MPC Protocols. IACR Trans. Symmetric Cryptol. **2025**(3), 289–336 (2025). https://doi.org/10.46586/TOSC.V2025.I3.289-336
4. Braeken, A., Preneel, B.: On the Algebraic Immunity of Symmetric Boolean Functions. In: Progress in Cryptology - INDOCRYPT 2005, 6th International Conference on Cryptology in India, Bangalore, India, December 10-12, 2005, Proceedings. pp. 35–48 (2005)
5. Bryant, R.E.: On the Complexity of VLSI Implementations and Graph Representations of Boolean Functions with Application to Integer Multiplication. IEEE Trans. Computers pp. 205–213 (1991)
6. Canteaut, A., Videau, M.: Symmetric Boolean Functions. IEEE Trans. Information Theory pp. 2791–2811 (2005)
7. Carlet, C.: On the Degree, Nonlinearity, Algebraic Thickness, and Nonnormality of Boolean Functions, With Developments on Symmetric Functions. IEEE Trans. Information Theory pp. 2178–2185 (2004)
8. Carlet, C.: Boolean Functions for Cryptography and Coding Theory. Cambridge University Press (2021). https://doi.org/10.1017/9781108606806
9. Carlet, C., Méaux, P.: A Complete Study of Two Classes of Boolean Functions: Direct Sums of Monomials and Threshold Functions. IEEE Transactions on Information Theory pp. 1–1 (2021)
10. Carlet, C., Sarkar, P.: The Nonlinear Filter Model of Stream Cipher Redivivus. Cryptology ePrint Archive, Paper 2025/160 (2025), https://eprint.iacr.org/2025/160
11. Carlet, C., Sarkar, P.: Use of Simple Arithmetic Operations to Construct Efficiently Implementable Boolean Functions Possessing High Nonlinearity and Good Resistance to Algebraic Attacks. Discret. Appl. Math. **373**, 256–270 (2025). https://doi.org/10.1016/J.DAM.2025.05.004
12. Chacal, N., Guimarães, A., Martinelli, A., Méaux, P., Poussier, R.: Nostalgia Cipher: Can Filtered LFSRs Be Secure Again? An Application to Hybrid Homomorphic Encryption with Sub-50 ms Latency. IACR Transactions on Symmetric Cryptology **2025**(4), 1–30 (Dec 2025). https://doi.org/10.46586/tosc.v2025.i4.1-30, https://tosc.iacr.org/index.php/ToSC/article/view/12609
13. Chen, Y., Lu, P.: Two Classes of Symmetric Boolean Functions With Optimum Algebraic Immunity: Construction and Analysis. IEEE Transactions on Information Theory **57**(4), 2522–2538 (April 2011). https://doi.org/10.1109/TIT.2011.2111810

14. Chillotti, I., Gama, N., Georgieva, M., Izabachène, M.: Faster Fully Homomorphic Encryption: Bootstrapping in Less Than 0.1 Seconds. In: Cheon, J.H., Takagi, T. (eds.) Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I. Lecture Notes in Computer Science, vol. 10031, pp. 3–33 (2016). https://doi.org/10.1007/978-3-662-53887-6_1
15. Chillotti, I., Gama, N., Georgieva, M., Izabachène, M.: TFHE: Fast Fully Homomorphic Encryption Over the Torus. J. Cryptol. **33**(1), 34–91 (2020). https://doi.org/10.1007/S00145-019-09319-X
16. Cong, K., Das, D., Park, J., Pereira, H.V.: SortingHat: Efficient Private Decision Tree Evaluation via Homomorphic Encryption and Transciphering. In: Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security. p. 563–577. Association for Computing Machinery (2022). https://doi.org/10.1145/3548606.3560702
17. Courtois, N.T., Meier, W.: Algebraic Attacks on Stream Ciphers with Linear Feedback. In: Biham, E. (ed.) Advances in Cryptology - EUROCRYPT 2003, International Conference on the Theory and Applications of Cryptographic Techniques, Warsaw, Poland, May 4-8, 2003, Proceedings. Lecture Notes in Computer Science, vol. 2656, pp. 345–359. Springer (2003). https://doi.org/10.1007/3-540-39200-9_21
18. Cusick, T.W., Stanica, P.: Fast Evaluation, Weights and Nonlinearity of Rotation-Symmetric Functions. Discret. Math. **258**(1-3), 289–301 (2002). https://doi.org/10.1016/S0012-365X(02)00354-0
19. Dalai, D.K., Maitra, S., Sarkar, S.: Basic Theory in Construction of Boolean Functions with Maximum Possible Annihilator Immunity. Designs, Codes and Cryptography (2006)
20. Ducas, L., Micciancio, D.: FHEW: Bootstrapping Homomorphic Encryption in Less Than a Second. In: Oswald, E., Fischlin, M. (eds.) 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. Lecture Notes in Computer Science, vol. 9056, pp. 617–640. Springer (2015). https://doi.org/10.1007/978-3-662-46800-5_24
21. D'Antona, A.G., Méaux, P., Ünal, A.: On the Common Bias of Majorities: Poly-Time Attacks on THR-XOR PRGs. Cryptology ePrint Archive, Paper 2026/880 (2026), https://eprint.iacr.org/2026/880
22. Gao, G., Guo, Y., Zhao, Y.: Recent Results on Balanced Symmetric Boolean Functions. IEEE Transactions on Information Theory **62**(9), 5199–5203 (Sep 2016). https://doi.org/10.1109/TIT.2015.2455052
23. Gao, G., Cusick, T.W., Liu, W.: Families of Rotation Symmetric Functions with Useful Cryptographic Properties. IET Inf. Secur. **8**(6), 297–302 (2014). https://doi.org/10.1049/IET-IFS.2013.0241
24. Gentry, C.: Fully Homomorphic Encryption using Ideal Lattices. In: Mitzenmacher, M. (ed.) Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009. pp. 169–178. ACM (2009). https://doi.org/10.1145/1536414.1536440
25. Gentry, C., Sahai, A., Waters, B.: Homomorphic Encryption from Learning with Errors: Conceptually-Simpler, Asymptotically-Faster, Attribute-Based. In: Canetti, R., Garay, J.A. (eds.) Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I. Lecture Notes in Computer Science, vol. 8042, pp. 75–92. Springer (2013). https://doi.org/10.1007/978-3-642-40041-4_5

26. Gérard, F., Gini, A., Méaux, P.: TooLIP: How to Find New Instances of FiLIP Cipher with Smaller Key Size and New Filters. In: Vaudenay, S., Petit, C. (eds.) Progress in Cryptology - AFRICACRYPT 2024 - 15th International Conference on Cryptology in Africa, Douala, Cameroon, July 10-12, 2024, Proceedings. pp. 21–45. Lecture Notes in Computer Science, Springer (2024). https://doi.org/10.1007/978-3-031-64381-1_2
27. Gini, A., Méaux, P.: On the Weightwise Nonlinearity of Weightwise Perfectly Balanced Functions. Discrete Applied Mathematics **322**, 320–341 (2022). https://doi.org/10.1016/j.dam.2022.08.017
28. Goldreich, O.: Candidate One-Way Functions Based on Expander Graphs. Electronic Colloquium on Computational Complexity (ECCC) **7**(90) (2000)
29. Hoffmann, C., Méaux, P., Ricosset, T.: Transciphering, Using FiLIP and TFHE for an Efficient Delegation of Computation. In: Bhargavan, K., Oswald, E., Prabhakaran, M. (eds.) Progress in Cryptology - INDOCRYPT 2020 - 21st International Conference on Cryptology in India, Bangalore, India, December 13-16, 2020, Proceedings. Lecture Notes in Computer Science, vol. 12578, pp. 39–61. Springer (2020). https://doi.org/10.1007/978-3-030-65277-7_3
30. Kavut, S., Maitra, S., Tang, D.: Construction and Search of Balanced Boolean Functions on Even Number of Variables towards Excellent Autocorrelation Profile. Des. Codes Cryptogr. **87**(2-3), 261–276 (2019). https://doi.org/10.1007/S10623-018-0522-1
31. Liu, M., Lin, D., Pei, D.: Fast Algebraic Attacks and Decomposition of Symmetric Boolean Functions. IEEE Trans. Inf. Theory **57**(7), 4817–4821 (2011). https://doi.org/10.1109/TIT.2011.2145690
32. Maitra, S., Sarkar, P.: Maximum Nonlinearity of Symmetric Boolean Functions on Odd Number of Variables. Information Theory, IEEE Transactions on **48**, 2626 – 2630 (10 2002). https://doi.org/10.1109/TIT.2002.801482
33. Méaux, P.: On the Fast Algebraic Immunity of Majority Functions. In: Schwabe, P., Thériault, N. (eds.) Progress in Cryptology - LATINCRYPT. LNCS, vol. 11774, pp. 86–105. Springer (2019). https://doi.org/10.1007/978-3-030-30530-7_5
34. Méaux, P.: On the Fast Algebraic Immunity of Threshold Functions. Cryptography and Communications **13**(5), 741–762 (2021). https://doi.org/10.1007/s12095-021-00505-y
35. Méaux, P.: From at Least n/3 to at Most 3 Sqrt n: Correcting the Algebraic Immunity of the Hidden Weight Bit Function – Algebraic Immunity Upper Bounds on Weightwise Degree-d Functions and Their Implications. In: Escudero, D., Damgård, I. (eds.) Progress in Cryptology - LATINCRYPT 2025 - 9th International Conference on Cryptology and Information Security in Latin America, Medellín, Colombia, October 1-3, 2025, Proceedings. Lecture Notes in Computer Science, vol. 16129, pp. 345–369. Springer (2025). https://doi.org/10.1007/978-3-032-06754-8_13
36. Méaux, P., Carlet, C., Journault, A., Standaert, F.: Improved Filter Permutators for Efficient FHE: Better Instances and Implementations. In: Hao, F., Ruj, S., Gupta, S.S. (eds.) Progress in Cryptology - INDOCRYPT. LNCS, vol. 11898, pp. 68–91. Springer (2019)
37. Méaux, P., Journault, A., Standaert, F., Carlet, C.: Towards Stream Ciphers for Efficient FHE with Low-Noise Ciphertexts. In: Fischlin, M., Coron, J. (eds.) Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9665, pp. 311–343. Springer (2016). https://doi.org/10.1007/978-3-662-49890-3_13

38. Méaux, P., Ozaim, Y.: On the Cryptographic Properties of Weightwise Affine and Weightwise Quadratic Functions. Discret. Appl. Math. **355**, 13–29 (2024)
39. Méaux, P., Park, J., Pereira, H.V.L.: Towards Practical Transciphering for FHE with Setup Independent of the Plaintext Space. IACR Commun. Cryptol. **1**(1), 20 (2024). https://doi.org/10.62056/ANXRXRXQI
40. Méaux, P., Seuré, T., Tang, D.: The Revisited Hidden Weight Bit Function. In: Boura, C., Mashatan, A., Miri, A. (eds.) Selected Areas in Cryptography - SAC 2025 - 32nd International Conference, Toronto, ON, Canada, August 13-15, 2025, Revised Selected Papers. Lecture Notes in Computer Science, vol. 16207, pp. 235–263. Springer (2025). https://doi.org/10.1007/978-3-032-10536-3_9
41. Meier, W., Pasalic, E., Carlet, C.: Algebraic Attacks and Decomposition of Boolean Functions. In: Cachin, C., Camenisch, J.L. (eds.) Advances in Cryptology - EUROCRYPT 2004. pp. 474–491. Springer Berlin Heidelberg, Berlin, Heidelberg (2004)
42. Mesnager, S., Su, S., Li, J., Zhu, L.: Concrete Constructions of Weightwise Perfectly Balanced (2-Rotation Symmetric) Functions with Optimal Algebraic Immunity and High Weightwise Nonlinearity. Cryptogr. Commun. **14**(6), 1371–1389 (2022)
43. Muroga, S.: Threshold Logic and its Applications. Wiley (1971)
44. Naehrig, M., Lauter, K., Vaikuntanathan, V.: Can Homomorphic Encryption Be Practical? In: Proceedings of the 3rd ACM Workshop on Cloud Computing Security Workshop. p. 113–124. CCSW '11, Association for Computing Machinery, New York, NY, USA (2011). https://doi.org/10.1145/2046660.2046682, https://doi.org/10.1145/2046660.2046682
45. O'Donnell, R.: Analysis of Boolean Functions. Cambridge University Press (2014)
46. Qu, L., Feng, K., Liu, F., Wang, L.: Constructing Symmetric Boolean Functions With Maximum Algebraic Immunity. IEEE Transactions on Information Theory **55**, 2406–2412 (05 2009). https://doi.org/10.1109/TIT.2009.2015999
47. Qu, L., Li, C., Feng, K.: A Note on Symmetric Boolean Functions With Maximum Algebraic Immunity in Odd Number of Variables. IEEE Transactions on Information Theory **53** (2007)
48. Sarkar, P., Maitra, S.: Balancedness and Correlation Immunity of Symmetric Boolean Functions. Discrete Mathematics pp. 2351 – 2358 (2007)
49. Savický, P.: On the Bent Boolean Functions That are Symmetric. European Journal of Combinatorics **15**(4), 407 – 410 (1994), http://www.sciencedirect.com/science/article/pii/S0195669884710444
50. Siegenthaler, T.: Correlation-Immunity of Nonlinear Combining Functions for Cryptographic Applications. IEEE Trans. Inf. Theory **30**(5), 776–780 (1984). https://doi.org/10.1109/TIT.1984.1056949
51. Stanica, P., Maitra, S.: Rotation Symmetric Boolean Functions – Count and Cryptographic Properties. Electron. Notes Discret. Math. **15**, 139–145 (2003). https://doi.org/10.1016/S1571-0653(04)00560-8
52. Tang, D., Chattopadhyay, A., Roy, M., Mandal, B., Maitra, S.: Construction of Maiorana-McFarland Type Cryptographically Significant Boolean Functions with Good Implementation Properties. IACR Commun. Cryptol. **2**(2), 15 (2025). https://doi.org/10.62056/AKMPGYL7S
53. Tang, D., Kavut, S., Mandal, B., Maitra, S.: Modifying Maiorana-McFarland Type Bent Functions for Good Cryptographic Properties and Efficient Implementation. SIAM J. Discret. Math. **33**(1), 238–256 (2019). https://doi.org/10.1137/18M1202864

54. Tang, D., Maitra, S.: Construction of n-Variable (n $\equiv$ 2 mod 4) Balanced Boolean Functions With Maximum Absolute Value in Autocorrelation Spectra $< 2^{n/2}$. IEEE Trans. Inf. Theory **64**(1), 393–402 (2018). https://doi.org/10.1109/TIT.2017.2769092
55. Wang, Q., Carlet, C., Stanica, P., Tan, C.H.: Cryptographic Properties of the Hidden Weighted Bit Function. Discret. Appl. Math. **174**, 1–10 (2014)