Electronic Colloquium on Computational Complexity, Report No. 83 (2026)

Boolean Derivative Certificates and Maximal ANF Terms

Nicholas Smirnov

Department of Computer Science, Stony Brook University

Abstract

For a Boolean function $f : \{0,1\}^n \to \{0,1\}$, the higher-order Boolean derivative $D_S f$ computes the parity of f over each S-dimensional subcube. We prove that $D_S f \equiv 1$ exactly when S is a maximal monomial support in the algebraic normal form of f. This correspondence motivates the derivative certificate depth $\Delta_\partial(f)$, defined as the minimum degree of a nonempty maximal algebraic normal form monomial. We study the decision problem BCD: given f and k, decide whether $\Delta_\partial(f) \leq k$. We show that this problem is coNP-complete for every fixed constant $k \geq 1$, NP-complete when $k = n$, and belongs to $\Sigma_2^{\oplus P}$ when k is part of the input; in the variable-k case it is also both NP-hard and coNP-hard. These results identify the $\exists\forall\oplus$ upper bound and leave $\Sigma_2^{\oplus P}$-completeness as an open problem.

1 Introduction

The *algebraic normal form* (ANF) of a Boolean function f, its unique representation as an $\mathbb{F}_2$-multilinear polynomial, has been quite foundational to Boolean function theory since Zhegalkin [17]. The supports of the monomials in the ANF form a finite subset of $2^{[n]}$. Its inclusion-maximal elements form an antichain and encode the top-level algebra of f.

Boolean derivatives, systematized by Thayse [14], offer another perspective. The first-order derivative $D_i f$ measures sensitivity of f to flipping x_i, while $D_S f$ tests simultaneous sensitivity to all of S. Applications include test generation [11], formal verification [16], and circuit complexity [6].

The main equivalence connecting these two objects does not appear to be widely emphasized in the literature:

$$D_S f \equiv 1 \quad \iff \quad S \text{ is a maximal ANF monomial support of } f.$$

The condition $D_S f \equiv 1$ is a *parity certificate* for satisfiability, it guarantees that every S-dimensional subcube contains an odd number of satisfying assignments, hence at least one. We call S a *constant derivative certificate*. Whenever such a nonempty certificate exists, it gives stronger evidence for satisfiability than a single point witness: the subcube-parity condition holds universally over all $2^{n-|S|}$ base vectors b, whereas a point witness is a single evaluation. We define $\Delta_\partial(f)$ as the minimum cardinality of a constant derivative certificate.

We study the decision problem BCD: does f admit a constant derivative certificate of size $\leq k$? Its natural $\exists\forall\oplus$ formulation separates the analysis into three cases: fixed k, where the problem is coNP-complete; $k = n$, where it is NP-complete; and variable k, where it lies in $\Sigma_2^{\oplus \mathsf{P}}$.

Related work. Boolean derivatives appear in Boolean difference calculus, including Thayse's monograph [14], and in applications such as test generation [11], formal verification [16], and symmetric cryptography [2]. Polynomial representations of Boolean functions over finite fields are foundational in algebraic approaches to circuit lower bounds, including the Razborov–Smolensky method [10, 12]. Toda's theorem [15] shows the uses of counting and parity in complexity theory. The polynomial-time hierarchy originates with Stockmeyer [13]; parity-oracle variants are related to work of Papadimitriou and Zachos [8].

Organization. Section 2 recalls Boolean derivatives and ANF. Section 3 proves the derivative–ANF correspondence. Section 4 introduces derivative certificate depth. Sections 5–9 establish the complexity results.

ISSN 1433-8092

2 Preliminaries

Throughout, $[n] \coloneqq \{1, \ldots, n\}$ and $f\colon \{0,1\}^n \to \{0,1\}$. All arithmetic on Boolean values is over $\mathbb{F}_2$ unless otherwise stated.

Boolean derivatives

Definition 1. The *Boolean derivative* of f with respect to x_i is

$$D_i f \;\coloneqq\; f\big|_{x_i=0} \oplus f\big|_{x_i=1}.$$

For $S \subseteq [n]$, the *higher-order derivative* is $D_S f \coloneqq \big(\prod_{i\in S} D_i\big) f$, where the product denotes operator composition. We set $D_\varnothing f \coloneqq f$.

The operators D_i pairwise commute over $\mathbb{F}_2$: for any $i \neq j$ and $x \in \{0,1\}^n$, writing $x^{(T)}$ for x with bits in T flipped,

$$D_i(D_j f)(x) = f(x) \oplus f(x^{(i)}) \oplus f(x^{(j)}) \oplus f(x^{(ij)}) = D_j(D_i f)(x).$$

Since the operators D_i commute, $D_S f$ is independent of the order in which the variables of S are differentiated. Equivalently, if the variables outside S are fixed to $b \in \{0,1\}^{[n]\setminus S}$, then

$$D_S f(b) = \bigoplus_{u\in\{0,1\}^S} f(b,u). \tag{1}$$

Thus $D_S f(b)$ is the parity of f on the S-dimensional coordinate subcube obtained by varying the variables in S and fixing all others according to b.

Algebraic normal form

Every Boolean function has a unique *algebraic normal form*:

$$f(x) \;=\; \bigoplus_{T\subseteq[n]} c_T \prod_{i\in T} x_i, \qquad c_T \in \{0,1\},$$

with coefficients determined by Möbius inversion on the Boolean lattice: $c_T = \bigoplus_{R\subseteq T} f(\mathbf{1}_R)$, where $\mathbf{1}_R$ denotes the indicator vector of $R \subseteq [n]$ [6]. We write

$$\operatorname{supp}(f) \;\coloneqq\; \{T \subseteq [n] : c_T = 1\}$$

for the *monomial support* of f.

Definition 2. A set $T \in \operatorname{supp}(f)$ is a *maximal ANF monomial support* of f if no proper superset $T' \supsetneq T$ lies in $\operatorname{supp}(f)$.

3 The Derivative–ANF Correspondence

Proposition 3 (Derivative formula)**.** *For every f and every $S \subseteq [n]$,*

$$D_S f \;=\; \bigoplus_{\substack{T\subseteq[n]\\ T\supseteq S}} c_T \prod_{i\in T\setminus S} x_i. \tag{2}$$

Proof. By $\mathbb{F}_2$-linearity of D_S, it suffices to verify (2) on each monomial $m_T := \prod_{i \in T} x_i$.

Case $S \not\subseteq T$. Pick $j \in S \setminus T$. Since m_T does not depend on x_j, we have $D_j m_T = m_T \oplus m_T = 0$, and therefore $D_S m_T = 0$. This is consistent with (2), which contributes no term from T (as $T \not\supseteq S$).

Case $S \subseteq T$. For each $j \in S$:

$$D_j\left(\prod_{i \in T} x_i\right) = \prod_{i \in T} x_i\Big|_{x_j=0} \oplus \prod_{i \in T} x_i\Big|_{x_j=1} = 0 \oplus \prod_{i \in T \setminus \{j\}} x_i = \prod_{i \in T \setminus \{j\}} x_i.$$

Applying each D_j for $j \in S$ iteratively (in any order, by commutativity) gives $D_S m_T = \prod_{i \in T \setminus S} x_i$, which matches the T-term in (2). Summing over $T \supseteq S$ with $c_T = 1$ completes the proof. □

Theorem 4 (Derivative–ANF correspondence)**.** *$D_S f \equiv 1$ if and only if S is a maximal ANF monomial support of f.*

Proof. Decompose (2) by separating the $T = S$ term:

$$D_S f = c_S + \bigoplus_{\substack{T \supsetneq S \\ T \in \mathrm{supp}(f)}} \prod_{i \in T \setminus S} x_i.$$

The first term is constant, while every term with $T \supsetneq S$ is a nonconstant multilinear monomial in the remaining variables. By uniqueness of the ANF on the variables $[n] \setminus S$, this derivative is identically 1 if and only if

$$c_S = 1 \qquad \text{and} \qquad c_T = 0 \text{ for every } T \supsetneq S.$$

The first condition says that the monomial with support S appears in the ANF of f. The second says that no monomial whose support properly contains S appears in the ANF of f. Thus S is maximal among the ANF supports under set inclusion. □

Remark 5. Equation (2) shows that D_S acts on the ANF by *projecting onto monomials containing S, then cancelling the S-variables*. This is the Boolean analogue of partial differentiation in $\mathbb{F}_2[x_1, \ldots, x_n]$.

4 Constant Derivative Certificates

Definition 6. A nonempty set $S \subseteq [n]$ is a *constant derivative certificate* for f if $D_S f \equiv 1$. The *derivative certificate depth* of f is

$$\Delta_\partial(f) := \min\{|S| : \varnothing \neq S \subseteq [n],\ D_S f \equiv 1\},$$

with $\Delta_\partial(f) = \infty$ if no such S exists. By Theorem 4, $\Delta_\partial(f)$ equals the minimum degree of a nonempty maximal element of $\mathrm{supp}(f)$.

Proposition 7. *$\Delta_\partial(f) < \infty$ if and only if f is nonconstant.*

Proof. If $f \equiv 0$: then $\mathrm{supp}(f) = \varnothing$, so no maximal element exists and $\Delta_\partial(f) = \infty$.

If $f \equiv 1$: then $c_\varnothing = 1$ and $c_T = 0$ for all $T \neq \varnothing$ (since Möbius inversion gives $c_\varnothing = f(\mathbf{0}) = 1$ and $c_{\{i\}} = f(\mathbf{0}) \oplus f(\mathbf{e}_i) = 1 \oplus 1 = 0$, etc.). So $\mathrm{supp}(f) = \{\varnothing\}$, whose unique element is empty. Since Definition 6 requires a *nonempty* certificate, $\Delta_\partial(f) = \infty$.

If f is nonconstant: $\mathrm{supp}(f)$ contains at least one nonempty set. Since $\mathrm{supp}(f)$ is a finite nonempty poset, it has a maximal element S. If $S = \varnothing$, then every proper superset satisfies $c_T = 0$, and $c_\varnothing = 1$; but then $f \equiv 1$ by Möbius inversion, contradicting nonconstancy. So $S \neq \varnothing$, giving $\Delta_\partial(f) \leq |S| < \infty$. □

Remark 8. The proposition shows $\Delta_\partial(f) = \infty$ for *both* constant functions, even though $f \equiv 1$ is satisfiable. The parameter Δ_∂ does not measure satisfiability.

5 Examples

Example 9 (Linear functions)**.** If $f = x_{i_1} \oplus \cdots \oplus x_{i_k}$, the ANF consists of k degree-one monomials $\{i_1\}, \ldots, \{i_k\}$, each of which is maximal, since no higher-degree monomial appears. Hence $\Delta_\partial(f) = 1$.

Example 10 (Pure conjunction)**.** $f = x_1 \cdots x_k$ has a single ANF monomial $[k]$, which is trivially maximal. Hence $\Delta_\partial(f) = k$, showing that the parameter can equal the degree of the unique maximal ANF monomial.

Example 11 (Majority on three variables)**.** $f = x_1x_2 \oplus x_1x_3 \oplus x_2x_3$. Möbius inversion gives $c_{\{1,2,3\}} = \bigoplus_{R\subseteq\{1,2,3\}} f(\mathbf{1}_R) = 0$, so the degree-three monomial is absent. All three degree-two monomials are present and maximal. Hence $\Delta_\partial(f) = 2$.

Example 12 (Constant functions)**.** $\Delta_\partial(f) = \infty$ for $f \equiv 0$ and $f \equiv 1$ by Proposition 7.

6 The Decision Problem

Problem 13 (BOUNDED CONSTANT DERIVATIVE (BCD))**.** Given a Boolean circuit $f(x_1, \ldots, x_n)$ and an integer $1 \le k \le n$, decide whether $\Delta_\partial(f) \le k$.

By Theorem 4, BCD asks whether $\mathrm{supp}(f)$ has a maximal element of cardinality $\le k$. Expanding via (1):

$$\exists\, S \subseteq [n],\ 0 < |S| \le k \quad \forall\, b \in \{0,1\}^{[n]\setminus S} \quad \bigoplus_{u\in\{0,1\}^S} f(b,u) = 1. \tag{$*$}$$

The $\exists\forall\oplus$ quantifier structure of $(*)$ controls the complexity of BCD. The three layers work differently with k: at fixed k the parity layer ($\oplus$) collapses to polynomial time; at $k = n$ the $\forall$-layer ranges over the single empty tuple; for general k all three are utilized.

7 Fixed-Depth Complexity

Theorem 14. *For every fixed constant $K \ge 1$, the problem* BCD_K *is* coNP*-complete.*

Proof. coNP**-membership.** We show $\overline{\mathrm{BCD}_K} \in \mathsf{NP}$. An instance (f, K) is a no-instance iff every nonempty $S \subseteq [n]$ with $|S| \le K$ satisfies $D_S f \not\equiv 1$, i.e., there exists $b_S \in \{0,1\}^{[n]\setminus S}$ with $D_S f(b_S) = 0$. A nondeterministic machine guesses the family $\{b_S\}_S$ and verifies each condition

$$\bigoplus_{u\in\{0,1\}^S} f(b_S, u) = 0$$

by evaluating f at at most 2^K points. Since K is fixed and there are $\sum_{j=1}^{K}\binom{n}{j} = O(n^K)$ choices of S, verification is polynomial in n. Hence $\overline{\mathrm{BCD}_K} \in \mathsf{NP}$, so $\mathrm{BCD}_K \in$ coNP.

coNP**-hardness.** We give a polynomial-time many-one reduction from TAUTOLOGY to BCD_K. Given a formula $\psi(y_1, \ldots, y_m)$, introduce variables $x_1, \ldots, x_K$ (disjoint from $\mathbf{y}$) and define

$$f(x_1, \ldots, x_K, y_1, \ldots, y_m) \;\coloneqq\; x_1 \cdots x_K \cdot \psi(y_1, \ldots, y_m).$$

Let $X \coloneqq \{1, \ldots, K\}$ denote the indices of the x-variables.

Step 1: ANF of f. Since the x-variables and y-variables are disjoint, the product distributes as

$$f = \left(\prod_{i\in X} x_i\right) \cdot \left(\bigoplus_{T\subseteq[m]} c_T^{\psi} \prod_{j\in T} y_j\right) = \bigoplus_{T\subseteq[m]} c_T^{\psi} \left(\prod_{i\in X} x_i\right)\left(\prod_{j\in T} y_j\right),$$

where c_T^ψ denotes the ANF coefficient of ψ for the monomial $\prod_{j\in T} y_j$. For any $T \subseteq [m]$, let $T' := \{j + K : j \in T\}$ be the corresponding indices in f. Hence

$$\operatorname{supp}(f) = \{X \cup T' : T \subseteq [m],\ c_T^\psi = 1\}.$$

Thus every ANF monomial of f contains X as a subset and has degree $\geq K$.

Step 2: Any certificate of size $\leq K$ must equal X. Suppose $S \subseteq [n]$ with $|S| \leq K$ is a constant derivative certificate, so $D_S f \equiv 1$. We first show $X \subseteq S$. Suppose for contradiction that $X \not\subseteq S$, and pick any $j \in X \setminus S$. By Proposition 3,

$$D_S f = \bigoplus_{\substack{M \in \operatorname{supp}(f) \\ M \supseteq S}} \prod_{i \in M \setminus S} x_i.$$

Since every $M \in \operatorname{supp}(f)$ contains all of X, we have $j \in M \setminus S$ for every contributing term. Thus the variable x_j divides every summand, giving $D_S f = x_j \cdot g$ for some Boolean polynomial g. Setting $x_j = 0$ then yields $D_S f\big|_{x_j=0} \equiv 0 \neq 1$, contradicting $D_S f \equiv 1$. Hence $X \subseteq S$. Combined with $|S| \leq K = |X|$, we get $S = X$.

Step 3: $\Delta_\partial(f) \leq K$ iff $\psi \equiv 1$. By Step 2, $\Delta_\partial(f) \leq K$ iff $D_X f \equiv 1$. Applying Proposition 3 with $S = X$:

$$D_X f = \bigoplus_{T \subseteq [m]} c_T^\psi \prod_{j \in T} y_j = \psi.$$

Hence $D_X f \equiv 1$ iff $\psi \equiv 1$. The reduction is polynomial-time and TAUTOLOGY is coNP-complete [3]. □

8 The Unrestricted Case: $k = n$

Theorem 15. BCD *with $k = n$ is* NP*-complete.*

Proof. When $k = n$, the condition $\Delta_\partial(f) \leq n$ is equivalent, by Proposition 7, to f being nonconstant (since every nonempty monomial has degree $\leq n$).

NP**-membership.** A pair of assignments $a, b \in \{0,1\}^n$ with $f(a) \neq f(b)$ is a polynomial-time verifiable witness for nonconstancy.

NP**-hardness.** We reduce SAT to nonconstancy. Given $\varphi(y_1, \ldots, y_m)$, introduce a variable z and define $f(y_1, \ldots, y_m, z) := z \cdot \varphi(y_1, \ldots, y_m)$.

If φ is unsatisfiable, then $\varphi \equiv 0$ as a Boolean function, so $f \equiv 0$ is constant. If φ is satisfiable with witness $a \in \{0,1\}^m$, then $f(a, 0) = 0 \neq 1 = f(a, 1)$, so f is nonconstant. The reduction runs in polynomial time. □

Remark 16. The transition from coNP to NP as k grows is quite interesting. In $(*)$, the $\forall$-layer ranges over 2^{n-k} base vectors: at fixed k this is exponential, making the $\forall$-layer the bottleneck; at $k = n$ it ranges over $\{0,1\}^0 = \{()\}$, a single empty tuple, making it vacuous. As k increases, the $\forall$-layer collapses and the complexity seems to go from coNP toward NP.

9 The Variable-k Case and $\Sigma_2^{\oplus\mathsf{P}}$

We define $\Sigma_2^{\oplus\mathsf{P}}$ as the class of decision problems of the form

$$x \in L \iff \exists\, S \in \{0,1\}^{p(|x|)}\ \forall\, b \in \{0,1\}^{q(|x|)} \bigoplus_{u \in \{0,1\}^{r(|x|)}} C(x, S, b, u) = 1$$

for some polynomial-time computable Boolean circuit C and polynomials p, q, r. This is the parity-oracle analogue of the second existential–universal level; equivalently, one may view it as $\mathsf{NP}^{\Pi_1^{\oplus\mathsf{P}}}$, where $\Pi_1^{\oplus\mathsf{P}} = \mathsf{coNP}^{\oplus\mathsf{P}}$, as considered by Papadimitriou and Zachos [8].

Proposition 17. *The general problem* BCD *(variable k) lies in* $\Sigma_2^{\oplus\mathsf{P}}$ *and is both* NP*-hard and* coNP*-hard under polynomial-time many-one reductions.*

Proof. **Hardness.** Immediate from Theorems 14 and 15: the fixed-K and $k = n$ specializations of BCD are coNP-hard and NP-hard, respectively.

$\Sigma_2^{\oplus\mathsf{P}}$**-membership.** The quantifier structure $(*)$ directly witnesses membership. Equivalently, a $\mathsf{NP}^{\Pi_1^{\oplus\mathsf{P}}}$ algorithm proceeds as follows. First, nondeterministically guess $S \subseteq [n]$ and check in polynomial time that $0 < |S| \leq k$. For this fixed S, the required verification condition is

$$\forall b \in \{0,1\}^{[n]\setminus S} \quad \bigoplus_{u\in\{0,1\}^S} f(b,u) = 1.$$

For fixed S and b, the inner parity predicate is decidable by an $\oplus\mathsf{P}$ oracle: it asks whether the number of $u \in \{0,1\}^S$ satisfying $f(b,u) = 1$ is odd. Hence the universal verification condition lies in $\mathsf{coNP}^{\oplus\mathsf{P}} = \Pi_1^{\oplus\mathsf{P}}$. The outer nondeterministic machine accepts exactly when this $\Pi_1^{\oplus\mathsf{P}}$-oracle verifies the condition.

Formally, encode S by its length-n characteristic vector. Let b and u range over $\{0,1\}^n$, with the coordinates of b inside S ignored. Define

$$x_i(S,b,u) = \begin{cases} u_i, & i \in S, \\ b_i, & i \notin S. \end{cases}$$

Now define the polynomial-time predicate

$$C(f,S,b,u) = \begin{cases} f(x(S,b,u)), & \text{if } u_i = 0 \text{ for every } i \notin S, \\ 0, & \text{otherwise.} \end{cases}$$

Then

$$\bigoplus_{u\in\{0,1\}^n} C(f,S,b,u) = \bigoplus_{u\in\{0,1\}^S} f(b,u).$$

Thus the variable-length parity over $\{0,1\}^S$ can be represented as a fixed n-bit parity predicate, placing the verification condition in $\Pi_1^{\oplus\mathsf{P}} = \mathsf{coNP}^{\oplus\mathsf{P}}$. Since the outer machine nondeterministically guesses S and then invokes this $\Pi_1^{\oplus\mathsf{P}}$ verification condition, we obtain $\mathrm{BCD} \in \mathsf{NP}^{\Pi_1^{\oplus\mathsf{P}}} = \Sigma_2^{\oplus\mathsf{P}}$. □

At fixed $k = K$, the inner parity $\bigoplus_{u\in\{0,1\}^K} f(b,u)$ is computable in polynomial time because K is constant. Thus the parity-oracle layer disappears, consistent with the coNP-completeness result above. At $k = n$, the $\forall$-layer is vacuous and the problem reduces to pure NP, consistent with Theorem 15.

The class $\Sigma_2^{\oplus\mathsf{P}}$ contains the ordinary Σ_2^P as a special case: take the inner parity range to have size one and let the circuit compute an ordinary polynomial-time predicate. It is also contained in PSPACE, since a polynomial-space machine can enumerate the existential witness, universally check all b, and compute each inner parity by enumeration. Thus we use only the conservative containments

$$\mathsf{NP},\ \mathsf{coNP} \ \subseteq\ \Sigma_2^P \ \subseteq\ \Sigma_2^{\oplus\mathsf{P}} \ \subseteq\ \mathsf{PSPACE}.$$

We do not rely on any strict separations among these classes.

We conjecture that BCD is $\Sigma_2^{\oplus\mathsf{P}}$-complete. A proof would reduce a $\exists S\,\forall b\,[\bigoplus_u C(S,b,u) = 1]$-complete problem to BCD by constructing a circuit f for which $\bigoplus_u f(b,u)$ simulates $\bigoplus_u C(S,b,u)$ under the identification $S \leftrightarrow$ existential witness, $b \leftrightarrow$ universal input; we leave this open.

The complexity of BCD as k varies is summarized in Table 1.

k	Established complexity	Active quantifier layers
Fixed constant K	coNP-complete	$\forall$-layer dominates; the $\oplus$-layer is polynomial-time
Variable k	In $\Sigma_2^{\oplus\mathsf{P}}$; NP-hard and coNP-hard	$\exists$, $\forall$, and $\oplus$ layers can all be active; completeness is left open
$k = n$	NP-complete	$\forall$-layer is vacuous; the problem reduces to nonconstancy

Table 1: Established complexity of BCD as k varies.

The Derivative Lattice

The family of all higher-order derivatives of f can be organized as the *derivative lattice*: nodes are the derivatives $D_S f$ for $S \subseteq [n]$, ordered by inclusion of S, with an edge from $D_S f$ to $D_{S\cup\{i\}} f$ whenever $i \notin S$. Level k contains exactly $\binom{n}{k}$ nodes, the derivatives of order k. This is actually the Hasse diagram of the Boolean lattice $2^{[n]}$, shown by the fact that the operators D_i commute: the node reached by applying D_i then D_j is the same as the node reached by applying D_j then D_i.

In this picture, Theorem 4 says that a node $D_S f$ is *accepting* (identically 1) if and only if S is a maximal ANF monomial support of f. The derivative certificate depth $\Delta_\partial(f)$ is then the *minimum level* containing an accepting node. Figure 1 illustrates this for $n = 3$.

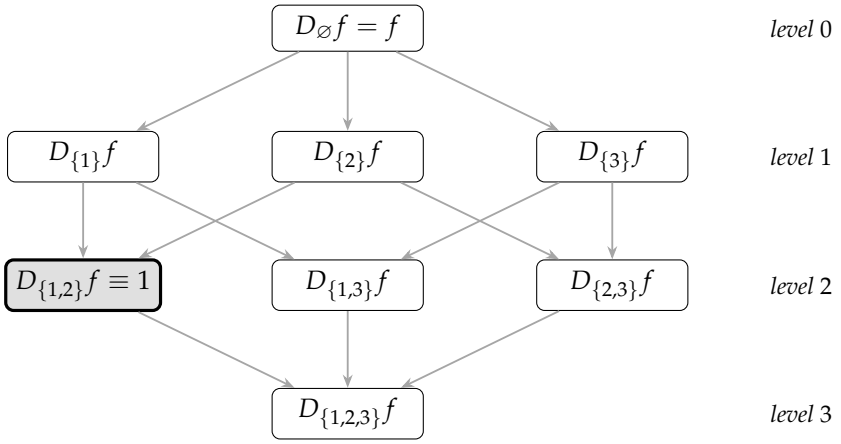

Figure 1: The derivative lattice for $n = 3$. Each node $D_S f$ sits at level $|S|$; edges go from $D_S f$ to $D_{S\cup\{i\}} f$ for $i \notin S$. A node is *accepting* (shaded, bold border) when $D_S f \equiv 1$, equivalently when S is a maximal ANF monomial support of f. In the example shown, an accepting node first appears at level 2, so $\Delta_\partial(f) = 2$.

References

[1] G. Boole. *An Investigation of the Laws of Thought*. Macmillan, London, 1854.

[2] C. Carlet. Boolean functions for cryptography and error correcting codes. In Y. Crama and P. L. Hammer, editors, *Boolean Models and Methods in Mathematics, Computer Science, and Engineering*, pages 257–397. Cambridge University Press, 2010.

[3] S. A. Cook. The complexity of theorem-proving procedures. In *Proc. 3rd Annual ACM STOC*, pages 151–158, 1971.

[4] Y. Crama and P. L. Hammer. *Boolean Functions: Theory, Algorithms, and Applications*. Cambridge University Press, 2011.

[5] M. R. Garey and D. S. Johnson. *Computers and Intractability*. W. H. Freeman, 1979.

[6] S. Jukna. *Boolean Function Complexity: Advances and Frontiers*. Springer, 2012.

[7] N. Nisan and M. Szegedy. On the degree of Boolean functions as real polynomials. *Computational Complexity*, 4(4):301–313, 1994.

[8] C. H. Papadimitriou and S. K. Zachos. Two remarks on the power of counting. In *Proc. 6th GI Conf. on Theoretical Computer Science*, LNCS, pages 269–275. Springer, 1983.

[9] C. H. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.

[10] A. A. Razborov. Lower bounds on the size of bounded depth circuits over a complete basis with logical addition. *Mathematical Notes*, 41(4):333–338, 1987.

[11] F. F. Sellers, M. Y. Hsiao, and L. W. Bearnson. Analyzing errors with the Boolean difference. *IEEE Trans. Computers*, C-17(7):676–683, 1968.

[12] R. Smolensky. Algebraic methods in the theory of lower bounds for Boolean circuit complexity. In *Proc. 19th Annual ACM STOC*, pages 77–82, 1987.

[13] L. J. Stockmeyer. The polynomial-time hierarchy. *Theoretical Computer Science*, 3(1):1–22, 1976.

[14] A. Thayse. *Boolean Calculus of Differences*. LNCS vol. 101. Springer, 1981.

[15] S. Toda. PP is as hard as the polynomial-time hierarchy. *SIAM J. Computing*, 20(5):865–877, 1991.

[16] I. Wegener. *The Complexity of Boolean Functions*. Wiley–Teubner, 1987.

[17] I. I. Zhegalkin. On the technique of calculating propositions in symbolic logic. *Matematicheskii Sbornik*, 34:9–28, 1927.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il