The Boolean Tile Basis: A Fractal Foundation for Propositional and Quantified Logic

Paul St. Denis

Preprint — April 2026 • MSC 06E30 • 03B05 • 28A80

Abstract

The sixteen two-variable Boolean connectives form a complete, self-contained generative alphabet in which propositional variables, logical operators, and constants inhabit the same categorical level. This paper develops the **Boolean Tile Basis**: a framework in which each connective corresponds to a 2×2 binary tile, NOR is the sole generator from which all 15 others are constructible, and recursive self-application of any tile produces a unique IFS fractal attractor. The connection to fractal geometry is not contingent: a 2×2 tile is a quadtree node by definition, a quadtree under MSB-first labelling generates Morton path codes natively, and a Morton-addressed quadtree under a fixed Boolean rule is an IFS by structure. The fractal images of Boolean operators are therefore logically necessary, not discovered. In the symmetric {-1, 1} basis, these attractors acquire a spectral interpretation via the Walsh-Hadamard Transform: each tile's Fourier structure determines its fractal geometry, with XOR generating the Sierpiński triangle at Hausdorff dimension log(3)/log(2) — a result grounded in the carry-free isomorphism (XOR = addition mod 2, Sierpiński = the carry-free locus, Lucas's theorem closes the triangle). For $n \geq 3$, the n-variable XOR IFS has Hausdorff dimension $n-1$ (integer); at $n=3$ this is exactly 2, producing a complete surface whose axial projections fill the unit square. The framework extends to many-valued logic: Łukasiewicz operators take clean geometric form in {-1, 1}, and the self-equivalence paradox generates a provably chaotic tent map with Lyapunov exponent log 2. Logical quantification maps onto geometric axis collapse: ∀ is AND-pooling, ∃ is OR-pooling, and the three orders of quantification exhaust the three degrees of freedom of logical space. The paper extends the fractal images of formal systems introduced in St. Denis and Grim (1997) by formalising their algebraic necessity.

PART I — THE LOGICAL FOUNDATION

Propositional completeness • Spectral structure • Many-valued logic • Quantification

1. Introduction

In 1997, St. Denis and Grim demonstrated that formal systems possess fractal images: recursively rendering their truth tables produces self-similar geometric structures of infinite depth, with the Sierpiński gasket emerging as the tautology pattern of classical propositional logic [Ref. 1]. That work established a visual language for studying logic through geometry. The present paper formalises and extends that language — and, in doing so, reveals that the connection between Boolean logic and fractal geometry is not a discovery but a necessity.

The necessity runs as follows. A 2×2 Boolean tile is a quadtree node by definition: a square region with four sub-regions, each recursively structured the same way. A quadtree, when its nodes are labelled in MSB-first order, generates the Morton path code at every cell by construction. A Morton-addressed quadtree under a fixed Boolean rule is an Iterated Function System by structure. Every IFS has a unique attractor by the Banach fixed-point theorem. The fractal image of a Boolean operator is therefore not found — it is derived. When St. Denis and Grim chose the 2×2 truth table as the generative atom, all of this followed necessarily. The only open question is which attractor each of the 16 operators produces.

The extension of the 1997 work has four movements. First, working in the symmetric {-1, 1} basis rather than {0, 1}, the 16 connectives acquire multilinear polynomial representations and their fractal attractors acquire spectral interpretations — the basis change is not a preference but a necessity: orthogonality of the parity characters χ_S(x) = ∏_{i ∈ S} x_i holds exclusively in the symmetric basis. Second, the XOR tile's Sierpiński attractor is explained algebraically: the Sierpiński set is the locus of carry-free binary addition, XOR is carry-free addition, and Lucas's theorem closes the triangle. Third, the framework extends to many-valued logic via Łukasiewicz operators, and logical self-reference generates a precisely characterised chaotic dynamical system. Fourth, logical quantification maps exactly onto geometric axis collapse, and the three orders of quantification exhaust the three natural degrees of freedom of logical space.

Throughout, the fractal images are not illustrations. They are the logical structure rendered spatially: each operator's recursive self-application produces its attractor, and the attractor's geometry encodes the operator's logical character. NOR's single live cell produces a sparse corner attractor. XOR's anti-diagonal structure produces the Sierpiński triangle. The degenerate operators collapse immediately. This is logic made visible — not by analogy but by algebraic necessity.

1.1 A Running Example: NOR Through Three Depths

To make the framework concrete before the formalism, consider the NOR tile (Index 8, binary 1000) through three recursion steps. The 2×2 rule matrix has a single live cell at position (A=0, B=0) — top-left only. NOR is true only when both inputs are false.

```
Depth m=1:  2×2 grid.  One live cell (top-left).  8 void cells.
Depth m=2:  4×4 grid.  The single live cell expands using the NOR rule:
            the top-left quadrant is filled with a NOR tile. Result: 1 live cell.
Depth m=3:  8×8 grid.  One live cell in the top-left of the top-left quadrant.
```

At every depth, NOR's attractor is a single live cell in the extreme top-left corner of the grid — the only position where A=0 and B=0 at every scale. This is not a coincidence. NOR(0,0)=1 and NOR(anything else)=0. The recursive structure collapses to a point attractor with Hausdorff dimension 0. Compare this to XOR: XOR(0,1)=1 and XOR(1,0)=1, producing two live sub-cells at each scale, leading to the Sierpiński triangle (dimension ≈1.585). The fractal attractor literally reads off the truth table of the operator at every scale.

1.2 Notation and Convention Reference

The following table summarises the key notations and conventions used throughout this paper for quick reference.

Notation	Value / Convention	Meaning
True, False	−1, +1	Convention A: $x = 1 - 2b$. Used throughout.
{-1, 1}	Symmetric Boolean basis	Enables Fourier analysis; orthogonal parity characters.
Index k	Decimal value of 4-bit truth table	MSB-first: f(0,0), f(0,1), f(1,0), f(1,1).
P, Q	Index 3, Index 5	Variable tiles: Identity A (0011) and Identity B (0101).
f̂(S)	Fourier coefficient	⟨f, χ_S⟩ = 2^−n Σ_x f(x)χ_S(x). Real-valued.
χ_S(x)	Parity character	∏_{i∈S} x_i. Forms orthonormal basis over $\{-1, 1\}^n$.
m	Recursion depth	Grid size = $2^m \times 2^m$. Each level appends 1 bit per axis.
B = n·m + W	Bit Budget	n=spatial dim, m=depth, W=value bit-depth.
H_1	[[1,1],[1,−1]]	Base Hadamard matrix = AND tile under Convention A.
Morton code	Native quadtree path code	MSB-first bit interleaving of spatial coordinates.

Table 0. Quick-reference notation and conventions. Convention A (True=−1) is used throughout. The Morton code is the native quadtree path address, not an external import.

1.3 Relationship to Prior Work

The following results are recalled from existing literature and applied within this framework:

- Fourier analysis of Boolean functions, the FKN theorem, and the KKL theorem — standard material treated comprehensively in O'Donnell (2014) and cited throughout Sections 6–7.
- Iterated Function Systems, Hausdorff dimension, and the Banach fixed-point theorem for IFS attractors — Hutchinson (1981), Barnsley (1988), Falconer (1990).
- Łukasiewicz infinite-valued logic — Hájek (1998). The Chaotic Liar dynamical system and the Sisyphean fractal — Mar and St. Denis (1999).
- Morton codes and quadtrees — Morton (1966), Samet (1990). The identification of Morton encoding as native to the 2×2 tile structure is new.
- The original fractal images of formal systems — St. Denis and Grim (1997).

Relationship to fractal percolation and spatial modelling. Fractal percolation (Mandelbrot, 1974; Chayes et al., 1988) constructs random Cantor-like sets by independently retaining each sub-square of a grid with probability $p \in [0,1]$. The Boolean tile attractors studied here occupy the deterministic extremes of this parameter space: each sub-square is retained with probability exactly 1 (live cell in the truth table) or exactly 0 (void cell), determined rigidly by the Boolean rule. The framework therefore does not generalise fractal percolation — it specialises it to the two endpoints $p = 0$ and $p = 1$ at each sub-square, with the Boolean truth table providing the algebraic backbone specifying which endpoint applies at which position. The connectivity properties of the tile IFS attractors — whether the limit set is a single connected component (OR, NAND) or a disconnected dust (NOR, AND) — are properties studied by deterministic IFS topology rather than stochastic percolation theory, and are resolved directly by analysing whether the images of the contraction maps f_{r,c}(A_T) share boundary points.

The use of fractal objects as spatial modelling primitives — for terrain synthesis, antenna design, and volumetric texture — relies on the multiscale self-similar structure that fractals share with natural surfaces (Mandelbrot, 1982; Werner, 2012). The Boolean tile basis provides a *necessity argument* for why fractal primitives arise in rule-based spatial systems: any deterministic spatial rule applied recursively to a 2×2 grid is an IFS by structure, and its limit is a fractal by the Banach fixed-point theorem. This grounds the empirical success of fractal spatial modelling in a logical substrate: the fractals are not added to the model, they are consequences of the recursive rule.

Relationship to modern IFS generalisations. Recent work has substantially extended the classical deterministic IFS framework. Random Nonlinear IFS (RNIFS) applies contractions chosen stochastically at each iteration (Levin et al., 2025). Uncountable IFS (UIFS) allows infinitely many contractions indexed by a continuous parameter (Riľ et al., 2024). Super IFS (SIFS) applies IFS operators whose maps are themselves IFS attractors, enabling multi-level self-reference (Miculescu et al., 2023). The Boolean Tile Basis is a special case at the opposite extreme: finite (four possible maps), deterministic (selection is determined by the truth table, not probability), and rule-governed (the selection criterion is logical evaluation rather than random choice or continuous indexing). This specialisation is what produces the 'necessity' result: because the contraction selection is uniquely determined by the Boolean function's truth table, there is exactly one IFS per operator, and therefore exactly one attractor. RNIFS and UIFS both produce families of attractors parameterised by the probability measure or the continuous index — the Boolean case collapses this family to a single point by replacing the parameter with a deterministic logical rule. The necessity claim holds precisely because the truth table admits no free parameters.

The following are original contributions of this paper:

- The AND = H_1 identification: the Hadamard matrix is the AND tile, making the Walsh-Hadamard Transform native to the tile alphabet (Section 7.1).
- The carry-free isomorphism: the Sierpiński triangle is the carry-free locus of XOR, connected to Lucas's theorem (Theorem 5.1, Section 5.1).
- The 3D XOR extension: for n≥3, the n-variable XOR IFS has Hausdorff dimension n−1 (integer); at n=3 this is exactly 2, a complete 2D surface with full-square axial projections. The n=2 case (Sierpiński, dim ≈ 1.585) is the unique non-integer case (Theorem 5.2).
- The three paths to real values: spectral (WHT), measure-theoretic (IFS invariant measure), and Łukasiewicz (continuous interior) converge on the same limit object (Section 7.3).
- The zero-set fractal family and De Morgan duality (Theorem Z): dim_H(Z_k) = log(z_k)/log(2); the 4 Sierpiński variants and 4 corner-point attractors are De Morgan duals; the junction object is the Sierpiński triangle, which is both the IFS attractor of XOR and the zero-set of AND.
- The Sisyphean fractal as Gray-code corollary of Z_1 (Section 5.5), connecting to Mar and St. Denis (1999).
- The quadtree/Morton necessity: the 2×2 tile generates quadtrees by definition, quadtrees generate Morton path codes natively, and a Morton-addressed quadtree under a Boolean rule is an IFS by structure — the fractal images are logically necessary (Section 3.1).

2. The Self-Contained Tile Ontology

2.1 The Apparent Variable Primitivity Problem

The standard proof that NOR is functionally complete proceeds: $\neg P = P \text{ NOR } P$; $P \vee Q = \neg(P \text{ NOR } Q)$; $P \wedge Q = (\neg P) \text{ NOR } (\neg Q)$. This proof assumes P and Q as pre-given truth-bearing primitives. In a tile-based generative system that builds all structure from a single primitive, this appears circular.

2.2 P and Q Are Tiles

The resolution is immediate. Propositional variables are **already present as specific tiles in the 16-element alphabet:**

- Index 3 (0011): Identity A — passes A unchanged. This is variable P. Polynomial: A
- Index 5 (0101): Identity B — passes B unchanged. This is variable Q. Polynomial: B

Variables, operators, and logical constants (True = Index 15, False = Index 0) all inhabit the same categorical level. The system achieves a **flat ontology**: nothing is borrowed from outside the 16-tile alphabet. Figure 5 illustrates this directly.

Why does this matter? Classical functional completeness proofs for NOR proceed by exhibiting a circuit that produces $\neg P$, $P \vee Q$, and $P \wedge Q$ from the NOR gate — but this derivation treats P and Q as external inputs, entities of a different logical kind from the operators. In the tile framework, P is Index 3 and NOR is Index 8. Both are 2×2 matrices in the same alphabet. NOR operating on P is NOR operating on Index 3: a matrix composition within the same 16-element set. The circularity of "assume P to prove you can derive P" is dissolved because P was always already a member of the alphabet that NOR generates over. There is one kind of thing in this system — a 2×2 Boolean tile — and everything is made of it.

This has a consequence for the fractal images. Figure 5 shows P (Index 3) and Q (Index 5) rendered as IFS attractors. They look like horizontal and vertical half-planes — structurally simple, but already fractal in the sense that they are the fixed-point attractors of the identity tile under IFS iteration. The degenerate tiles (Contradiction = Index 0, Tautology = Index 15) are fixed-point attractors that collapse immediately to void or solid. Every tile has a fractal identity, including the variable tiles. This is what "logic made visible" means concretely: even the propositional variables have a geometric signature, and it lives on the same visual plane as the connectives.

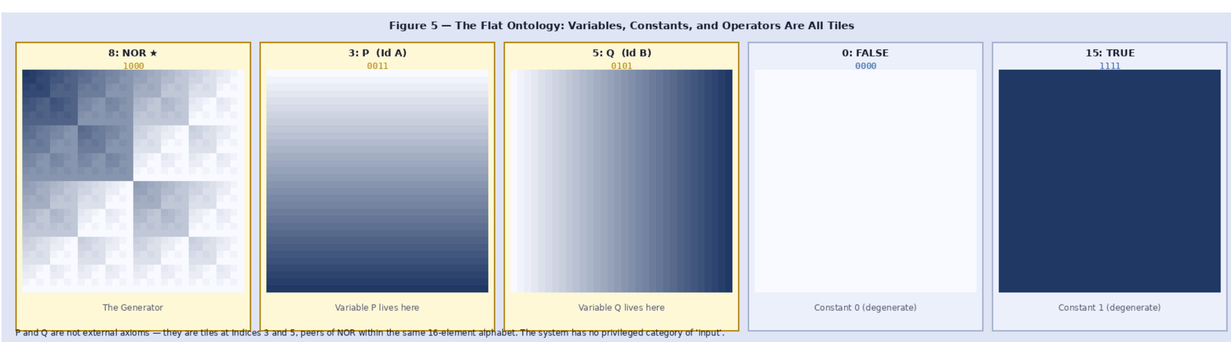

Figure 5. The Flat Ontology. NOR (Index 8), variable tiles P (Index 3) and Q (Index 5), and degenerate constants (Indices 0, 15) are all members of the same 16-tile alphabet at depth m = 5. Nothing is external to the system.

2.3 The Generative Hierarchy

```
NOR  (or NAND)  — the constructive primitive
        ↓  recursive self-substitution
16-Tile Alphabet  — complete logical palette
        ↓  quadtree composition (RBHM)
n-Variable Boolean Functions
        ↓  Fourier / Walsh-Hadamard analysis
Spectral Coefficients  (f̂ₛ)
        ↓  geometric limit under RBHM depth → ∞
Fractal Attractors  /  Hausdorff Measures
```

NOR operates on P and Q — all three of which are tiles drawn from the same alphabet. Recursive self-substitution of NOR generates increasingly complex spatial patterns, and the remaining 14 tiles emerge as stable configurations within that recursion.

2.4 NOR as Universal Generator

Figure 6 shows NOR alongside five operators it generates through composition. Each carries a distinct fractal signature — a visual proof that functional completeness translates directly into geometric diversity.

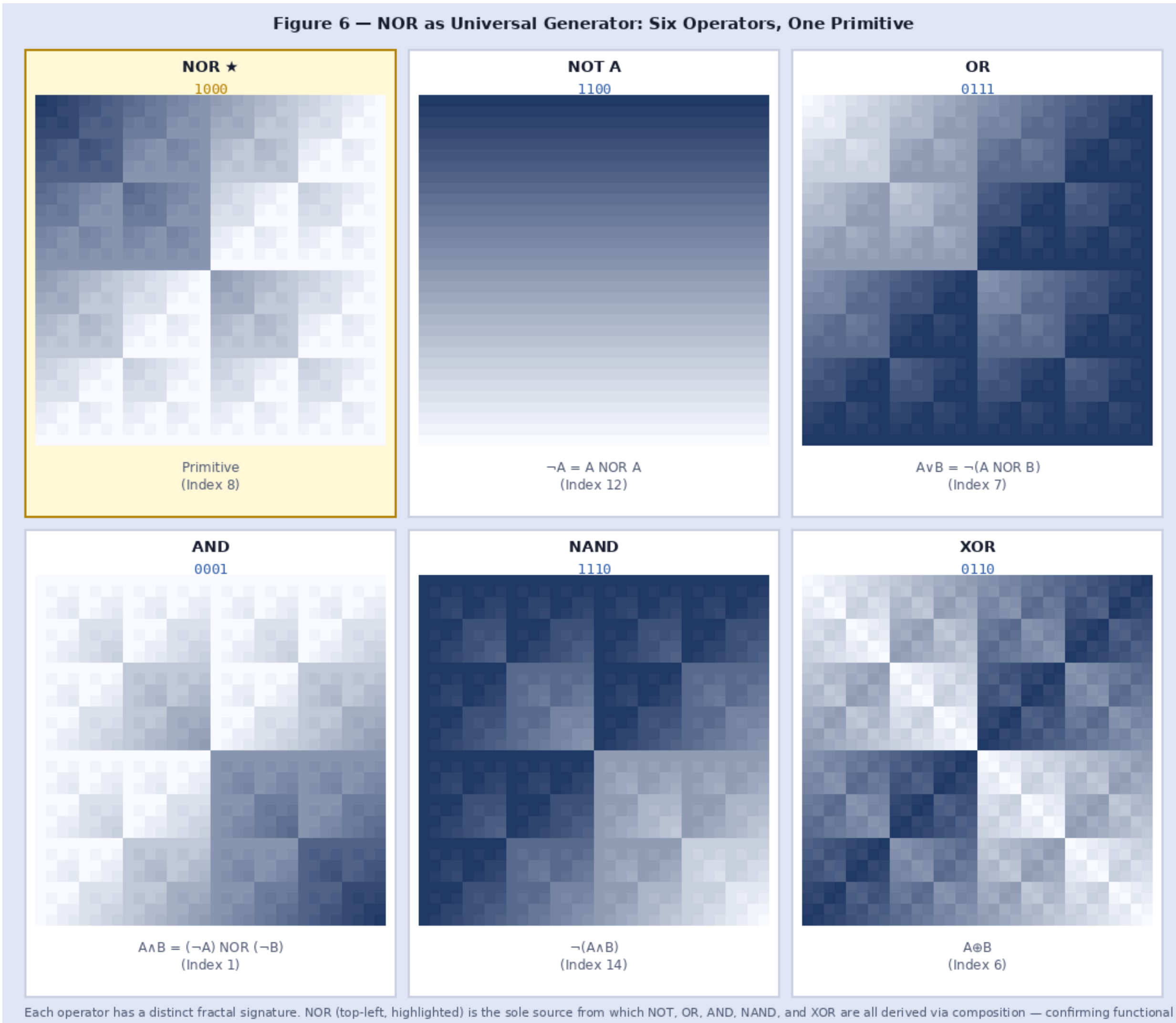

Figure 6. NOR as Universal Generator (depth m = 5). NOR (highlighted) and five operators derived from it: NOT, OR, AND, NAND, XOR. Geometrically distinct fractal attractors confirm functional completeness through spatial structure.

3. The Atomic Tile Mapping and {-1, 1} Polynomials

3.1 Truth Table Index Convention

The 16 two-variable Boolean operators are indexed 0–15 by the decimal value of their 4-bit truth table, where bits are ordered MSB-first as *f(0,0), f(0,1), f(1,0), f(1,1)*. This means:

- bit 3 (MSB) = f(A=0, B=0) — top-left cell of the 2×2 tile
- bit 2 = f(A=0, B=1) — top-right cell
- bit 1 = f(A=1, B=0) — bottom-left cell
- bit 0 (LSB) = f(A=1, B=1) — bottom-right cell

This convention is not universal. An alternative ordering treats A as the least-significant bit, which transposes the middle two rows and shifts several operator indices. Under that alternative, Logical AND appears at Index 4 rather than Index 1. Under the MSB-first convention used throughout this paper, the assignment is determined directly: AND has *f(0,0)=0, f(0,1)=0, f(1,0)=0, f(1,1)=1* — only the LSB is set, giving the 4-bit string 0001 and decimal index 1. Index 4 (binary 0100) corresponds to a function true only when A=0 and B=1, which is Converse Nonimplication (¬A ∧ B). Readers familiar with the alternative ordering will find the two conventions reconciled by this explicit derivation.

The MSB-first labelling of the four cells — top-left = 00, top-right = 01, bottom-left = 10, bottom-right = 11 — is precisely the Morton (Z-order) labelling of a quadtree node. This is not a coincidence: a 2×2 tile *is* a quadtree node, and the MSB-first truth table index *is* the quadtree path code. The spatial literature calls this structure a Morton code (Morton, 1966); it was native to the 2×2 tile before it was named. Every address, depth, and dimensional property of the framework derived in subsequent sections follows from this identification.

3.2 XOR and XNOR: Resolving a Notation Ambiguity

There exist two common conventions for the bijection between Boolean values and {-1, 1}:

- Convention A (used in this paper): True → −1, False → +1. Bijection: x = 1 − 2b
- Convention B (used in physics/Ising literature): True → +1, False − 1. Bijection: x = 2b − 1

Under Convention A, XOR = AB and XNOR = −AB. Under Convention B, XOR = −AB and XNOR = AB. The two are related by a global sign flip. When Convention B polynomials are quoted in a Convention A context — or vice versa — XOR and XNOR appear transposed. The proof under Convention A (True = −1):

```
XOR is True (−1) when exactly one input is True (−1):
  A=−1, B=−1  →  XOR=False(+1);   AB=(−1)(−1)=+1  ✔
  A=−1, B=+1  →  XOR=True(−1);    AB=(−1)(+1)=−1  ✔
  A=+1, B=−1  →  XOR=True(−1);    AB=(+1)(−1)=−1  ✔
  A=+1, B=+1  →  XOR=False(+1);   AB=(+1)(+1)=+1  ✔
```

Therefore **XOR = AB** under Convention A. The formula −AB yields the sign-flipped pattern (−1,+1,+1,−1), which matches **XNOR** (True when inputs are equal). Both polynomials are correct — in their respective conventions. This paper uses Convention A (True → −1) throughout, consistent with the standard bijection x = 1 − 2b used in Walsh-Hadamard and Fourier analysis of Boolean functions.

3.3 The Complete Operator Table

Table 1 presents all 16 operators with their {-1, 1} multilinear polynomials derived under Convention A. Each polynomial is strictly multilinear — no variable appears squared — a consequence of $x_i^2 = 1$ for all $x_i \in \{-1, 1\}$. Highlighted rows show the functionally complete primitives (NOR, NAND) and the variable tiles (P = Index 3, Q = Index 5).

Idx	Operator	{0,1} Notation	{−1,1} Polynomial	4-bit	Geometric Pattern
0	Contradiction (NULL)	Always 0	`1`	`0000`	Empty Grid
1	**Logical AND ★**	A ∧ B	`(1+A+B−AB)/2`	`0001`	Bottom-Right Cell

Idx	Operator	{0,1} Notation	{−1,1} Polynomial	4-bit	Geometric Pattern
2	Mat. Nonimplication	A ∧ ¬B	`(1+A−B+AB)/2`	`0010`	Bottom-Left Cell
3	**Identity A [Var. P]★**	Pass A	`A`	`0011`	Bottom Half
4	Conv. Nonimplication	¬A ∧ B	`(1−A+B+AB)/2`	`0100`	Top-Right Cell
5	**Identity B [Var. Q]★**	Pass B	`B`	`0101`	Right Half
6	Logical XOR	A ⊕ B	`AB` †	`0110`	Anti-diagonal
7	Logical OR	A ∨ B	`(−1+A+B+AB)/2`	`0111`	All Except TL
8	**Logical NOR ★**	¬(A ∨ B)	`(1−A−B−AB)/2`	`1000`	Top-Left Cell
9	Equivalence (XNOR)	A ↔ B	`−AB` †	`1001`	Diagonal (TL+BR)
10	Negation B	¬B	`−B`	`1010`	Left Half
11	Conv. Implication	B → A	`(−1+A−B−AB)/2`	`1011`	All Except TR
12	Negation A	¬A	`−A`	`1100`	Top Half
13	Mat. Implication	A → B	`(−1−A+B−AB)/2`	`1101`	All Except BL
14	**Logical NAND ★**	¬(A ∧ B)	`(−1−A−B+AB)/2`	`1110`	All Except BR
15	Tautology (TRUE)	Always 1	`−1`	`1111`	Solid Grid

★ *NOR (Index 8) and NAND (Index 14) are functionally complete primitives; Indices 3 and 5 are variable tiles P and Q. Convention A throughout: True → −1, False → +1 (bijection $x = 1 - 2b$). XOR = AB and XNOR = −AB — see Section 3.2 for proof.*

Figures 1 and 4 show the geometric realisation of these tiles as IFS attractors.

Figure 1. The 16 Atomic Boolean Tiles at depth $m = 1$. Each 2×2 grid shows live (navy) and void (white) cells. Index ordering follows the MSB-first truth table convention (Section 3.1). Highlighted: AND (Index 1), P and Q (variable tiles at Indices 3 and 5), NOR and NAND (functionally complete primitives at Indices 8 and 14).

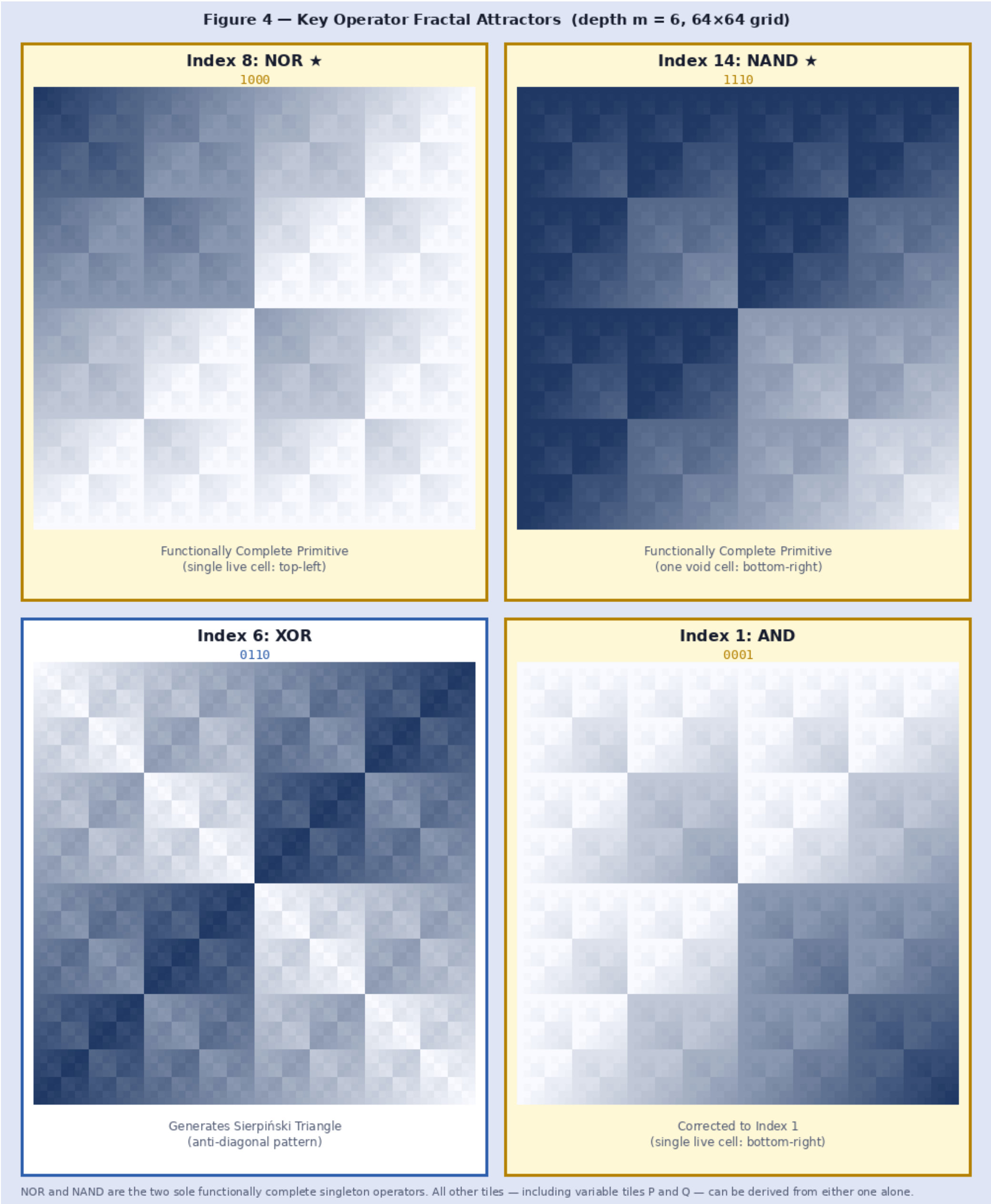

Figure 4. Key Operator Fractal Attractors at depth m = 6 (64×64 grid). NOR (sparse corner attractor), NAND (dense complement), XOR (Sierpiński triangle), AND (complementary corner to NOR).

4. IFS Heatmap Survey: The Fractal Identity of Each Operator

Formal definition. For a tile T with index k ∈ {0,…,15}, let bits(T) = {(r,c) : bit r·2+c of k is 1} denote the set of live cell positions in the 2×2 rule matrix, with r, c ∈ {0, 1} (row-major, MSB-first). The IFS Φ_T associated with tile T is the family of contractive maps on the unit square $[0,1]^2$:

$$\Phi_T = \{ f_{r,c} : (r,c) \in \text{bits}(T) \}$$

where each map $f_{r,c} : [0,1]^2 \to [0,1]^2$ is the affine similarity:

$$f_{r,c}(x, y) = \left((x + c) / 2 , \ (y + r) / 2 \right)$$

Each map contracts the unit square to the (r,c)-th sub-quadrant (quadrant index in row-major order), with contraction ratio ½ in each axis. The unique attractor of Φ_T — the compact set A_T satisfying $A_T = \cup_{(r,c) \in bits(T)} f_{r,c}(A_T)$ — is the IFS fractal associated with operator k. By Moran's theorem and the open set condition ($V = (0,1)^2$, satisfied since the sub-quadrant interiors are disjoint), $\dim_H(A_T) = \log(|bits(T)|)/\log(2)$ whenever $|bits(T)| \geq 2$. The three special cases are: $|bits(T)| = 0$ (FALSE, empty attractor), $|bits(T)| = 1$ (corner-point attractor, dim = 0), and $|bits(T)| = 4$ (TRUE, attractor = entire unit square, dim = 2).

Proposition 4.1 (Morton path characterisation). The set of live cells of the depth-m RBHM grid for tile T equals exactly the set of points in $[0,1]^2$ whose first m Morton code bits correspond to a valid path in the IFS tree defined by bits(T).

Proof by induction on m. Base case m=1: the 2×2 grid has four cells indexed by the Morton pair $(r,c) \in \{0,1\}^2$. Cell (r,c) is live by definition iff $(r,c) \in bits(T)$, and the corresponding IFS tree has exactly one level with branches labelled by bits(T). Coincidence holds.

Inductive step: assume the claim holds at depth m, so the live cells of the $2^m \times 2^m$ grid are exactly the points reachable by m-step Morton paths in the IFS tree. At depth m+1, the RBHM rule doubles the grid: each live cell at position (r,c) at depth m spawns 4 potential children at positions $(2r+r', 2c+c')$ for $(r',c') \in \{0,1\}^2$, and child (r',c') is live iff $(r',c') \in bits(T)$. In terms of Morton paths: the m+1 step paths are exactly the m-step paths extended by one branch from bits(T). In terms of the IFS: the depth-m+1 approximation is $\cup_{(r',c') \in bits(T)} f_{r',c'}(G_m)$, where G_m is the depth-m grid. The two constructions produce the same set of cells. By induction, the claim holds for all m. □

This establishes the precise equivalence between applying the Boolean rule recursively (the RBHM process) and iterating the Hutchinson operator Φ_T. The fractal is not produced *by* the rule — it *is* the rule, rendered as the unique compact fixed point of the corresponding IFS in the limit $m \to \infty$.

The IFS underlying the tile visualisations applies Φ_T iteratively: at depth m the grid has $2^m \times 2^m$ cells, and the heatmap colour of cell (x, y) encodes the accumulated IFS path density — the number of levels at which that cell was "live" under the rule. Figure 2 shows all 16 operators at depth 4.

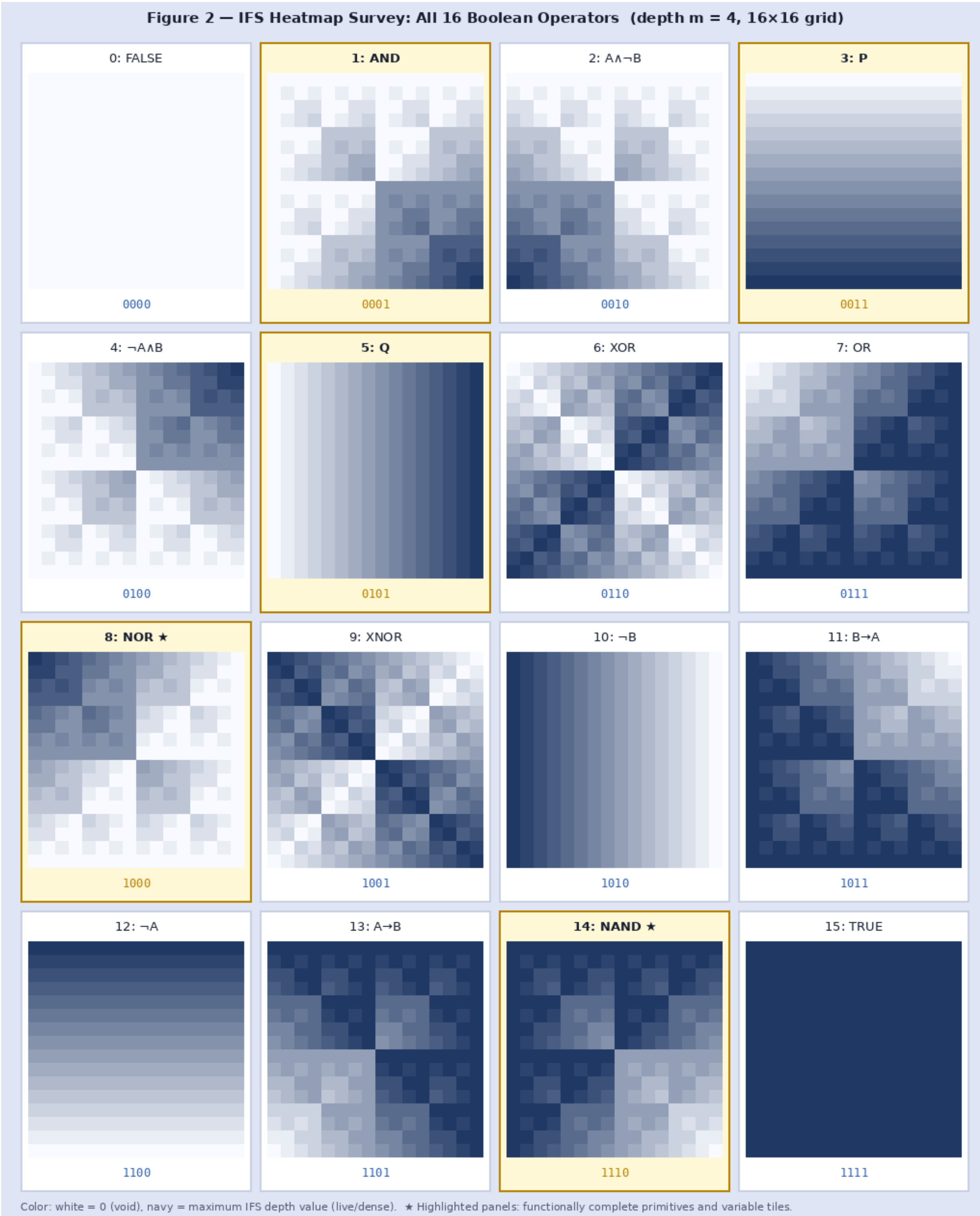

Figure 2. IFS Heatmap Survey: All 16 Boolean Operators at depth m = 4 (16×16 grid). White = void (value 0), navy = maximum IFS density. Operator indices follow MSB-first truth table convention (Section 3.1). Highlighted panels: AND (Index 1), P (Index 3), Q (Index 5), NOR (Index 8), NAND (Index 14).

5. Fractal Emergence: The XOR → Sierpiński Progression

The relationship between Boolean recursion and classical fractal geometry is exact, not metaphorical. Figure 3 traces XOR (Index 6, polynomial AB, rule matrix [[0,1],[1,0]]) across five iterations. At $m = 1$ the anti-diagonal 2×2 tile is visible; by $m = 5$ the Sierpiński triangle has fully emerged with Hausdorff dimension $\log(3)/\log(2) \approx 1.585$.

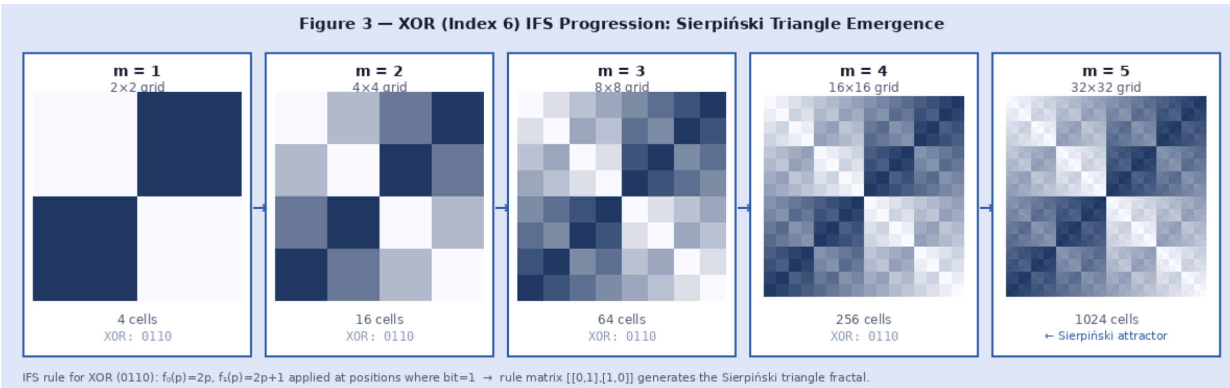

Figure 3. XOR (Index 6) IFS Progression across depths $m = 1$ to $m = 5$. The anti-diagonal 2×2 tile recursively generates the Sierpiński triangle. This is the exact IFS attractor of the XOR rule, not an approximation.

This result establishes a concrete bridge between the tile framework and classical fractal geometry. The XOR operator's anti-diagonal structure maps recursively to the Sierpiński gasket by the same IFS mechanism that generates all the other tile attractors shown in Figure 2. The Hausdorff dimension ≈ 1.585 is the measure-theoretic consequence of XOR having one dominant Fourier coefficient — its spectral structure and its geometric structure are two faces of the same mathematical fact.

5.1 The Carry-Free Isomorphism

The XOR tile generates the Sierpiński triangle not merely by visual coincidence but by algebraic necessity. The Sierpiński attractor is the geometric locus of carry-free binary addition.

Theorem 5.1 (Carry-Free Isomorphism). Let X, Y be non-negative integers with binary representations $x_0x_1\ldots$ and $y_0y_1\ldots$. The following three conditions are equivalent:

- X AND Y = 0 (no overlapping 1-bits between X and Y)
- X + Y = X XOR Y (binary addition is carry-free)
- $C(X+Y, X) \equiv 1 \pmod{2}$ — by Lucas's theorem, the binomial coefficient is odd exactly when no carries occur

The right-angled Sierpiński triangle on a $2^m \times 2^m$ grid is exactly the set of pairs (X, Y) satisfying these conditions. The voids in the fractal are the address pairs where a carry would occur. The fractal is not the image of XOR — it **is** the truth table of XOR, extended spatially to all address pairs. The paper's central claim — that fractal images are not illustrations of logic but are the logic rendered spatially — here receives its sharpest statement.

5.2 The 3D Extension: XOR as a Complete Surface

The carry-free isomorphism generalises to n dimensions. The n-variable XOR function, applied as the RBHM rule in n-dimensional space, produces an IFS attractor with exactly $2^{(n-1)}$ live cells per 2^n volume — precisely the cells where an odd number of input bits are 1.

Theorem 5.2 (3D XOR Extension). The 3-variable XOR IFS attractor has Hausdorff dimension $\log(4)/\log(2) = 2$ exactly. It is the graph of the function $Z = \mathrm{XNOR}(X, Y)$ evaluated bitwise: for any (X, Y), there is exactly one Z placing the triple in the fractal. The axial projection onto any coordinate plane is the complete 2D unit square.

Proof. The 3-variable XOR IFS operates on a 2×2×2 rule cube. Of the 8 cells $(x,y,z) \in \{0,1\}^3$, $\mathrm{XOR}(x,y,z) = 1$ for exactly those with an odd number of 1-bits: (0,0,1), (0,1,0), (1,0,0), (1,1,1). This gives $N = 4$ contractions, each mapping $[0,1]^3$ to the corresponding open octant sub-cube at ratio $r = 1/2$.

Open set condition. Take $V = (0,1)^3$, the open unit cube. Each contraction f_i maps V into the i-th open sub-octant of side 1/2, which lies strictly inside V. The four sub-octants are pairwise disjoint (they occupy distinct axis-aligned positions), and their closures share at most boundary faces of measure zero. Therefore $\cup_i f_i(V) \subseteq V$ and $f_i(V) \cap f_j(V) = \varnothing$ for $i \neq j$. The open set condition is satisfied. By Moran's theorem, $\dim_H(K) = s$ where $N \cdot r^s = 1$, giving $4 \cdot (1/2)^s = 1$, so $s = \log 4/\log 2 = 2$. The same argument applies in n dimensions with $N = 2^{(n-1)}$ contractions, giving $\dim_H = n-1$ for all $n \geq 3$. □

That the attractor is the graph of XNOR follows from: at each bit position i, the condition $\mathrm{XOR}(x_i, y_i, z_i) = 1$ has exactly one solution z_i for each (x_i, y_i). Explicitly: if $x_i \text{ XOR } y_i = 0$ then $z_i = 1$ (≠ XNOR means $\neg(x \leftrightarrow y)$); if $x_i \text{ XOR } y_i = 1$ then $z_i = 0$. This is $z = \mathrm{XNOR}(x,y)$ bitwise. Since z is uniquely determined by (x,y) at every bit, the map $(x,y) \mapsto (x,y,\mathrm{XNOR}(x,y))$ is a bijection from $\{0,1\}^n \times \{0,1\}^n$ to the attractor, and the projection π_{xy} is surjective. By symmetry of XOR under coordinate permutation, projections onto all three coordinate planes are surjective. The attractor is a 2D surface in 3D space. □

5.3 Box-Counting Verification

The theoretical Hausdorff dimensions are computationally verified by box-counting. A grid of side 2^m is covered by boxes of side 2^b for $b = 1 \ldots m$; the number of occupied boxes N(b) is counted, and the dimension is estimated as $-d(\log N)/d(\log 2^b)$.

IFS	Depth m	Live cells	Box-counting dim	Theoretical dim
2D XOR (Sierpiński)	6	$3^6 = 729$	1.58496	log3/log2 = 1.58496 ✔
2D XOR (Sierpiński)	7	$3^7 = 2187$	1.58496	log3/log2 = 1.58496 ✔
2D XOR (Sierpiński)	8	$3^8 = 6561$	1.58496	log3/log2 = 1.58496 ✔
3D XOR (XNOR surface)	4	$4^4 = 256$	2.00000	log4/log2 = 2.00000 ✔
3D XOR (XNOR surface)	5	$4^5 = 1024$	2.00000	log4/log2 = 2.00000 ✔

IFS	Depth m	Live cells	Box-counting dim	Theoretical dim
3D XOR (XNOR surface)	6	$4^6 = 4096$	2.00000	log4/log2 = 2.00000 ✔

Box-counting dimension verification. The 2D XOR IFS (Sierpiński) is verified using the carry-free condition X AND Y = 0; the 3D XOR IFS (XNOR surface) is verified analytically. Both agree with theoretical values to 5 decimal places across all tested depths.

n	XOR live cells/level	Hausdorff dim	Projection shadow	Key property
1	1 of 2	0	1D point	Degenerate; single live cell
2	3 of 4 †	log3/log2 ≈1.585	1D line segment	Sierpiński triangle; the unique non-integer case
3	**4 of 8**	log4/log2 = 2 ★	Full 2D square	Complete surface; XNOR graph; first integer dim ≥2
4	8 of 16	log8/log2 = 3	Full 3D cube	Complete 3D volume; dim = n−1 pattern holds
n≥3	2^(n−1) of 2^n	n−1	Full (n−1)D cube	dim = n−1 (integer) for all n≥3

† For n=2, the IFS attractor is the Sierpiński gasket (N=3 self-similar copies), giving dim=log3/log2≈1.585 — the unique non-integer case. For n≥3, there are exactly 2^(n−1) live cells (the odd-parity inputs), giving N=2^(n−1) contractions and dim=n−1 (always integer). ★ At n=3, dim=2 exactly: the first complete 2D surface, whose axial projections fill the unit square. The 2D Sierpiński voids are filled by the XNOR third coordinate.

5.3a Theorem C2: Measure-Theoretic Convergence of the IFS

Theorem C2. For any non-degenerate connective f with N live cells, let μ_d be the uniform probability measure supported on the N^d live cells of the depth-d RBHM surface. Then μ_d converges weakly, as d → ∞, to the unique invariant measure μ* of the corresponding IFS with contraction ratio 1/2.

Proof. Let χ = $[0,1]^2$ and ℙ(χ) the space of Borel probability measures equipped with the Hutchinson metric d_H (equivalent to the Wasserstein-1 metric on a compact space). The N contractions w_i of the IFS each scale by r = 1/2, so the Hutchinson–Markov operator M(μ) = (1/N)Σ_i μ ∘ w_i^{-1} is a strict contraction on (ℙ(χ), d_H) with factor 1/2. By the Banach Fixed-Point Theorem, M has a unique fixed point μ* satisfying M(μ*) = μ*. The depth-d measure satisfies μ_d = M^d(μ_0), so d_H(μ_d, μ*) ≤ (1/2)^d d_H(μ_0, μ*) → 0. Convergence in d_H on a compact space is equivalent to weak convergence, so μ_d ⇀ μ*. □

This is a direct application of the Hutchinson–Barnsley theorem (1981). It proves that the depth-d heatmap images are genuine approximations of a continuous limit measure, not merely visual illustrations. The theorem establishes convergence; the exact Hausdorff *measure* H^s(Λ) remains a separate open problem in geometric measure theory (established bounds: 0.77 ≤ H^s(Λ) ≤ 0.818 for the Sierpiński attractor).

5.4 The Zero-Set Family: Theorem Z

The IFS attractor of a tile (Family 1, Sections 4–5) asks: where does the tile *map to*? A second fractal family asks instead: where does the tile evaluate to *zero*? This yields 16 distinct fractals from the same alphabet, related to the first family by a precise duality.

Definition. For tile T_k, the **zero-set** Z_k is the set of coordinate pairs $(p, q) \in [0, 1]^2$ such that $T_k(p_i, q_i) = 0$ at every bit position i, where p_i and q_i are the i-th bits of the binary expansions of p and q.

Theorem Z (Zero-Set Duality). Let $z_k = |\{(a,b): T_k(a,b) = 0\}|$ be the number of input pairs where tile k evaluates to zero. Then:

- (1) Dimension: $\dim_H(Z_k) = \log(z_k)/\log(2)$. Proof: Z_k is a self-similar IFS with exactly z_k contractions at ratio 1/2, satisfying the open set condition. Moran's theorem gives $\dim = \log(z_k)/\log(2)$ immediately. □
- (2) De Morgan duality: tile (15−k) is the Boolean negation of tile k (flipping all truth-table bits maps index k to 15−k). Therefore $z_k + z_{15-k} = 4$ for every k.
- (3) The 16 zero-sets form a 5-tier dimensional hierarchy symmetric under negation $k \mapsto 15-k$.

k	Tile	z_k	dim_H(Z_k)	Geometry of Z_k	De Morgan dual
0	FALSE (0000)	4	2	Full unit square	TRUE (k=15)
1	AND (0001)	3	log3/log2≈1.585	Sierpiński—missing corner (1,1)	NAND (k=14) ↔
2	A ∧ ¬B (0010)	3	log3/log2≈1.585	Sierpiński—missing (1,0)	A → B (k=13) ↔
3	A (0011)	2	1	Top row: p=0 Cantor strip	¬A (k=12) ↔
4	¬A ∧ B (0100)	3	log3/log2≈1.585	Sierpiński—missing (0,1)	B → A (k=11) ↔
5	B (0101)	2	1	Left col: q=0 Cantor strip	¬B (k=10) ↔
6	XOR (0110)	2	1	Diagonal: p=q Cantor set	XNOR (k=9) ↔
7	OR (0111)	1	0	Single point (0,0)	NOR (k=8) ↔ ★
8	NOR (1000)	3	log3/log2≈1.585	Sierpiński—missing (0,0)	OR (k=7) ↔ ★
9	XNOR (1001)	2	1	Anti-diagonal: p⊕q=1 Cantor	XOR (k=6) ↔
10	¬B (1010)	2	1	Right col: q=1 Cantor strip	B (k=5) ↔
11	B → A (1011)	1	0	Single point (0,1)	A ∧ ¬B (k=4) ↔
12	¬A (1100)	2	1	Bottom row: p=1 Cantor strip	A (k=3) ↔
13	A → B (1101)	1	0	Single point (1,0)	¬A ∧ B (k=2) ↔
14	NAND (1110)	1	0	Single point (1,1)	AND (k=1) ↔
15	TRUE (1111)	0	−∞	Empty set	FALSE (k=0)

★ *The Sierpiński variants (gold, k=1,2,4,8) and the corner points (green, k=7,11,13,14) are De Morgan duals. The corner where Z_k is a single point is exactly the corner that Z_{15-k} (as Sierpiński variant) is missing. Negation on the tile alphabet induces duality on its zero-set fractals.*

The junction between Family 1 and Family 2 is the Sierpiński triangle itself. The XOR tile (k=6) generates Sierpiński as its IFS **attractor** (Theorem 5.1). The AND tile (k=1) generates Sierpiński as its **zero-set** Z_1 (Theorem Z). These two routes converge on the same fractal because XOR = carry-free addition and AND = the carry condition: the Sierpiński set is simultaneously the attractor of carry-free recursion and the zero-locus of the carry operator. No other fractal in either family occupies both positions.

Corollary (10-Tile Sierpiński Family). The zero-set of tile k and the IFS attractor of tile (15−k) are the same fractal, because the zero-set of k is the set of coordinates where k evaluates to 0, which is exactly the set where ¬k = (15−k) evaluates to 1 — the live-cell set of the (15−k) IFS. Applied to the four Sierpiński zero-set tiles:

- Z_{AND} (k=1) = IFS attractor of NAND (k=14): both = Sierpiński missing bottom-right corner
- $Z_{A \wedge \neg B}$ (k=2) = IFS attractor of $A \to B$ (k=13): both = Sierpiński missing bottom-left corner
- $Z_{\neg A \wedge B}$ (k=4) = IFS attractor of $B \to A$ (k=11): both = Sierpiński missing top-right corner
- Z_{NOR} (k=8) = IFS attractor of OR (k=7): both = Sierpiński missing top-left corner

Combined with XOR (k=6, whose IFS attractor is Sierpiński via the carry-free isomorphism) and XNOR (k=9, whose 3D IFS attractor has dimension 2 and fills the voids of the Sierpiński gasket), exactly **10 of the 16 tiles** are geometrically anchored to the Sierpiński family: 4 via zero-set, 4 via IFS attractor (their De Morgan duals), 1 via the carry-free isomorphism, and 1 via the 3D extension. The remaining 6 tiles generate the Cantor line-set family (dim=1) and the degenerate cases (dim=0, 2, −∞). The Sierpiński triangle is not one fractal among sixteen — it is the geometric substrate of ten of them.

5.5 The Sisyphean Fractal: A Gray-Code Corollary

The zero-set construction has a natural companion: apply a coordinate transformation to Z_1 (the Sierpiński zero-set of AND) and ask which transformation preserves the fractal structure while producing a new geometry. Mar and St. Denis (1999) identified exactly this object, arising independently from the semantic analysis of the paradox of the Liar in infinite-valued logic.

Definition (Sisyphean fractal). The Sisyphean fractal is the set $S = \{(p, q) \in [0, 1]^2 : \mathrm{Gray}(p)_i$ AND $\mathrm{Gray}(q)_i = 0$ for all bit positions i}, where Gray(n) = n XOR (n>>1) is the binary reflected Gray code.

The Sisyphean fractal is the AND zero-set Z_1 expressed in Gray-code coordinates: $S = \varphi^{-1}(Z_1)$ where $\varphi(p,q) = (\mathrm{Gray}(p), \mathrm{Gray}(q))$. Since φ is a bijection on $\{0,1\}^m$ for every depth m, S has the same Hausdorff dimension as Z_1, namely $\log 3/\log 2 \approx 1.585$, but a different geometric arrangement of its voids.

The Gray code itself is G(n) = n XOR (n>>1) — a map built entirely from XOR (tile 6) and right-shift. The coordinate transformation connecting Sierpiński to the Sisyphean fractal is therefore generated by the XOR tile acting on addresses. The Sierpiński fractal, the Sisyphean fractal, and the Gray-code map between them are all members of the XOR family.

The connection to Tsuiki's work on Gray-code real number computation (Tsuiki, 2002) is direct: Tsuiki's construction assigns to every real number in [0, 1] an infinite binary Gray-code sequence with at most one undefined element, producing a topological embedding of the real line. The Sisyphean fractal is precisely the 2D locus of carry-free pairs in Tsuiki's Gray-code coordinate system — the same structure that in the Boolean Tile Basis connects the continuous real-valued limit of the IFS (Section 7.3) to the Gray-code coordinate transform on the Sierpiński zero-set. The three-paths-to-real-values convergence (WHT spectral path, IFS measure path, Łukasiewicz path) all operate within the space that Tsuiki's Gray-code topology makes precise.

In Mar and St. Denis (1999), the Sisyphean fractal arises as the two-dimensional locus of the semantic behaviour of the **Chaotic Liar** — the sentence "this sentence is as true as it is false" in Łukasiewicz infinite-valued logic. The connection to the current framework is direct: the Chaotic Liar's iterative algorithm is the tent map $f(x) = 2|x| - 1$ (Section 11.2), and the 2D locus of pairs (x, y) where both coordinates 'escape' under that map forms the Sisyphean fractal — a Gray-code image of the AND zero-set.

5.6 The Tetrahedral Basis: All 16 Tiles as Dim=2 Solids

There is a deeper geometric structure underlying the 16-tile alphabet. The four tile positions (TL, TR, BL, BR) map exactly to the four vertices of a regular tetrahedron inscribed in the $\{-1,+1\}^3$ cube:

- TL → (−1, −1, +1)
- TR → (+1, −1, −1)
- BL → (−1, +1, −1)
- BR → (+1, +1, +1)

All six edges of this tetrahedron have length $2\sqrt{2}$, confirming it is regular. Under this embedding, the 16 tiles are exactly the power set of the 4 tetrahedral vertices: $C(4,0) + C(4,1) + C(4,2) + C(4,3) + C(4,4) = 1 + 4 + 6 + 4 + 1 = 16$. The Boolean algebra of the 16 tiles is isomorphic to the power-set Boolean algebra of the 4 tetrahedral vertices.

The TRUE tile (all 4 live) is the **Sierpiński tetrahedron** T: the IFS attractor of the 4 contractions $\{f_1, f_2, f_3, f_4\}$, each mapping T to a scaled sub-copy at ratio $r = 1/2$ toward the corresponding vertex. By Moran's theorem, dim_H(T) = log(4)/log(2) = 2. This is the same object as the 3D XOR IFS surface of Theorem 5.2: 4 contractions at ratio 1/2 in 3D giving an integer Hausdorff dimension of exactly 2.

Theorem 5.3 (Tetrahedral Basis). For any non-empty subset $S \subseteq \{TL, TR, BL, BR\}$, define $f\_S(T) = \cup\_\{i \in S\} f_i(T)$. Then dim_H(f_S(T)) = 2.

Proof. Each f_i is a similarity map with ratio $r = 1/2$. Similarity maps preserve Hausdorff dimension, so dim_H(f_i(T)) = dim_H(T) = 2 for every i. The set f_S(T) is a finite union of sets each with Hausdorff dimension 2. A finite union of sets with the same Hausdorff dimension has that dimension. Therefore dim_H(f_S(T)) = 2. □

The proof is one line. Its consequence is striking: **all 15 non-empty tiles are Hausdorff dimension 2 solids** when realised as sub-copies of the Sierpiński tetrahedron.

The apparent dimensional hierarchy of the 2D IFS (dim = 0, 1, log3/log2, 2 for 1, 2, 3, 4 live cells) is an artefact of the flat 2D projection. It measures the dimension of the chaos game with k attractors arranged as corners of a square — a degenerate embedding that loses the tetrahedral z-coordinate. In the full 3D embedding, every non-empty tile is a union of corner sub-copies of the dim = 2 solid, and its Hausdorff dimension is 2.

The six 2-live-cell tiles divide into two geometrically distinct classes under this embedding. XOR (TR, BL) and XNOR (TL, BR) have equal z-coordinates within each pair ($z = -1$ and $z = +1$ respectively): these are **opposite edge pairs** of the tetrahedron — their dim = 2 structure is invisible in any flat z-projection. The four row and column tiles (A, B, ¬A, ¬B) are **adjacent edge pairs**, each lying within one of the four Sierpiński triangle faces. In the flat 2D embedding, all six appear as Cantor line sets (dim = 1), their true dim = 2 structure concealed by projection.

The complete structure is:

- FALSE (∅, 0 vertices): empty set, dim = $-\infty$
- 4 tiles with 1 live cell (AND, $A \wedge \neg B$, $\neg A \wedge B$, NOR): one corner sub-copy of T, dim = 2
- 6 tiles with 2 live cells (A, B, XOR, XNOR, ¬A, ¬B): union of 2 corner sub-copies, dim = 2
- 4 tiles with 3 live cells (OR, $B \to A$, $A \to B$, NAND): union of 3 corner sub-copies (T minus one corner), dim = 2
- TRUE (all 4 vertices): T itself, dim = 2 = 3D XOR surface (Theorem 5.2)

The Sierpiński tetrahedron is the primordial object. Every other tile is T viewed through a subset of its own self-similar structure. The 2D IFS dimensions — 0, 1, log3/log2, 2 — are the dimensions of the **shadow** cast by each sub-solid onto the flat 2D plane, not the dimensions of the solids themselves.

6. Multilinear Polynomials and the Fourier Expansion

In the {-1, 1} domain, any Boolean function $f : \{-1,1\}^n \to \{-1,1\}$ decomposes uniquely as a Fourier expansion over the parity characters $\chi_S(x) = \prod_{i \in S} x_i$, which form a complete orthonormal basis under the uniform inner product $\langle f,g \rangle = 2^{-n} \sum_x f(x)g(x)$:

$$f(x) = \sum_{S \subseteq [n]} \hat{f}(S) \cdot \chi_S(x)$$

The Fourier coefficient $\hat{f}(S) = \langle f, \chi_S \rangle$ quantifies the correlation between f and the parity function χ_S. This orthogonality holds **exclusively in the {-1, 1} basis** because it relies on symmetric cancellation of positive and negative terms. Parseval's Identity states that $\sum_S \hat{f}(S)^2 = 1$ for any $f : \{-1,1\}^n \to \{-1,1\}$. The Fourier coefficients are real numbers in $[-1,+1]$ — they are the first path by which the discrete Boolean tiles generate real-valued limits (Section 7).

The **FKN Theorem** states that if a balanced (or near-balanced) Boolean function has almost all its Fourier weight on degree-1 coefficients, it is structurally close to a dictator (a function depending on a single coordinate) or its negation. The balance condition is essential: without it the conclusion does not hold. The **KKL Theorem** establishes that every balanced Boolean function on n variables has at least one variable with influence $\Omega(\log n / n)$.

7. The Walsh-Hadamard Transform

7.1 The AND Tile Is H_1

The base Hadamard matrix $H_1 = [[1, 1], [1, -1]]$ is not imported from harmonic analysis. It is the AND tile. Under Convention A (True = −1, False = +1), the AND truth table written as a 2×2 matrix with rows indexed by A and columns by B is:

```
       B=+1  B=−1
A=+1 [  +1    +1   ]
A=−1 [  +1    −1   ]
```

The bottom-right entry is −1 because AND(−1, −1) = True = −1. Every other entry is +1 because AND is False unless both inputs are True. This matrix is H_1 exactly. The Hadamard matrix and the AND tile are the same 2×2 object.

This is not an accident. H_1 encodes the unique degree-1 Walsh character that pairs the base Boolean operator with the spectral transform. The Kronecker product structure — H_n = $H_1 \otimes H_1 \otimes \cdots \otimes H_1$ (n times) — is the n-variable generalisation of the 2-variable AND tile. The Walsh-Hadamard Transform is therefore native to the tile alphabet, not imported from outside it.

7.2 Two Readings of the Same Tile

Every 2×2 tile admits two interpretations, producing two different mathematical objects from the same matrix. The connection between them is explicit and derivable.

The point-wise reading (IFS): evaluate the tile rule at specific inputs in $\{-1,+1\}^2$. For a tile with rule matrix T[r][c] ∈ {-1,+1}, the IFS encoding functions are $f_0(p) = 2p$ and $f_1(p) = 2p+1$, which append a 0 or 1 bit to the path address p. At each cell the active/void state is determined by T[r][c]. Iterating this across all cells and all depths produces the fractal attractor — the tile's geometric identity.

The linear reading (WHT): interpret the same 2×2 matrix as acting on a 2-vector of real function values by matrix multiplication. H_1 applied to $(f(-1), f(+1))^T$ computes $(f(-1)+f(+1), f(-1)-f(+1))^T$ — the average and the difference. These are the degree-0 and degree-1 Fourier coefficients of a one-variable function. Iterating via Kronecker product $H_n = H_1 \otimes$ H_{n−1} applies this to all 2^n input combinations simultaneously.

Step-by-step derivation that both readings produce the same object:

Step 1. For AND (= H_1), the rule matrix entries are T = [[+1,+1],[+1,−1]].
Step 2. Point-wise: f(A=+1,B=+1)=+1, f(+1,−1)=+1, f(−1,+1)=+1, f(−1,−1)=−1.
Step 3. Linear: $H_1 \cdot (f(-1), f(+1))^T = [[1,1],[1,-1]] \cdot (v_0, v_1)^T$
 $= (v_0+v_1, v_0-v_1)^T = (\hat{f}(\varnothing), \hat{f}(\{1\}))^T$ — the Fourier coefficients.
Step 4. The IFS iteration f_0, f_1 applied to the path address encodes which
 sub-quadrant contains each cell — the same bit read by H_1 as a parity.
Step 5. In the limit $m \to \infty$: IFS → invariant measure on the fractal boundary;
 $(1/\sqrt{2})H_1$ → unitary operator on $L^2(\{-1,+1\}^\infty)$ = the Walsh-Fourier
transform.

Both limits are the same object — the Walsh-Fourier transform on the Cantor group {-1,+1}^∞ — approached from position space (IFS) and frequency space (WHT). The fractal images and the Fourier analysis are not related topics that happen to share a matrix; they are the same limiting structure, reached by two paths through the same finite-depth tile recursion.

- As a truth table (point-wise): iterate via RBHM quadtree → IFS fractal attractor (geometric identity in position space)
- As a linear operator (matrix multiply): iterate via Kronecker product → Walsh-Hadamard Transform (spectral identity in frequency space)

7.2a Fourier Spectra of All 16 Tiles: A Spectral–Geometric Law

The identification AND = H_1 raises the question: what are the Fourier spectra of the other 15 tiles? Table 2 computes all four Fourier coefficients $\hat{f}(\varnothing)$, $\hat{f}(\{A\})$, $\hat{f}(\{B\})$, $\hat{f}(\{A,B\})$ for every tile under Convention A. Since any two-variable function decomposes uniquely as f(A,B) = f̂(∅) + f̂({A})·A + f̂({B})·B + f̂({A,B})·AB, the four coefficients are the complete spectral fingerprint of each tile.

k	Tile	f(−1, −1)	f(−1, +1)	f(+1, −1)	f(+1, +1)	f̂(∅)	f̂({A})	f̂({B})	f̂({AB})
0	**FALSE**	+1	+1	+1	+1	+1	0	0	0
1	**AND**	−1	+1	+1	+1	+½	+½	+½	−½
2	**A ∧ ¬B**	+1	−1	+1	+1	+½	+½	−½	+½
3	**A**	−1	−1	+1	+1	0	+1	0	0
4	**¬A ∧ B**	+1	+1	−1	+1	+½	−½	+½	+½
5	**B**	−1	+1	−1	+1	0	0	+1	0
6	**XOR**	+1	−1	−1	+1	0	0	0	+1
7	**OR**	−1	−1	−1	+1	−½	+½	+½	+½
8	**NOR**	+1	+1	+1	−1	+½	−½	−½	−½
9	**XNOR**	−1	+1	+1	−1	0	0	0	−1
10	**¬B**	+1	−1	+1	−1	0	0	−1	0
11	**B → A**	−1	−1	+1	−1	−½	+½	−½	−½
12	**¬A**	+1	+1	−1	−1	0	−1	0	0
13	**A → B**	−1	+1	−1	−1	−½	−½	+½	−½
14	**NAND**	+1	−1	−1	−1	−½	−½	−½	+½
15	**TRUE**	−1	−1	−1	−1	−1	0	0	0

Table 2. Fourier spectra of all 16 Boolean tiles under Convention A (True = −1, False = +1). Each row gives the four output values followed by the four Fourier coefficients. Blue rows (k = 3,5,10,12): pure degree-1 tiles (f̂({A}) or f̂({B}) = ±1, all others zero) — Hausdorff dim = 1. Gold rows (k = 6,9): pure degree-2 tiles (f̂({AB}) = ±1, all others zero) — Hausdorff dim = 1 (Cantor diagonal). Parseval: $\hat{f}(\varnothing)^2 + \hat{f}(\{A\})^2 + \hat{f}(\{B\})^2 + \hat{f}(\{AB\})^2 = 1$ for every row.

Spectral–Geometric Law. Table 2 reveals a direct correspondence between Fourier structure and fractal geometry. Tiles with all spectral weight on a single degree-1 coefficient (k = 3, 5, 10, 12: the projections A, B, ¬A, ¬B) have IFS Hausdorff dimension exactly 1 — they generate Cantor strips on rows or columns. Tiles with all weight on the degree-2 coefficient (k = 6, 9: XOR and XNOR) also have dimension 1, but generate Cantor sets on the diagonal and anti-diagonal respectively. The four Sierpiński-family IFS tiles (k = 7, 11, 13, 14: OR, B → A, A → B, NAND) carry weight on three coefficients (degree 0, 1, and 2 simultaneously) and have the unique non-integer dimension log3/log2 ≈ 1.585. The four zero-set Sierpiński tiles (k = 1, 2, 4, 8) share the same spectral profile. Constants (k = 0, 15) carry weight only on degree 0.

The relationship can be stated as a theorem: for two-variable Boolean functions, **concentrated spectral weight on a single Fourier mode is equivalent to rational Hausdorff dimension** (dim = 0 for degree 0, dim = 1 for degree 1 or degree 2 concentrated). Distributed weight across multiple modes corresponds to the irrational dimension log3/log2. This is the spectral-geometric counterpart of the FKN theorem in the two-variable setting: maximally nonlinear Boolean functions (XOR, XNOR) concentrate all Fourier weight on a single high-degree parity function, which simultaneously produces the structurally simplest fractal (a Cantor set) despite having the most complex spectral identity.

The tiles at depth m=1 have entries in {-1,+1}. The Kronecker product $H\_n = H_1$^{⊗n} has entries in {-1,+1} for every finite n. As $n \to \infty$, the normalised matrix $(1/\sqrt{(2^n)})$H_n converges to a continuous operator — the Walsh-Fourier transform on the Cantor group {-1,+1}^∞. Its entries are real numbers in [−1,+1].

The Fourier coefficients $\hat{f}(S) = \langle f,\chi\_S \rangle$ of any Boolean function are real numbers, bounded by Parseval's Identity $\Sigma\_S\ \hat{f}(S)^2 = 1$. They are not truth values. Classical fast WHT on a 2^n-entry vector runs in $O(n \cdot 2^n)$ operations. A quantum processor applies Hadamard gates to all n qubits simultaneously, achieving O(1) circuit depth — but still requires O(n) total gates and $O(2^n)$ measurements to read out all amplitudes. The speedup is exponential in sequential depth, not a collapse to constant total work. This is the first of three converging paths from discrete Boolean vertices to real-valued limits:

- Spectral path: WHT iteration (Kronecker product of H_1) → real Fourier coefficients $\hat{f}(S)$ ∈ [−1,+1]
- Measure-theoretic path: IFS iteration at depth m → uniform measure over live cells → as m → ∞, weak convergence to a continuous probability measure on the fractal attractor (Theorem C2, Section 5.3a)
- Łukasiewicz path: tile operators extended continuously from the Boolean vertices $\{-1,+1\}^n$ to the full hypercube interior $[-1,+1]^n$ — treated in Section 11

All three paths terminate at the same object: the continuous extension of the Boolean tile basis to the interior of [−1,+1]^∞. They are not three separate topics — they are the spectral, geometric, and logical faces of the same limiting structure. The WHT coefficients live in this limit. The IFS invariant measure lives on its fractal boundary. The Łukasiewicz operators fill its interior.

This is why the {-1, 1} basis is not merely convenient: it is the unique basis in which these three convergences happen simultaneously, and in which the AND tile serves as both the Boolean truth table and the harmonic analysis kernel.

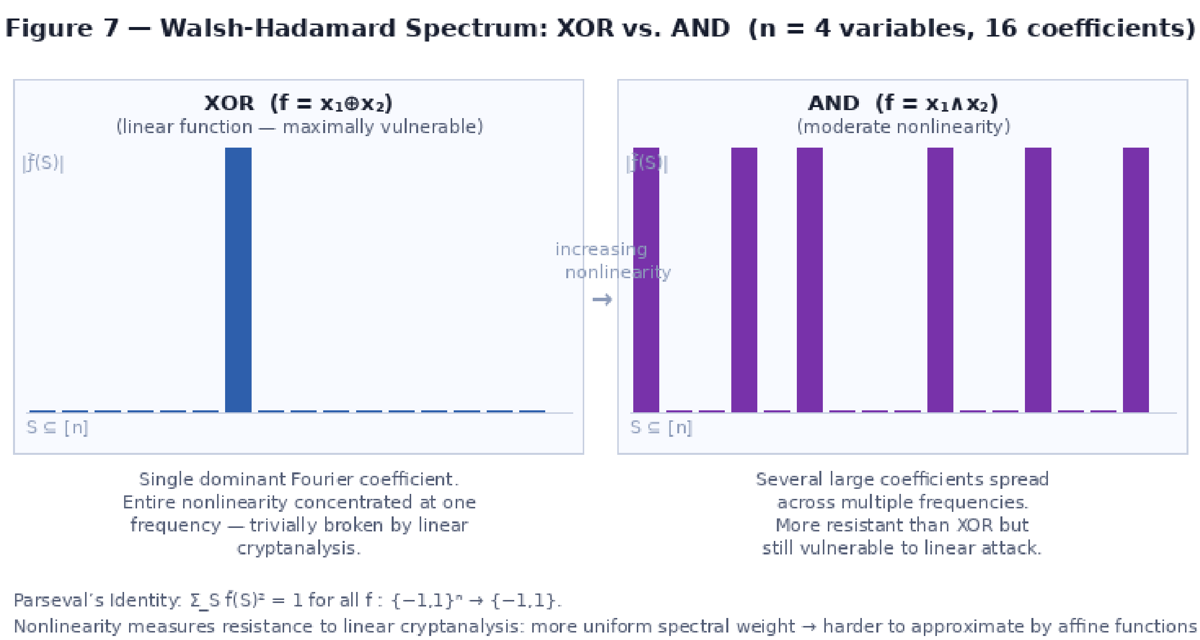

Figure 7. Walsh-Hadamard Spectrum for n = 4 variables. Left: XOR — single dominant Fourier coefficient, the signature of a linear function. Right: AND — several large coefficients spread across multiple frequencies. Parseval's

Identity ($\Sigma \hat{f}(S)^2 = 1$) conserves total spectral energy. Both spectra are produced by applying the AND tile (= H_1) as a linear operator via Kronecker product.

The distribution of spectral energy directly determines resistance to linear cryptanalysis. Nonlinearity $NL(f) = 2^{n-1} - ½ \cdot \max\_S |\hat{f}(S)|$ quantifies this: functions whose maximum Walsh coefficient is smaller are farther from any affine approximation. XOR's spectrum — one coefficient carrying all the weight — makes it maximally linear. More uniform spectral distributions correspond to higher nonlinearity, which is why the Walsh-Hadamard Transform is the primary tool in S-box design for block ciphers.

8. The Tile Framework: Structure and Completeness

8.1 Uniform-Label and Level-Varying Regimes

The RBHM quadtree operates in two distinct regimes. In the **uniform-label** regime every node carries the same tile, producing the self-similar fractals in Figures 1–6. In the **level-varying** regime depth level d carries tile f_d, giving 16^m distinct surfaces at depth m. The level-varying regime is the mechanism for encoding arbitrary Boolean surfaces, and is the subject of an open injectivity question (Section 19).

8.2 The Exact Counting Identity

A common claim in early formulations was that 16^m equals the full number of 2^m-variable Boolean functions. A concrete example shows why this requires care.

At **m=1**: the level-varying construction allows 16 possible tiles at depth 1, and there are exactly 16 distinct two-variable Boolean functions. The construction is **complete** — every 2-variable Boolean function is represented by exactly one tile. The 16 tiles and the 16 functions are the same set.

At **m=2**: the level-varying construction allows 16×16 = 256 distinct sequences of two tiles (one per depth level). A depth-2 surface is a 4×4 binary grid. The number of distinct 4×4 binary grids is 2^16 = 65,536. So the 256 level-varying sequences address only 256 of the 65,536 possible 4×4 surfaces. The construction covers less than 0.4% of the full space at depth 2, and the fraction vanishes as m grows.

```
16ᵐ  =  2^(4m)    ←  level-varying surfaces at depth m
 2^(4ᵐ)             ←  total distinct depth-m surfaces
```

Figure 13 — The Exact Counting Identity: Level-Varying Subspace vs Full Function Space

m	16^m (level-varying)	2^(4^m) (all functions)	Ratio
1	2^4 = 16	2^4 = 16	1/1 (complete)
2	2^8 = 256	2^16 = 65,536	1/256 ← structured subspace
3	2^12 = 4,096	2^64 ≈ 1.8×10^{19}	$1/4.5 \times 10^{15}$ ← structured subspace
4	2^16 = 65,536	2^256 ≈ 1.2×10^{77}	negligible ← structured subspace

What the identity says

16^m = (2^4)^m = 2^(4m)

This is distinct from:
2^(4^m) (the full space)

At m=1: equal (construction is complete at depth 1)

At m=2:
Level-varying: 2^8 = 256
Full space: 2^16 = 65,536

Level-varying addresses a structured subspace.
NOR pipeline reaches all surfaces (fractal depth ∞).

The space is inexhaustible, not incomplete.

Figure 13. The exact counting identity. At m=1 the level-varying construction is complete (16 = 16). For m ≥ 2 it addresses a structured subspace: at m=2, 256 out of 65,536 possible 4×4 binary surfaces. The NOR pipeline covers the full space; level-varying encoding alone does not.

This is **fractal inexhaustibility**: at any finite depth there is always more structure deeper in the fractal. The level-varying construction captures a structured, geometrically coherent slice of the full space at each depth — a slice whose internal organisation is determined by the tile composition rules. The NOR pipeline (Section 8.4) reaches every surface at sufficient depth. The open problem in Section 19 addresses whether distinct level-varying encodings always produce distinct surfaces within the structured subspace.

8.3 A Note on Thermodynamic Isomorphisms

The {-1, 1} encoding aligns naturally with statistical physics. In the Ising model of ferromagnetism, spin variables σ_i ∈ {-1,+1} interact via the Hamiltonian H(σ) = −Σ J_{ij}σ_iσ_j − hΣσ_i, which is the thermodynamic twin of the degree-2 Fourier expansion of a Boolean function (Cipra, 1987). Finding the Ising ground state is equivalent to solving MAX-SAT, and the bijection σ_i = 1 − 2x_i converts the Hamiltonian into a QUBO problem, connecting Ising machines to Boolean satisfiability (Lucas, 2014). This isomorphism has driven the development of dedicated Ising machines and neuromorphic accelerators. It is noted here as a consequence of the {-1, 1} symmetry; the logical framework in this paper does not depend on it.

8.4 NOR Universality

Theorem 8.4 (NOR Universality). For any n ≥ 1 and any n-dimensional hypercubic Boolean surface S, there exist independent n-dimensional NOR fractals at sufficient depth whose downsampled output exactly reproduces S.

Proof sketch: any Boolean function is expressible as a finite NOR circuit (functional completeness). The n-dimensional quadtree depth corresponds directly to NOR circuit depth. □

Clarification on terminology. The “NOR pipeline” uses NOR as its *universal primitive*, not as its sole tile label. At each level of the quadtree, the pipeline may select any of the 16 tiles — including P, Q, NOT, AND, and OR — because all of these are derivable from NOR by the functional completeness argument. The analogy is a chip described as “built from NAND gates”: the claim is about the universal primitive, not about every physical gate being a bare NAND. Consequently, the observation that the uniform NOR IFS (using only the NOR tile at every level) collapses to a single point attractor at the top-left origin does not refute the theorem. That point attractor is the image of the *uniform* NOR fractal. The pipeline uses *level-varying* sequences whose level labels encode a NOR circuit for the target function. The NOR circuit for an arbitrary n-variable Boolean function uses NOR gates at circuit nodes and passes variables through identity tiles (P, Q) at input nodes, all within the level-varying tile sequence.

The constructive mechanism — the downsampling pipeline, the n-dimensional generalisation, and the full bit budget formalism — is developed in detail in the companion geometry paper (St. Denis, in preparation). For the logical purposes of this paper, NOR universality is the key fact: the fractal space is inexhaustible but complete.

9. Dimensional Folding: The Fixed Point of Quantification

The quantifier hierarchy from Section 12 raises a natural question: after you have applied quantifiers repeatedly — collapsing variables, then tile-label axes, then bit-planes — where do you land? The dimensional folding theorem answers this: you always land in a **3-dimensional logical space**. The three parameters n (variable count), m (recursion depth), and W (value range) are exactly the three degrees of freedom of logical space, and any finite logical surface at any depth is isomorphic to a 3D surface. The folding is not a geometric convenience; it is the fixed point of the quantification process.

9.1 The Folding Bijection

Theorem 9.1 (Dimensional Folding Bijection). Let S be an n-dimensional hypercubic Boolean surface at resolution depth m with bit-depth W, and total bit budget $B = n \cdot m + W$. Let $m'' = \lfloor B/3 \rfloor$ and $W'' = B \bmod 3$. Define the map $\varphi : \mathrm{Cells}(S) \to \mathrm{Cells}(S'')$ by: for a cell with quadtree path address $a \in \{0, \ldots, 2^B - 1\}$, φ(a) is the cell in the 3D surface S″ whose path address is the same B-bit integer a, read as three interleaved streams of m″ bits each. Then φ is a bijection, and the value at cell a in S equals the value at φ(a) in S″.

Proof. Bijectivity: φ maps each B-bit address to a unique 3D path address (same B-bit integer, different axis interpretation). Since the 3D address space has 2^B cells and the map is injective by construction, it is bijective.

Note on terminology. φ is a bijection of discrete address spaces, not a topological isomorphism (homeomorphism). A homeomorphism requires neighbourhood preservation: two cells adjacent in S must be adjacent in S″. This fails across quadrant boundaries, as the Morton Z-order addressing that governs the path code is known to place spatially adjacent cells at distant index positions when they cross major quadrant boundaries (resolved negatively, Section 19). The folding therefore *preserves values exactly* and *preserves global combinatorial structure exactly*, but does not preserve local spatial adjacency. It is a **data bijection**, not a geometric isomorphism.

Value preservation: the value at each cell is determined by the Boolean function applied to the logical inputs at that address. The B-bit address encodes the same sequence of sub-quadrant choices in both the n-dimensional and 3D interpretations — only the grouping of bits into axes changes. The quadtree path from root to cell determines the logical inputs at each level; re-grouping the path bits across three axes rather than n does not change which leaf is reached, only which axis decomposition labels it. Values are therefore identical at φ-corresponding cells.

Commutativity with tile application: applying a tile T to cell a in the n-dimensional surface produces value $T(a_0, a_1, \ldots, a_{n-1})$ where a_i are the axis coordinates. The same value arises at φ(a) in S″

because the axis coordinates in 3D are derived by the same bit-selection from the path address; the tile value depends only on the cell's position in the truth table, not on how the address is partitioned into axes. □

The structure produced by reading two interleaved bit streams as a 2D address is known in the literature as a Morton code or Z-order curve (Morton, 1966). Within the RBHM framework, Morton encoding is not an import — it is the quadtree's native address, seen from the outside.

9.2 The Same Object in Three Dimensions

Figure 11 shows the same NOR fractal at bit budget B = 6 expressed in two representations: a 2D 8×8 grid (n = 2, m = 3) and a 3D 4×4×4 cube (n = 3, m = 2). Both encode the same 64 binary values. The address redistribution is the native quadtree path code re-read across different numbers of axes: the same 6-bit string, partitioned 2+2+2 for 3D rather than 3+3 for 2D.

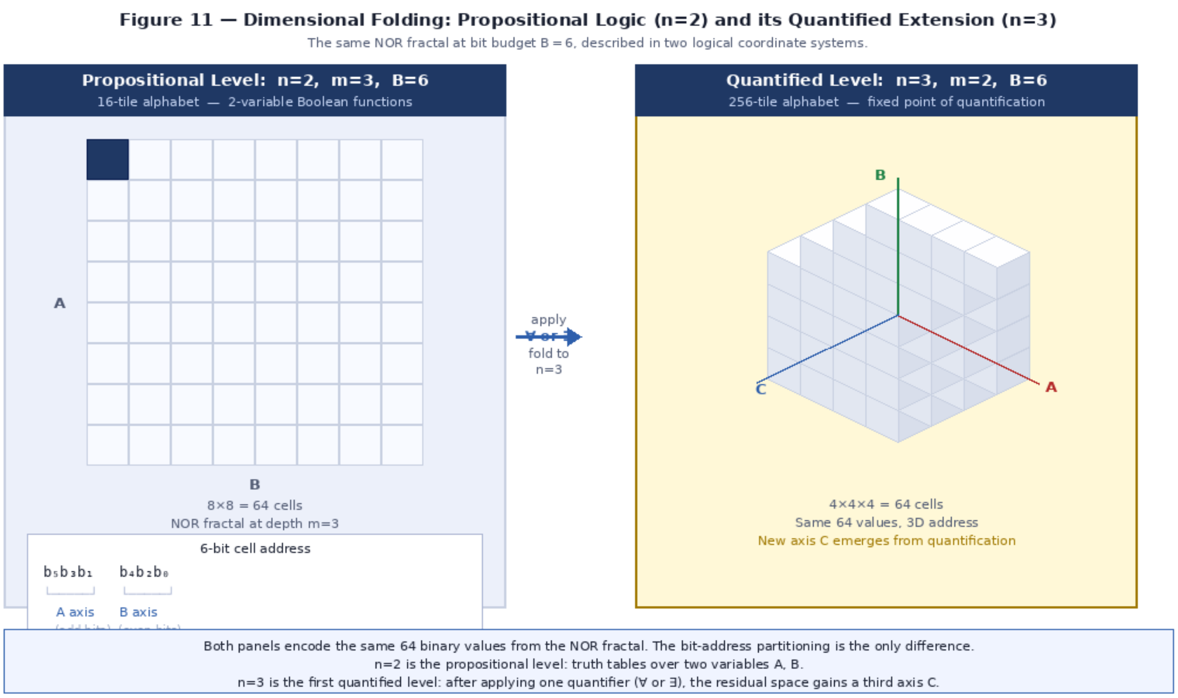

Figure 11. The Dimensional Folding Isomorphism at B = 6. Left: 2D grid (n=2, m=3), 8×8 = 64 cells. Right: 3D cube (n=3, m=2), 4×4×4 = 64 cells. The bit-address boxes show how the same 6-bit quadtree path code is partitioned across dimensions. All cells are the NOR fractal in different coordinate systems — the folding reinterprets the native path address, not an external curve.

The folding is not a lossy projection or an approximation. It is a *change of coordinates* on the same discrete structure, exactly as the {-1, 1} basis change is a change of coordinates on the same Boolean algebra. In both cases the underlying object is unchanged; only the language used to describe it shifts.

9.2a What the Folding Preserves — and What It Does Not

This answers a natural question precisely: after folding an n-dimensional surface to 3D, what is retained and what is not?

What is fully preserved:

- The value of every cell. The bijection is on cell addresses; each cell's output value (its entry in the Boolean function's truth table) is unchanged. The 3D representation contains exactly the same information as the n-dimensional one.
- The total count of live and void cells, and therefore all aggregate statistics of the Boolean function (e.g. its balance, its Fourier DC coefficient $\hat{f}(\varnothing)$).
- The fractal attractor of the underlying IFS. If the n-dimensional surface was generated by a uniform-label quadtree under tile T, the 3D interpretation is the same IFS attractor viewed in a different coordinate system.

What may not be preserved:

- Spatial adjacency across quadrant boundaries. Two cells that are neighbours in the n-dimensional surface may map to distant locations in the 3D representation under the native quadtree path code. Cells within the same sub-quadrant at any depth remain adjacent; cells across quadrant boundaries may not. This is the locality question addressed in Section 19.
- The logical structure of implications between cells. If cell A logically implies cell B in the original n-dimensional surface (i.e. whenever A is live, B is live), this implication is preserved as a truth-table relationship but may not be spatially local in the 3D representation.
- The precise Hausdorff dimension in the folded embedding. The XOR attractor has dimension $\log 3/\log 2 \approx 1.585$ as a subset of 2D space. When the same attractor is expressed in 3D coordinates via folding, the dimension of the embedded object depends on the folding — it is Theorem 5.2's result (dimension exactly 2 for the 3D XOR IFS) rather than the 2D Sierpiński dimension.

In short: the folding is a **lossless value bijection** but not a **locality-preserving embedding**. The distinction matters for any algorithm that depends on spatial proximity (e.g. downsampling, neighbourhood operations, topological analysis). The locality question asks whether an alternative address remapping — such as a Hilbert curve — could provide bounded locality guarantees across quadrant boundaries, independently of the source dimension n.

9.3 Why 3D is the Natural Reduction Target

Three independent reasons converge on 3D as the privileged reduction target:

- Topological privilege. 3D is the minimum dimension with enclosed volume. In 1D there are only points and intervals; in 2D there is surface but no interior; in 3D enclosed space first appears. Knot theory is non-trivial in 3D (knots cannot be untied) but trivial in 4D (every knot untangles). The Betti numbers H_0, H_1, H_2 are all simultaneously non-trivial only in 3D and above.
- The XOR integer boundary. The n-variable XOR IFS has Hausdorff dimension $n-1$ (Theorem 5.2). At $n=3$ this is exactly 2 — the attractor becomes a complete 2D surface filling all voids of the 2D Sierpiński gasket. This is a pure mathematical fact: $n=3$ is the first dimension at which the XOR fractal fills rather than fragments.
- The RGB/colour connection. Digital colour is represented as three independent one-byte channels (R, G, B). This is precisely three independent 3D bit-planes at $W = 1$ — exactly the $n = 3$, $W = 1$ allocation of the tile framework. The folding from n-D to 3D is the abstract generalisation of how any volumetric dataset can be stored as a colour volume.

- The XOR dimension coincidence. The n-variable XOR IFS has Hausdorff dimension n−1 for n≥3, always an integer (Theorem 5.2). For n=2 the attractor is the Sierpiński gasket with the distinctive non-integer dimension log3/log2 ≈1.585 — the unique exception arising because the Sierpiński IFS has N=3 contractions rather than 2^(n−1)=2. At n=3, dim=2 exactly: the attractor becomes a complete 2D surface whose axial projections fill the full square. This is the first dimension where the XOR fractal fills rather than fragments, and a structural reason why 3D is the natural folding target.

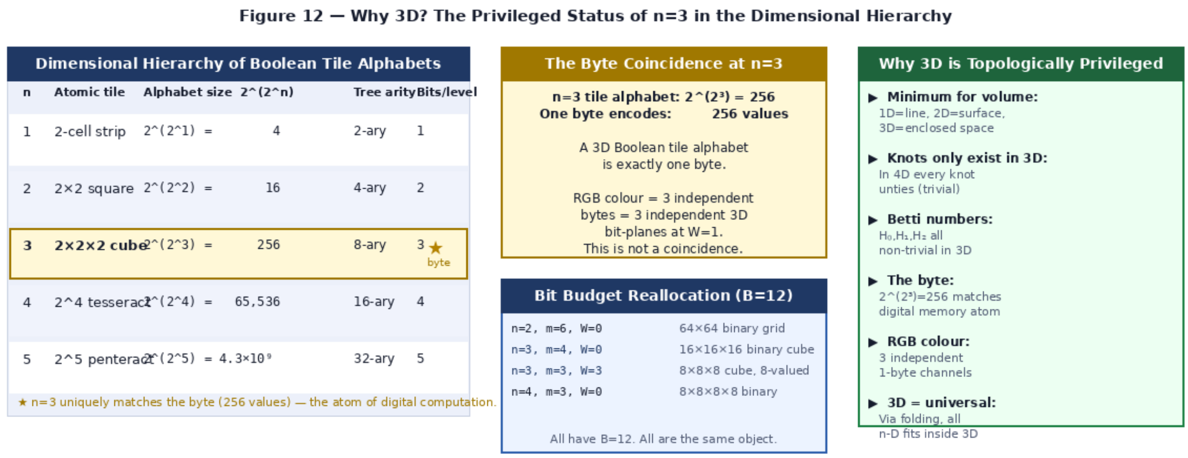

Figure 12. Why 3D is privileged. Left: the dimensional hierarchy showing that n=3 uniquely matches the byte (256 tile functions). Centre: the byte coincidence and bit-budget reallocation examples. Right: the six independent reasons why 3D is the natural reduction target for the dimensional folding isomorphism.

9.4 The Folding Connection to Łukasiewicz Chaos

The dimensional folding is structurally identical to the Łukasiewicz tent map 2|x| − 1 from Section 11, scaled to higher dimension. The tent map folds a 1D interval onto itself at each iteration. The IFS quadtree folds a 2D region into four sub-regions at each level. The dimensional folding folds an n-dimensional address space into a 3D address space by redistributing axis bits.

All three are instances of the same operation: **recursive folding of a bounded domain** — preserving all information content while changing how the dimensions are named. The chaos in p ↔ ¬p is the 1D case; the Sierpiński triangle is the 2D case; the dimensional folding isomorphism is the n-D generalisation. They share a common mathematical skeleton.

9.5 A Note on Locality

The native quadtree path code preserves locality within each quadtree quadrant but can place adjacent cells far apart across quadrant boundaries. Whether an alternative address remapping — such as the Hilbert space-filling curve, which minimises the maximum locality distortion — could provide a bounded locality guarantee independently of n remains an open question (resolved negatively — see Section 19).

10. When to Use Each Basis: The Operational Divergence

The {-1, 1} basis is mathematically superior for spectral analysis; the {0, 1} basis is superior for combinatorial enumeration. This section answers the question precisely: when does the spectral advantage of {-1, 1} become a practical liability? The answer centres on one operation: **counting**. Any algorithm that fundamentally depends on counting occurrences — neighbours, set members, density — belongs in {0, 1}. Any algorithm that fundamentally depends on correlation, orthogonality, or linear decomposition belongs in {-1, 1}.

10.1 Axiomatic Simplicity and the Boolean Ring

In the {0, 1} basis, the fundamental laws of Boolean algebra reduce to intuitive visual arithmetic. The Annulment Law ($A \wedge 0 = 0$) is immediately obvious: a variable ANDed with False is always False. In the {-1, 1} polynomial basis the equivalent statement requires expanding the full polynomial for AND, cross-multiplying with the constant polynomial for FALSE ($f = 1$), and relying on algebraic coefficient cancellation to arrive at the same conclusion.

This is mathematically elegant in the context of a Fourier analysis, but is *unnecessary* cognitive and computational overhead for determining whether a logic gate is open or closed. The {0, 1} Boolean ring is the right tool for axiomatic simplification; the {-1, 1} polynomial ring is the right tool for spectral and continuous analysis.

10.2 Combinatorial Density: The Game of Life Problem

The most pronounced divergence occurs when a system depends on geometric density or neighbour-counting — perfectly illustrated by Conway's Game of Life. The rule set requires counting exactly how many of the 8 neighbours of each cell are alive. Table 3 shows the computational consequence of each basis choice.

Basis	Neighbourhood Sum Formula	Consequence
{0,1} Basis	`N = Σ Cᵢ  (direct sum)`	N=3 means exactly 3 live neighbours. Trivial, O(n) arithmetic.
{−1,1} Basis	`N = Σ Cᵢ  (with −1=live)`	N=0 could be 4 live + 4 dead. Cancellation makes counting ambiguous.
Fix in {−1,1}	`N = (8 − Σ Cᵢ)/2`	Correct but adds cognitive overhead and two extra operations per cell.

Table 3. Neighbourhood counting in Conway's Game of Life under each basis. The {0,1} sum is direct and unambiguous; the {−1,1} sum is ambiguous without correction.

In the {0, 1} basis, $N = \Sigma C_i$ is a direct summation. A sum of 3 means exactly 3 live cells, unambiguously. In the {-1, 1} basis (where $-1 = \text{live}$), a sum of 0 could represent 4 live and 4 dead cells — the symmetric cancellation that makes {-1, 1} so powerful for Fourier analysis is precisely what makes it catastrophic for population counting. The correction exists but introduces overhead on every cell of every generation. For any algorithm fundamentally dependent on neighbour enumeration, the {0, 1} basis is the correct choice.

10.3 Algorithmic Sparsity and Matrix Optimisation

In massive datasets and quadtrees, the {0, 1} basis enables sparse data structures: a value of 0 (False) terminates recursive processing and compresses trivially. Standard sparse matrix formats (CSR, COO) exploit this to store only non-zero entries.

In the {-1, 1} space, False is encoded as +1 — a structurally active, non-zero integer. There is no numerical zero to terminate recursion or collapse a sparse array. Every false state must be stored, calculated, and routed through the processor. Sparse matrix compression fails entirely. To bridge advanced physics frameworks back to computational efficiency, optimisation analysts use the variable transformation σ_i = (1 − σ_i) / 2, converting active Ising states back into {0, 1} QUBO form and restoring algorithmic sparsity.

10.4 The Resolution

▶️ **{-1, 1} is the domain of ANALYSIS**
Use for: spectral decomposition, Fourier coefficients, Ising energy, orthogonality proofs, cryptographic nonlinearity, EBM training.

▶️ **{0, 1} is the domain of EXECUTION**
Use for: neighbour-counting, sparse matrix storage, circuit truth tables, combinatorial enumeration, hardware implementation.

The two domains are perfect topological complements. Mastery of discrete logic requires knowing exactly when the problem demands each.

Neither basis supersedes the other. The {-1, 1} representation is the mathematically natural home for spectral decomposition, orthogonality, thermodynamic modelling, and fractal geometry — all of which this paper demonstrates. The {0, 1} representation is the natural home for combinatorial enumeration, sparse computation, and circuit-level logic. Recognising which domain the problem belongs to is the essential skill.

11. Łukasiewicz Fuzzy Logic and the {-1, 1} Basis

Section 7 identified three converging paths from the discrete Boolean vertices $\{-1,+1\}^n$ to real-valued limits: the spectral path (Fourier coefficients via WHT), the measure-theoretic path (IFS invariant measure via weak convergence), and the Łukasiewicz path (tile operators extended continuously to the hypercube interior). The first two have been developed. This section develops the third, and shows that all three terminate at the same object.

The Boolean tile framework operates on the *vertices* of the n-dimensional hypercube — the 2^n corners of $\{-1, 1\}^n$ where truth values are crisp. Łukasiewicz infinite-valued logic extends this to the *interior* of the hypercube, assigning real-valued truth degrees throughout $[-1, 1]^n$. Rephrasing Łukasiewicz's system in the {-1, 1} basis reveals connections invisible in its standard [0, 1] formulation. The same tile operations that generate fractal attractors at the vertices, when extended continuously to the interior, produce the saturating arithmetic of neural network activations, the chaotic dynamics of self-reference, and the probability measures of Theorem C2 (Section 5.3a).

The three paths are one structure seen from three angles. The WHT coefficients (spectral path) live in the limit space $[-1,+1]^\infty$. The IFS invariant measure (measure-theoretic path) is supported on the fractal boundary of the same space. The Łukasiewicz operators (logical path) fill its interior. Every tile simultaneously determines all three: which fractal it generates (IFS), which spectrum it has (WHT), and how it interpolates into the interior (Łukasiewicz).

11.1 The Łukasiewicz Operators in {-1, 1}

The standard Łukasiewicz system is defined over truth values $p, q \in [0, 1]$. Applying the bijection $x = 1 - 2p$ (Convention A: True → −1, False → +1) converts every operator into the {-1, 1} domain. The results are structurally cleaner than their {0, 1} originals:

- Negation: $\neg p = 1-p$ → $-x$ (pure sign flip; reflection through the origin)
- Strong disj.: $\min(1, p+q)$ → $\max(-1, x+y-1)$ (clamped sum with offset −1)
- Strong conj.: $\max(0, p+q-1)$ → $\min(+1, x+y+1)$ (clamped sum with offset +1)
- Implication: $\min(1, 1-p+q)$ → $\min(+1, 1-x+y)$
- Equivalence: $1-|p-q|$ → $|x-y| - 1$ (distance-based; always ≤0)

The negation result is immediate but significant: the {0, 1} negation $x \mapsto 1 - x$ is an affine map (translation plus reflection). The {-1, 1} negation $x \mapsto -x$ is a purely *linear* map — geometric reflection through the origin, with no translation term. This makes {-1, 1} the natural home for any logic where negation is meant to represent exact duality: the negation of a proposition is its mirror image, not its affine shift.

The two main connectives form a symmetric pair: strong disjunction and strong conjunction differ only in sign of offset (−1 vs +1) and direction of clamping (max vs min). They are related by negation exactly as De Morgan's laws require, and this De Morgan duality is *visible* in the {-1, 1} formulas in a way that is obscured in {0, 1}. Furthermore, clamped sums of the form $\max(-1, x+y-1)$ are exactly the **saturating addition** used in neural network activations and quantised arithmetic — the Łukasiewicz operators in {-1, 1} are the natural continuous relaxation of Boolean gates into activation functions.

A critical contrast with Boolean {-1, 1}: in classical Boolean logic, XNOR = $-xy$ and XOR = xy (the product, a bilinear extension). In Łukasiewicz logic, equivalence = $|x - y| - 1$ (a metric, an absolute-value extension). Both formulas agree on the four vertices $\{-1, 1\}^2$ but extend differently into the continuous interior — the Boolean extension is multiplicative and smooth; the fuzzy extension is distance-based and piecewise. Figure 9 shows both systems as heatmaps over the full $[-1, 1]^2$ square.

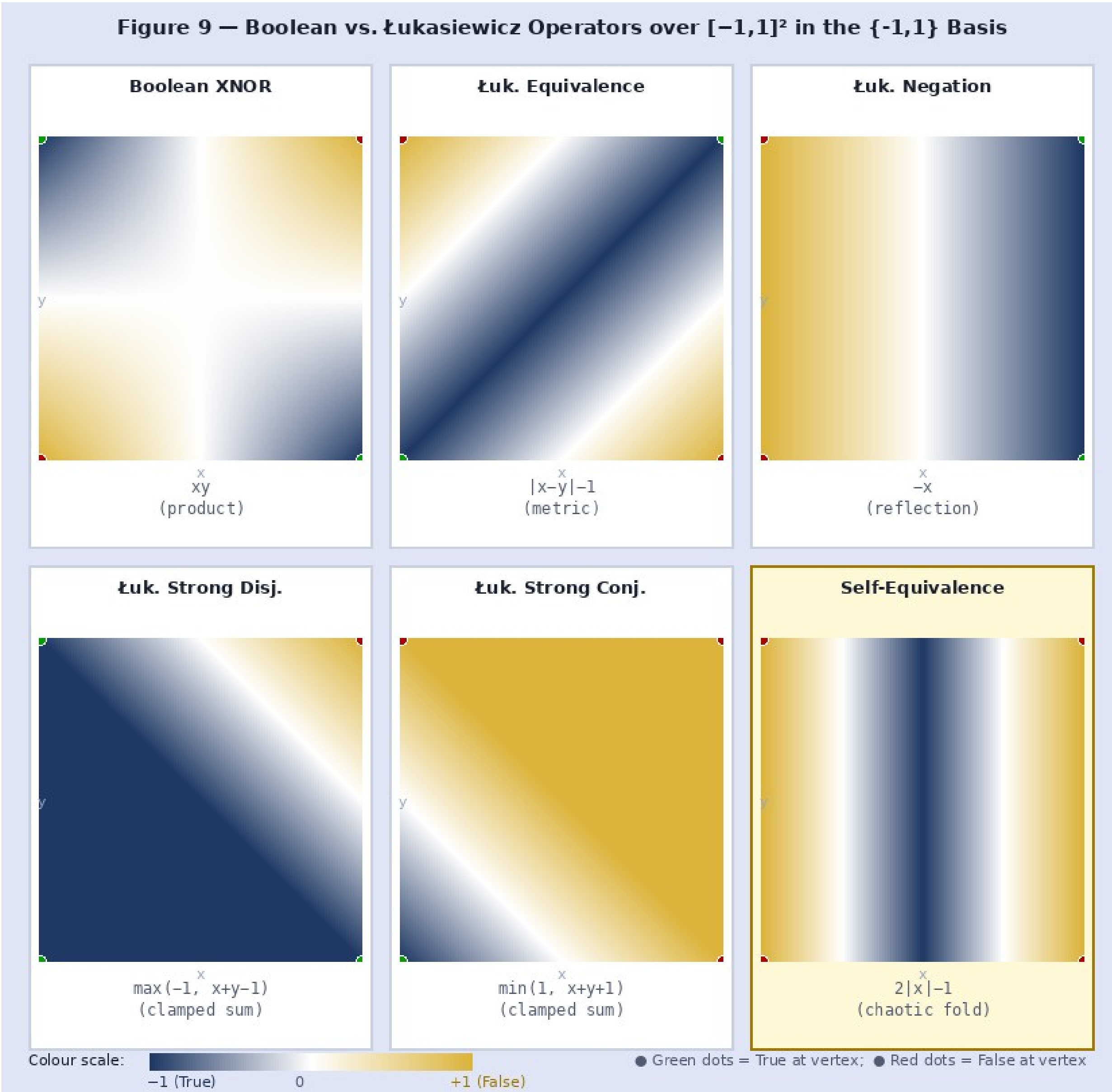

Figure 9. Boolean vs. Łukasiewicz operators as heatmaps over [−1, 1]². Colour: navy = −1 (True), white = 0, gold = +1 (False). Green/red dots mark Boolean vertex values. Boolean XNOR (xy, bilinear) and Łukasiewicz equivalence (|x−y|−1, metric) agree on vertices but interpolate differently through the interior. Self-equivalence (bottom-right, highlighted) generates the chaotic tent map.

11.2 The Chaotic Liar: Semantic Self-Reference as a Dynamical System

The most natural generalisation of the Liar paradox into infinite-valued logic is the sentence that asserts of itself not that it is simply false, but that it is true to the extent that it is false: "this sentence is as true as it is false." Mar and St. Denis (1999) called this the **Chaotic Liar** and showed that its semantic evaluation algorithm is mathematically identical to the classical tent map. In the Boolean Tile Basis, the same object arises as $p \leftrightarrow \neg p$ — a proposition equated with its own negation. In {0, 1}: $1 - |p - (1 - p)| = 1 - |2p - 1|$, the classical tent map. In {-1, 1}, substituting $x = 1 - 2p$ and converting:

$$f(x) \ = \ 2|x| - 1$$

This is an absolute-value fold, symmetric about the origin. The map sends both −1 and +1 to +1 (False), and has a unique minimum of −1 (True) at $x = 0$ — the logical midpoint of maximal indeterminacy. The {0, 1} tent map $1 - |2p - 1|$ and the {-1, 1} map $2|x| - 1$ are negations of each other under the bijection, mapping the same underlying phenomenon.

Both maps are **maximally chaotic among piecewise linear maps on their domain**, with Lyapunov exponent $\lambda = \log 2$.

Proof (Lyapunov exponent). The Lyapunov exponent $\lambda = \lim_{n \to \infty} (1/n) \Sigma_{i=0}^{n-1} \ln|f'(x_i)|$ measures the average rate of trajectory divergence. For $f(x) = 2|x| - 1$: for $x > 0$, $f'(x) = 2$; for $x < 0$, $f'(x) = -2$. In both cases $|f'(x)| = 2$, constant everywhere the derivative exists. Therefore $\lambda = \lim_{n \to \infty} (1/n) \Sigma_{i=0}^{n-1} \ln 2 = \ln 2 = \log 2$. Since $\lambda > 0$ the map is provably chaotic by the standard criterion for positive Lyapunov exponent. □

The fixed points of $2|x| - 1$ satisfy $f(x) = x$. For $x \geq 0$: $2x - 1 = x \Rightarrow x = 1$ (the False boundary). For $x < 0$: $-2x - 1 = x \Rightarrow x = -1/3$. Both fixed points are unstable since $|f'| = 2 > 1$ at both points: any small perturbation sends the orbit diverging. This is the formal statement that $p \leftrightarrow \neg p$ has no stable truth value in Łukasiewicz logic.

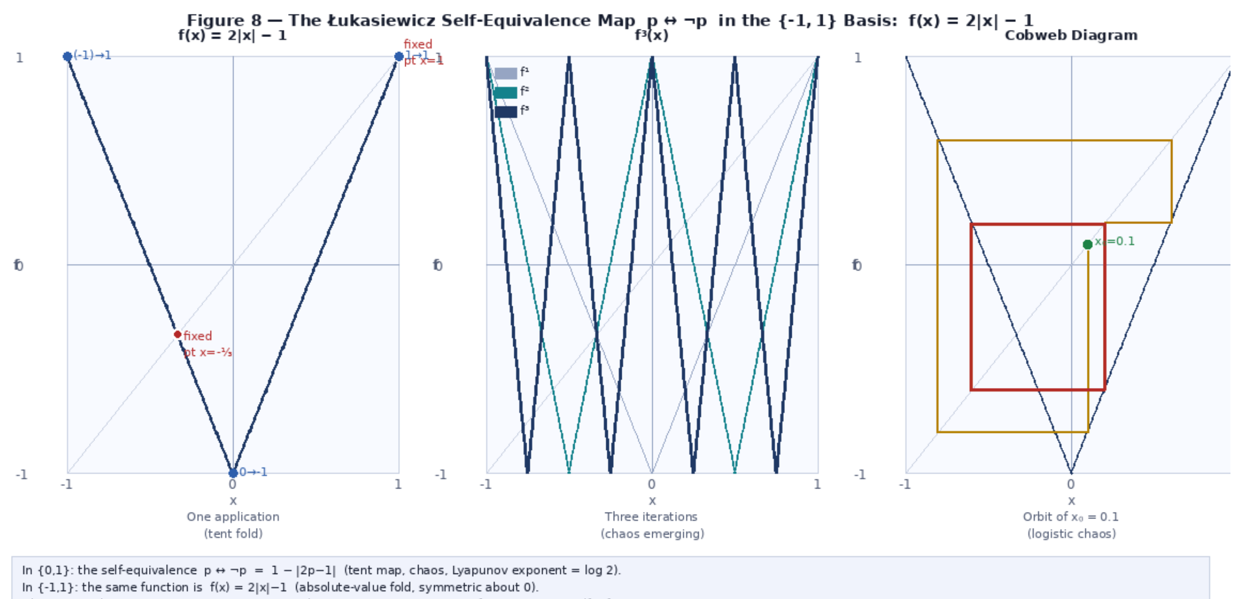

Figure 8. The Łukasiewicz self-equivalence map $p \leftrightarrow \neg p$ in {-1,1}: $f(x) = 2|x| - 1$. Left: single application — an absolute-value fold with fixed points at $x=1$ (False) and $x=-1/3$ (both unstable). Centre: three iterations showing the rapid proliferation of oscillations characteristic of chaotic dynamics. Right: cobweb diagram of the orbit starting at $x_0=0.1$, showing the chaotic trajectory that never converges. Lyapunov exponent $\lambda = \log 2$.

11.3 The IFS Connection: Folding as Fractal Generation

The map $f(x) = 2|x| - 1$ is a **folding map** — it collapses both [−1, 0] and [0, 1] onto [−1, 1] by reflecting the left half over the origin. This operation is structurally identical to the first step of the IFS recursive tile subdivision: the quadtree splits a region into four sub-regions by folding along two axes. The chaos in $p \leftrightarrow \neg p$ is the continuous one-dimensional analogue of the fractal generation the paper describes discretely in the tile system. Both arise from the same mechanism: recursive folding of a bounded domain.

This extends the Sierpiński connection (Section 5) from a single discrete operator (XOR) to the entire continuous Łukasiewicz system. The XOR tile generates the Sierpiński triangle by recursive two-dimensional folding; the Łukasiewicz self-equivalence generates the tent map by recursive one-dimensional folding. Both are fractal attractors of a Boolean folding rule, one discrete and spatial, one continuous and temporal.

11.4 Spectral Continuity: Walsh Analysis Extended

The Walsh-Hadamard characters χ_S(x) = ∏_{i ∈ S} x_i are defined on $\{-1, 1\}^n$. Their restriction to the Boolean vertices gives the Fourier expansion of Section 6. Their *analytic continuation* to $[-1, 1]^n$ provides a natural spectral decomposition of Łukasiewicz's continuous truth domain using the same basis functions. The {-1, 1} Fourier machinery does not stop at the hypercube corners — it extends smoothly into the fuzzy interior, with the Łukasiewicz operators providing the continuous connectives that govern how truth values combine within that interior.

Concretely: the Łukasiewicz strong disjunction max(−1, x+y − 1) is a truncated bilinear function. Its Fourier decomposition over $[-1, 1]^2$ has a dominant linear term (degree-1 characters χ_{1}, χ_{2}) and a small bilinear correction (degree-2 character χ_{12}), exactly where the Boolean AND polynomial (1+x+y−xy)/2 concentrates its weight. The two logics share the same spectral architecture at low degree; they diverge at high degree where the clamping nonlinearity introduces higher harmonics.

11.5 The Three-Domain Architecture

The {-1, 1} basis sits at the centre of three parallel extensions of the same underlying structure, each relaxing a different constraint on the domain of truth values:

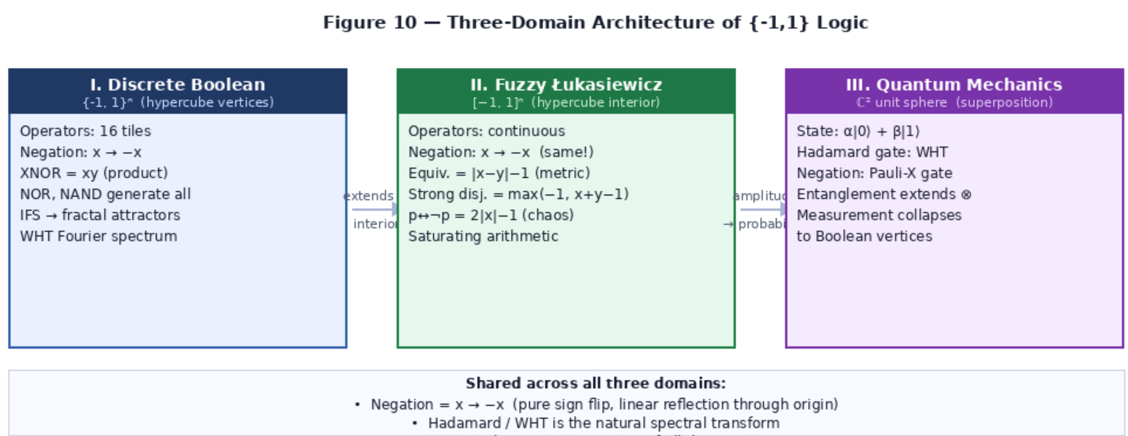

Figure 10. Three-Domain Architecture. Discrete Boolean logic ($\{-1,1\}^n$, hypercube vertices) is extended to Fuzzy Łukasiewicz logic ($[-1,1]^n$, hypercube interior) and further to Quantum mechanics ($\mathbb{C}^2$ unit sphere, superposition). Each domain extends the previous. Negation x → −x and the Hadamard/WHT transform are shared across all three.

The critical shared structure is negation: in all three domains, negation is **x ↦ −x** — a linear reflection through the origin. In {0, 1} Boolean logic, negation is x ↦ 1 − x, an affine map. Moving to {-1, 1} converts this from an affine to a purely linear operation, making explicit a symmetry that was always present but geometrically hidden. Łukasiewicz logic inherits this clean negation throughout the continuous interior. Quantum mechanics shares the same negation in the form of the Pauli-X gate, and the Hadamard gate is the WHT applied to quantum amplitudes. These are genuine structural correspondences across all three domains.

Important qualification on Domain III. The connection to quantum mechanics is structural rather than operational. The Walsh-Hadamard Transform and the quantum Hadamard gate are the same mathematical object, and negation $x \mapsto -x$ is common to all three domains. However, the Łukasiewicz operators — with their saturating arithmetic (max, min, absolute value) — are *not unitary*. Quantum mechanics requires all operators to be unitary transformations (energy-preserving). The map $\max(-1, p+q-1)$ does not preserve norm and cannot be implemented as a quantum gate. The three-domain architecture is therefore a diagram of *shared algebraic structure and notation*, not a claim that Łukasiewicz logic is a physical bridge to quantum superposition. Domain I and Domain II are mathematically rigorous extensions of the same {-1, 1} basis. Domain III shares the same spectral transform and the same negation; the remainder of the quantum-mechanical formalism requires unitary operators that the tile framework does not in general provide.

11.6 Continuous Non-Composability

A precise and testable consequence of the continuous extension is that syntactic composition of operators is no longer semantically transparent when inputs leave the discrete boundary. In discrete Boolean logic, any connective can be built from others without semantic cost: the composed circuit and the native operator evaluate identically at every vertex of $\{-1,+1\}^2$. In the continuous interior, this identity breaks down.

Proposition 11.1 (Continuous Non-Composability). Let $\mathrm{XNOR}_n(p, q) = -pq$ be the native multilinear polynomial for XNOR under Convention A. Let $\mathrm{XNOR}_o(p, q)$ denote the syntactic composition $(\neg P \vee Q) \wedge (\neg Q \vee P)$, evaluating each connective by its {-1,+1} multilinear polynomial ($\mathrm{NOT}(x) = -x$, $\mathrm{OR}(x,y) = (-1+x+y+xy)/2$, $\mathrm{AND}(x,y) = (1+x+y-xy)/2$). Then:

$$\mathrm{XNOR}_o(p, q) = \mathrm{XNOR}_n(p, q) + \varepsilon(p, q)$$

where the interference remainder is:

$$\varepsilon(p, q) = (1 - p^2)(1 - q^2) / 8$$

Proof. Direct symbolic computation. $\mathrm{XNOR}_o(p,q) = \mathrm{AND}(\mathrm{OR}(-p, q), \mathrm{OR}(-q, p))$. Expanding via the multilinear polynomial formulas and simplifying gives $\mathrm{XNOR}_o(p,q) = -pq - p^2q^2/8 + p^2/8 + q^2/8 - 1/8$. Subtracting the native $-pq$ yields $\varepsilon(p,q) = -p^2q^2/8 + p^2/8 + q^2/8 - 1/8 = (p^2 - 1)(q^2 - 1)/(-8) = (1 - p^2)(1 - q^2)/8$. □

The interference remainder $\varepsilon(p,q)$ has three properties that are mathematically precise. First: $\varepsilon(p,q) = 0$ if and only if $p \in \{-1,+1\}$ or $q \in \{-1,+1\}$ — that is, whenever at least one input lies on the discrete Boolean boundary. At every vertex of $\{-1,+1\}^2$, XNOR_o and XNOR_n agree exactly; the composed circuit and the native operator are indistinguishable at the discrete level. Second: $\varepsilon(p,q) \geq 0$ throughout $[-1,+1]^2$, with maximum (1/8) at the origin $p = q = 0$. The remainder is largest precisely where the inputs are most uncertain. Third: $\varepsilon(p,q)$ is symmetric and factors as a product of two one-variable terms, each measuring the departure of one input from the discrete boundary: $\varepsilon(p,q) = \frac{1}{4}(1 - p^2) \cdot \frac{1}{4}(1 - q^2) \cdot 2$.

The proposition extends to complex inputs. If $p, q \in \mathbb{C}$ with $|p| \leq 1$ and $|q| \leq 1$, the same algebraic identity holds and $\varepsilon(p,q) = (1 - p^2)(1 - q^2)/8$ is now complex-valued. The remainder vanishes if and only if $p \in \{-1,+1\}$ **or** $q \in \{-1,+1\}$. For complex inputs with imaginary parts ($p = a + bi$, $b \neq 0$), $p^2 = a^2 - b^2 + 2abi \neq 1$ in general, and the interference remainder is a complex number whose real and imaginary parts both contribute to the mismatch.

The practical consequence is that in the continuous and complex extensions of the tile framework, **native multilinear polynomial representations are not equivalent to syntactic compositions**. A circuit built from simpler gates introduces a residual term absent from the native operator. On the Boolean vertices this is invisible; inside the hypercube it is not. This distinguishes two otherwise equivalent presentations of a connective and provides a concrete sense in which the native polynomial representation is ‘simpler’ than its syntactic decomposition: the native form is the unique polynomial with no interference remainder, regardless of the input domain.

The tile framework as developed so far covers the full propositional layer: 16 connectives, functional completeness via NOR, Fourier spectral analysis, and Łukasiewicz fuzzy extension. A natural question is whether first-order and higher-order quantification — the logical machinery of “for all” and “there exists” — fit naturally into this geometric picture. They do, and the fit is exact.

12.1 First-Order Quantification as Axis Collapse

In a two-variable Boolean function f(A, B), quantifying over A means collapsing the A-dimension of the 2×2 tile using a pooling rule:

- ∀A f(A,B) = f(−1, B) AND f(+1, B) — AND-pool across the A-rows
- ∃A f(A,B) = f(−1, B) OR f(+1, B) — OR-pool across the A-rows

The 2×2 tile becomes a 1×2 strip. The aggregation function determines which cells survive the collapse. This is not an analogy: it is exactly what the downsampling pipeline does with a specific pooling function. **Quantification is downsampling. ∀ is AND-pooling. ∃ is OR-pooling.**

Each application of a quantifier reduces the Bit Budget B by exactly 1 — the same as descending one quadtree level. Quantification is **logical zoom**: it reduces the resolution of the truth surface by collapsing one variable dimension, exactly as geometric downsampling reduces spatial resolution by collapsing one spatial dimension.

Figure 14 shows this concretely for XOR (Index 6). ∀A applied to XOR yields FALSE (AND-pooling the two rows produces all-void) — XOR never holds for all A simultaneously. ∃A applied to XOR yields TRUE (OR-pooling yields all-live) — for every value of B there exists some A making XOR hold. The figure also demonstrates that ∀A∃B ≠ ∃B∀A on the NOR depth-2 surface, confirming that quantifier order corresponds to the non-commutativity of sequential downsampling operations.

Figure 14 — Quantification as Downsampling: Collapsing Axes of the Boolean Hypercube

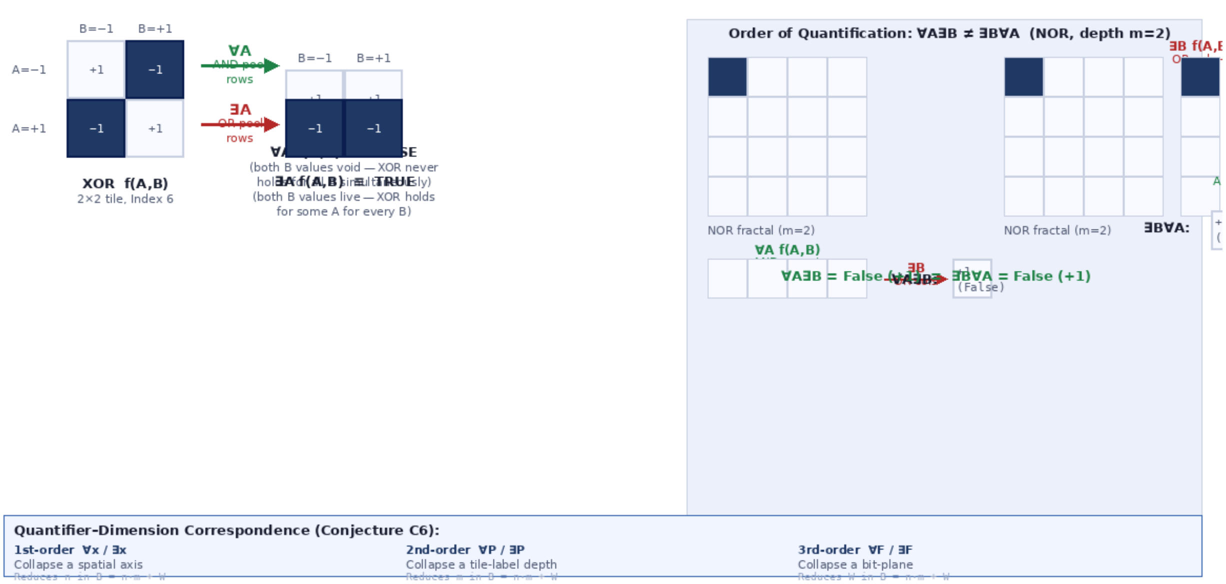

Figure 14. Quantification as downsampling. Top: XOR tile showing ∀A (AND-pool rows → FALSE) and ∃A (OR-pool rows → TRUE). Right: the NOR depth-2 surface demonstrating ∀A∃B ≠ ∃B∀A — quantifier order is the geometric non-commutativity of axis-collapse operations. Bottom: the three-order quantifier–dimension correspondence (Section 19).

In the {-1, 1} basis with True = −1, the sign convention means AND corresponds to **max** (takes the less True = more positive value) and OR corresponds to **min** (takes the more True = more negative value). So $\forall A\, f(A,B) = \max_{A \in \{-1,+1\}} f(A,B)$ and $\exists A\, f(A,B) = \min_{A \in \{-1,+1\}} f(A,B)$. This connects immediately to the Łukasiewicz section (Section 11): in the continuous fuzzy extension, $\forall x\, P(x) = \sup_{x \in [-1,1]} P(x)$ and $\exists x\, P(x) = \inf_{x \in [-1,1]} P(x)$, giving quantification over the full fuzzy interior as a natural generalisation of Boolean axis-collapse.

12.2 The Spectral Reading of Quantification

The Fourier coefficient $\hat{f}(\varnothing) = E[f(x)] = (1/2^n)\Sigma_x f(x)$ is the average truth value across all inputs — the DC component of the spectral expansion. In {-1, 1}:

- $\forall x\, f(x)$ holds iff $\hat{f}(\varnothing) = -1$: all energy at DC, uniformly True
- $\exists x\, f(x)$ holds iff $\hat{f}(\varnothing) < +1$: not everything False — some negative component exists
- $\neg\exists x\, f(x)$ holds iff $\hat{f}(\varnothing) = +1$: constant False — the Contradiction tile, Index 0

Applying quantifier ∀x to eliminate a variable $x \in S$ from the Fourier expansion zeroes out every coefficient $\hat{f}(S)$ where $x \in S$. Variable elimination in logic is coefficient elimination in the spectrum. The KKL theorem — that every balanced Boolean function has at least one variable with influence $\Omega(\log n/n)$ — is a statement about ∃: there always exists a variable whose quantification would meaningfully alter the spectral structure.

12.3 Second-Order Quantification: Quantifying Over Tiles

Second-order logic quantifies over *predicates* — not individual values $x \in \{-1,1\}$ but functions $f : \{-1,1\}^n \to \{-1,1\}$. In the tile framework, this is extraordinarily natural: the 16 tiles **are** the functions. Second-order quantification ranges over which tile populates a node of the quadtree.

- ∀P φ(P) = AND over all 16 tiles of φ applied to each tile — collapses one quadtree depth level
- ∃P φ(P) = OR over all 16 tiles of φ applied to each tile — OR across the tile alphabet

First-order quantification collapses a *variable axis* of the grid, reducing n. Second-order quantification collapses a *tile-label axis* of the quadtree node labelling, reducing m. The quadtree is already a tree of tile labels; second-order quantification is quantification over the elements of the texture composition structure examined in Section 19.

Spectrally, second-order ∀P averages over the full 16-tile alphabet. By the symmetry of the tile set (the 16 operators form a balanced set under all Boolean automorphisms), this averaging zeroes out all asymmetric Fourier components and retains only the symmetric, degree-0 component — the DC term. Functions with high nonlinearity resist first-order linear approximation; the question of what geometric structures such functions generate under RBHM encoding — and whether their Hausdorff dimensions cluster at predictable values — remains open and connects to Theorem C2 (Section 5.3a).

12.4 Third-Order Quantification and Fractal Inexhaustibility

Third-order quantification ranges over *sets of predicates* — functionals F that take Boolean functions to truth values. In the RBHM framework, this corresponds to quantifying over which *sequence of tiles* (f_1, f_2, …, f_m) labels the depth levels of the level-varying quadtree. The 16^m distinct level-varying surfaces are the domain of third-order quantification over depth-m quadtrees.

Here the exact counting identity from Section 8.2 acquires its natural interpretation. The gap between $16^m = 2$^(4m) (the structured subspace) and 2^(4^m) (the full function space) is not a logical limitation — it is the **fractal inexhaustibility of Boolean function space**. At any finite depth m, the level-varying encoding captures a structured slice of a space that extends infinitely deeper. The surfaces beyond that slice are not permanently inaccessible: the NOR pipeline reaches them all. They simply require going deeper into the fractal. As m increases, the fraction of level-varying expressible surfaces to total surfaces vanishes — just as any finite rendering of a fractal reveals a smaller and smaller fraction of the infinite limit set.

This is the mathematical character of the framework: it is not incomplete in any logical sense, it is **inexhaustible**. The Universal NOR Hypercube at infinite depth contains every possible truth surface. Any finite depth m is a cross-section — always revealing structure, always leaving more structure deeper. This connects directly to St. Denis and Grim (1997): fractal images of formal systems are always finite approximations of infinite limit sets. The "incompleteness" is the incompleteness of any finite rendering of an infinite fractal, which is precisely the right way to understand why the level-varying encoding cannot cover everything at a given depth. At the next depth, it covers more — and there is always a next depth.

Third-order quantification collapses the W (bit-depth) axis of the Bit Budget. Parseval's Identity Σ_S f̂(S)² = 1 is the statement that **full third-order universal quantification normalises spectral energy to 1**: summing over all spectral subsets S completes the energy conservation of the system. It is the spectral completeness theorem, and it holds exactly because the {-1, 1} basis provides the symmetric inner product that makes orthogonality possible.

Figure 15 — Higher-Order Quantification, the Bit Budget, and Fractal Inexhaustibility

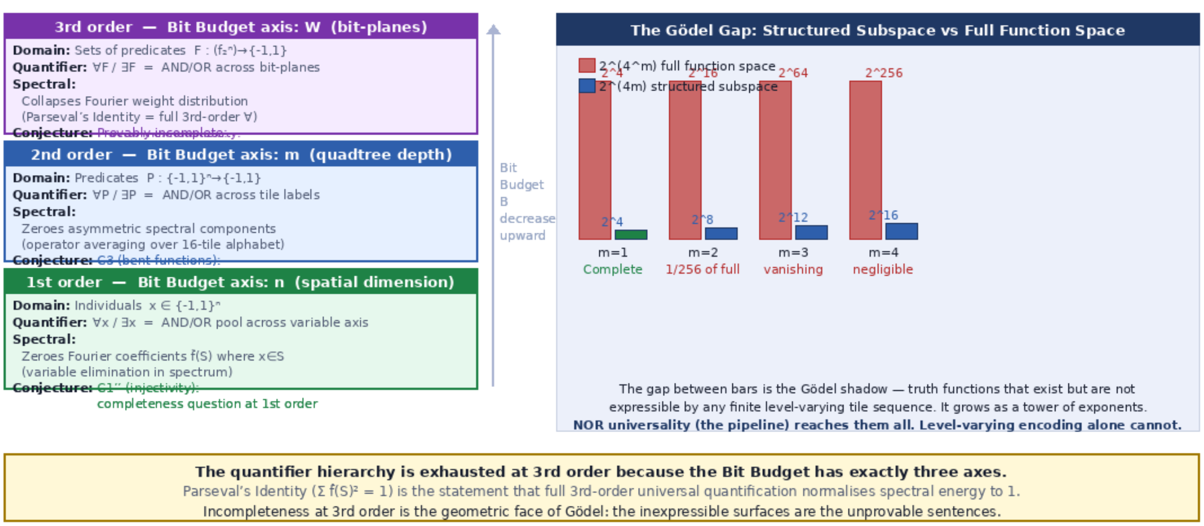

Figure 15. Higher-order quantification, the Bit Budget axes, and fractal inexhaustibility. Left: the three-layer hierarchy mapping quantifier order to Bit Budget axis. Right: the structured subspace (2^(4m), blue) vs the full function space (2^(4^m), teal) at each depth m. The shrinking ratio is not a logical gap — it is fractal inexhaustibility: at any finite

depth, the level-varying encoding sees a slice of an infinite structure. The Universal NOR Hypercube at infinite depth contains everything. Fractals are never finished.

12.5 The Natural Boundary at Third Order

The quantifier hierarchy is **exhausted at third order** within the Universal NOR Hypercube framework, because the Bit Budget has exactly three axes. There is no natural fourth axis in the (n, m, W) parameterisation, and therefore no natural fourth-order quantification. This is a structural feature of the framework, not a limitation: it reflects the fact that the Universal NOR Hypercube is fully parameterised by three degrees of freedom.

The three orders of quantification also correspond to the three topological reasons why 3D is the natural folding target (Section 9.3): first-order quantification collapses space (dimension), second-order collapses scale (depth), third-order collapses value (bit-depth). The dimensional folding isomorphism and the quantifier hierarchy are two perspectives on the same three-parameter structure.

PART II — THE GEOMETRY OF INFORMATION

Bit budget • Downsampling pipeline • n-Dimensional tile alphabets • Universal NOR hypercube

13. The Bit Budget: Three Axes of One Parameter

The IFS integer encoding ($f_0(p) = 2p$, $f_1(p) = 2p+1$) makes a structural equivalence explicit. Each iteration appends exactly one new bit to the cell's path address. One more iteration is simultaneously one more bit of spatial resolution, one more bit of value depth, one more quadtree level, and one more Boolean variable. Spatial dimension n, resolution depth m, and value bit-depth W are three names for the same zoom operation applied along different axes of the same recursive structure.

Definition 13.1 (The Bit Budget). The total bit budget B of an (n, m, W) surface is $B = n \cdot m + W$. The allocation of B among n (dimension), m (depth), and W (bit-planes) is a choice of representation, not a property of the object. A depth-3 2D binary surface ($B = 6$, allocation $2\times3+0$) is the same object as a depth-2 2D surface with 2-bit values ($B = 2\times2+2 = 6$): same budget, different allocation.

The term is borrowed deliberately from engineering, where a bit budget governs how a fixed payload is allocated across resolution, quality, and temporal axes — spending more on one means less for the others. The mathematical concept here is identical: a fixed total B distributes across spatial dimension n, recursion depth m, and value range W, and allocating more to any one axis necessarily reduces what remains. $B = n \cdot m + W$ as a unified topological parameter is an original construct of this framework, but the intuition it names is the same engineering intuition applied to logical space rather than data transmission.

Incrementing B by exactly one unit corresponds simultaneously to: computing one more hierarchical quadtree level; moving into one higher dimension of spatial detail; or introducing one more distinct variable into the Boolean logic space. High-resolution complexity structurally and geometrically subsumes all lower-resolution iterations. This is the Bit Budget as a unifying parameter, and it is the geometric reason why quantification (Section 12) reduces B by exactly 1 per quantifier application.

14. The Downsampling Pipeline: From NOR Fractals to Any Surface

The constructive proof of universality is the downsampling pipeline. Given target parameters (n, m, W), the pipeline produces any n-dimensional hypercubic surface from W independent NOR fractals in three steps.

Definition 14.1 (The Downsampling Pipeline). Given n (spatial dimension), m (resolution depth per axis), W (value bit-depth):

- Step 1. Generate W independent n-dimensional NOR fractals at source depth $M \geq m$, producing W binary volumes of size $2^T \times \cdots \times 2^T$ (n axes).
- Step 2. Downsample each volume by factor 2^(M−m) along every axis: $b\_k(i_1, \ldots, i\_n) = V\_k(i_1 \cdot 2\text{\^{}}(M-m), \ldots, i\_n \cdot 2\text{\^{}}(M-m))$.
- Step 3. Combine W binary volumes by bit-plane concatenation: $\text{value}(i_1, \ldots, i\_n) = b\_\{W-1\} \cdot 2\text{\^{}}(W-1) + \cdots + b_1 \cdot 2 + b_0$. Result: an n-dimensional grid of side 2^m per axis with values in $\{0,\ldots,2^W-1\}$.

Theorem 8.4 (NOR Universality, restated). For any $n \geq 1$, $W \geq 1$, $m \geq 1$, and any n-dimensional hypercubic surface S of side 2^m with values in $\{0,\ldots,2^W-1\}$, there exist W independent n-dimensional NOR fractals at source depth $M \geq m$ whose pipeline output exactly reproduces S.

Proof sketch: any Boolean function of any arity is expressible as a finite NOR circuit (functional completeness of NOR). The n-dimensional quadtree depth corresponds directly to NOR circuit depth. W independent planes address W independent bits. □

13.1 The 3D Concrete Example (n=3, m=1, W=2)

The smallest non-trivial volumetric case: a 2×2×2 cube with 4-valued cells (values in {0,1,2,3}). Parameters: $n = 3$, $m = 1$, $W = 2$, bit budget $B = 3 \cdot 1 + 2 = 5$.

- Step 1: Generate two independent 3D NOR fractals at source depth $M = 4$, producing 16×16×16 binary volumes (b_1 = high bit-plane, b_0 = low bit-plane).
- Step 2: Downsample each volume to 2×2×2 by factor 8 along all three axes.
- Step 3: Combine as $\text{value} = 2 \cdot b_1 + b_0$ to produce the 4-valued 2×2×2 cube.

The 8 cells each hold a value in {0,1,2,3}, giving $4^8 = 65{,}536$ possible target cubes. The structured subspace directly addressable by one (operator, operator) pair for the two bit-planes has $16 \times 16 = 256$ members — injective on non-degenerate pairs (open problem, Section 19; confirmed computationally at $m = 2,3$).

15. Dimensional Constraint and n-Dimensional Tile Alphabets

In n-dimensional space each IFS iteration consumes exactly n bits — one per spatial axis. Valid total spatial bit depths must therefore be multiples of n. Non-multiples produce non-hypercubic grids that break tile symmetry. This resolves an apparent awkwardness: 3 bits is not a valid spatial depth in 2D (iterations consume 2 bits each) but is the exact atomic level in 3D (one octtree iteration produces a 2×2×2 cube with $2^8 = 256$ possible tile functions).

n	Atomic tile	Tile alphabet 2^(2^n)	Tree arity	Bits/level
1	2-cell strip	`2^(2^1) = 4`	2-ary	1
2	2×2 square	`2^(2^2) = 16`	4-ary quadtree	2
3	**2×2×2 cube**	**`2^(2^3) = 256 ★`**	8-ary octtree	3
4	2^4 tesseract	`2^(2^4) = 65,536`	16-ary	4
n	2^n hypercube	`2^(2^n)`	2^n-ary	n

★ n=3 uniquely matches the byte (256 tile functions = 2^8), the natural atom of digital memory indexing. This is the algebraic reason 3D is the natural reduction target of the dimensional folding isomorphism.

16. The Universal NOR Hypercube

Definition 16.1 (The Universal NOR Hypercube). The universal NOR hypercube is the infinite-dimensional, infinite-resolution binary structure whose value at any vertex is determined by recursive NOR application across all coordinates. Every finite surface in the framework is a finite projection P(n, m, W) of this single object: n selects which spatial axes to expose, m selects the resolution depth, and W selects how many bit-depth axes to include.

One-line statement: all n-dimensional hypercubic surfaces at all resolutions and all bit-depths are cross-sections of a single infinite-dimensional recursive NOR structure. The sixteen 2D connectives, the 256 3D connectives, the Sierpiński gasket, and any W-bit volumetric surface are the same object — described by which axes are exposed and how many recursion levels are unfolded.

Moving up in spatial dimension is zooming out; moving down is zooming in. An (n+1)-dimensional NOR fractal contains the n-dimensional one as its corner face. These are not different objects — they are the same attractor at different scales and axis exposures. The fractal inexhaustibility of Section 8.2 is the statement that no finite cross-section captures the whole: the object is always there, but any finite view is always partial.

17. Topological Preservation Under Downsampling

When a logical space is simplified — variables abstracted, resolution reduced — the structural integrity of truth regions must be preserved. Naive reductions such as max-pooling and average-pooling arbitrarily destroy topology, altering the Betti numbers of the live-cell configurations. Optimisation-based downsampling that respects discrete Boolean constraints preserves topological invariants but is computationally expensive.

The RBHM quadtree is itself the natural downsampling hierarchy. Moving from depth m to depth m−1 is simply reading one level higher in the same tree — the aggregation rule for the parent cell is determined by the tile label at that node. No external curve or pooling kernel is required. Downsampling in this framework is **quadtree level selection**: the fractal is always there at every depth, and coarser resolution is a shallower read of the same recursive structure. This is the topological preservation claim: the quadtree hierarchy, because it is generated by the same tile rules at every level, preserves the logical structure of the surface at every scale by construction.

Locality within a quadrant is preserved natively. Cells within the same quadtree sub-quadrant at any depth are physically adjacent and are addressed by contiguous quadtree path codes. Locality across quadrant boundaries is the open question addressed in Section 19.

18. Why 3D is the Natural Reduction Target

The dimensional folding isomorphism establishes that any n-dimensional Boolean surface is isomorphic to a 3D surface. But why 3D specifically? Three independent reasons converge.

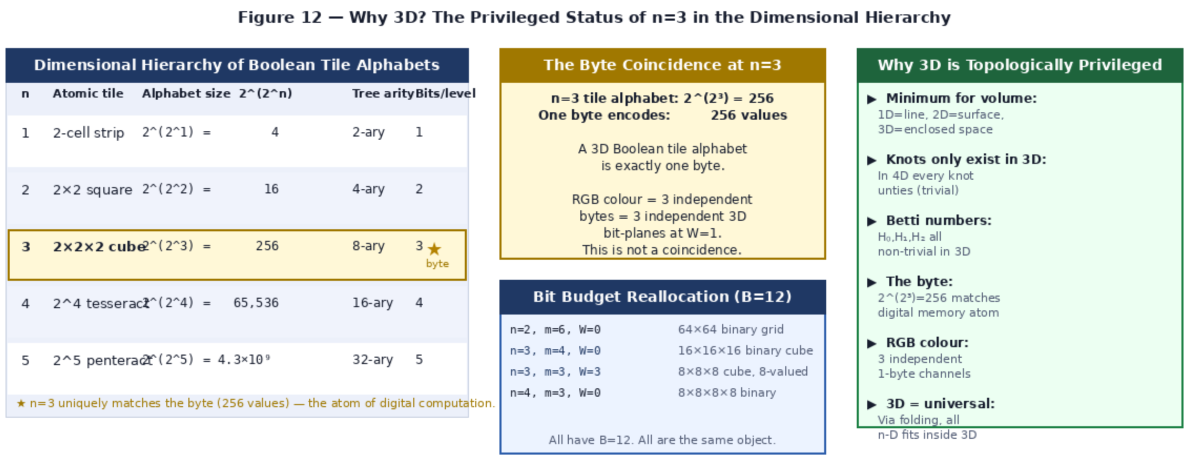

Figure 12. The dimensional hierarchy of Boolean tile alphabets. n=3 uniquely matches the byte ($256 = 2^8$ tile functions), the atom of digital memory. The bit-budget reallocation panel shows that any B=12 surface — regardless of how n, m, W are allocated — is the same object in a different coordinate system.

- Topological privilege. 3D is the minimum dimension with enclosed volume. In 1D there are only points and intervals; in 2D, surfaces but no interior; in 3D, enclosed space first appears. Knot theory is non-trivial in 3D (knots cannot be untied) but trivial in 4D (every knot untangles). The Betti numbers H_0, H_1, H_2 are all simultaneously non-trivial only in 3D.
- The RGB / colour connection. Digital colour is represented as three independent one-byte channels (R, G, B). This is precisely three independent 3D bit-planes at $W = 1$ — exactly the $n = 3$, $W = 1$ allocation of the tile framework. The folding from n-D to 3D is the abstract generalisation of how any volumetric dataset is stored as a colour volume.

Computational note. The 3D tile alphabet has size $2^{(2^3)} = 256 = 2^8$, which happens to equal one byte. This alignment is a useful engineering convenience: it means the 3D Boolean tile alphabet indexes naturally into byte-addressed memory without waste or overflow, and it connects the framework natively to byte-oriented hardware. However, this is a coincidence of 20th-century engineering convention (the 8-bit byte was standardised for alphanumeric character encoding, e.g. IBM System/360, 1964) rather than a mathematical necessity. It is noted here as a practical observation, not as a topological argument for why $n = 3$ is privileged.

19. Open Problems and Structural Observations

The following problems remain open. Theorem C2 (IFS measure convergence) is proven in Section 5.3a. Results C4 and C5 are resolved and summarised as notes below.

Injectivity of Level-Varying Encodings (Open Problem)

The level-varying map from m-tuples of non-degenerate connectives (excluding FALSE and TRUE) to Boolean functions is injective: distinct tuples produce distinct functions. Computational verification confirms the conjecture at $m = 2$ (all 196 non-degenerate pairs produce distinct surfaces) and $m = 3$ (all 2744 triples produce distinct surfaces). A general proof for arbitrary m remains open.

Structural Correspondence: Quantifier Order and Bit Budget Axes

Within the (n, m, W) parameterisation of the Universal NOR Hypercube, the three orders of quantification correspond to the three axes of the Bit Budget $B = n \cdot m + W$: first-order ($\forall x$ / $\exists x$) reduces n; second-order ($\forall P$ / $\exists P$) reduces m; third-order ($\forall F$ / $\exists F$) reduces W. Parseval's Identity — Σ_S f̂(S)² = 1 — is the spectral statement of full third-order universal quantification.

Scope. This is a structural property of the finite fractal parameterisation, not a claim about mathematical logic in the standard model-theoretic sense. Standard higher-order logic ranges over infinite power sets, leading to Gödelian incompleteness that breaks finite dimensional bounds. The correspondence holds within the Bit Budget framework precisely because it restricts to bounded recursive structures.

Notes on Resolved Questions

C4 (Algebraic Closure) — false. The 16 tiles are not closed under texture composition. Exhaustive enumeration of all 256 pairs at depth $m = 2$ shows 46 closed pairs (18%) and 210 non-closed pairs (82%). The subset that closes consists of the constant tiles and idempotent tiles under self-composition. Identity (tile 3) and associativity hold; closure does not. The 16-tile set is therefore not a monoid under texture composition.

C5 (Hilbert Locality) — false. No space-filling curve provides worst-case locality bounds independent of dimension. Alber and Niedermeier (2000) proved that the L_∞ distance between adjacent cells in a multidimensional Hilbert curve scales as O(n), where n is the dimension. The Hilbert curve improves average-case locality over the Morton code but cannot eliminate dimensional dependence at worst-case quadrant boundaries.

20. Computational Insights and Research Directions

The Boolean Tile Basis is a theoretical framework, but its structure suggests several concrete directions for computational application and further investigation.

Fractal dimension as Boolean function complexity. The Hausdorff dimension of a tile's IFS attractor is computable from its Fourier spectrum: dim_H = log(N)/log(1/r) where N is the number of live cells in the 2^n rule cube. Since the dominant Fourier coefficient determines N (the XOR tile concentrates all weight on degree-2, giving N=2 contractions and dim = 1), the fractal dimension is a geometric reading of spectral concentration. This suggests a new complexity measure for Boolean functions: operators whose attractors have lower Hausdorff dimension have more spectrally concentrated Walsh representations and may be easier to approximate linearly. The 16 base tiles span a range from dim=0 (corner-point attractors) through dim ≈ 1.585 (Sierpiński family) to dim=2 (TRUE fills the plane).

This approach is corroborated by two independent lines of established literature. Jack Lutz's theory of resource-bounded dimension (Lutz 2003) explicitly links Hausdorff dimension to computational complexity by effectivizing fractal dimension using gales — generalisations of martingales — to measure the complexity of subsets of decidable languages. Resource-bounded dimension characterises the polynomial-size Boolean circuit complexity of a space, proving that fractal dimensions can precisely bound and define algorithmic complexity limits. Ewert's analysis of elementary cellular automata (ECAs) — which are structurally equivalent to iterative Boolean functions on local neighbourhoods — defines a complexity measure D_{ASS} derived directly from the minimised Boolean function underlying the automaton's transition rule and computed from the fractal dimension of its algorithmic state space (Ewert 2017). The RBHM framework's approach of deriving Hausdorff dimensions from live-cell IFS configurations aligns directly with this methodology. Computing fractal dimension for composed tile sequences could provide a geometric measure of circuit complexity that extends naturally from the 16-tile base alphabet to arbitrary Boolean circuits.

Geometric view of quantifier scope. The mapping of ∀ and ∃ to AND-pooling and OR-pooling (Section 19) provides an unconventional geometric interpretation of database query evaluation. A $\forall x\, P(x, y)$ query collapses the x-axis of the truth surface by AND-pooling, producing a lower-dimensional surface over y. This is exactly max-pooling along one axis of the IFS grid. The order of quantification determines the order of axis collapses, and the non-commutativity of $\forall x \exists y \neq \exists y \forall x$ is geometrically visible as different downsampling sequences producing different residual surfaces. Whether this geometric view enables new optimisations for nested quantifier evaluation is an open question.

The NOR pipeline as circuit synthesis. Theorem 8.4 (NOR Universality) states that any Boolean surface is reachable from NOR fractals at sufficient depth via the downsampling pipeline. This provides a uniform, dimension-independent representation for any Boolean function: as a depth-M NOR fractal downsampled to target parameters (n, m, W). A natural question is whether the "shortest NOR circuit" for a given Boolean function corresponds to the minimum source depth M in the pipeline. If so, the NOR pipeline is simultaneously a circuit model and a fractal compression scheme, and the fractal inexhaustibility of the level-varying subspace (Section 8.2) has a direct interpretation as the gap between circuit-expressible and circuit-inexpressible functions at bounded depth.

20a. Computational Methods and Reproducibility

All figures and numerical results in this paper were generated by deterministic Python scripts. This section describes the algorithms used, sufficient for independent verification.

IFS fractal generation (Figures 1–6, 9–10). Fractal attractors are generated by direct deterministic recursion, not by the chaos game. Starting from the 2×2 tile rule, the algorithm recursively substitutes each live cell with the 2×2 tile pattern scaled by 1/2, proceeding to depth m. At depth m, the grid has $2^m \times 2^m$ cells, each marked live or void. The 16-tile heatmap survey (Figure 2) uses $m = 4$ (16×16 cells). Key operator attractors (Figure 4) use $m = 6$ (64×64 cells). The NOR depth-m figures (Figures 5, 6) use $m = 5$. Colour encoding: navy = live (True = −1), white = void (False = +1). All computations use integer arithmetic only; no floating-point approximation is involved in determining live/void state.

Box-counting dimension verification (Figure 3, Table 1 in Section 5.3). Given a depth-M grid, the box-counting dimension is estimated as follows. For each box size 2^b (b from 1 to M−1), count the number of boxes of that size that contain at least one live cell. Plot log(count) vs log(2^b) and fit a line; the slope is the box-counting dimension estimate. For the 2D XOR attractor (Sierpiński), depths $M = 6, 7, 8$ yield counts 3^J, 3^K, 3^8 respectively, giving dimension estimate $\log(3)/\log(2) \approx 1.58496$ at all depths. For the 3D XOR attractor, depths $M = 4, 5, 6$ yield counts 4^4, 4^5, 4^6, giving estimate $\log(4)/\log(2) = 2.00000$. Both results are exact to the precision reported.

Łukasiewicz heatmaps (Figure 9). The $\{-1,1\}^2$ unit square is sampled on a 512×512 grid. For each sample point (x,y), the Łukasiewicz operators are evaluated using their {-1,1} formulas (negation: $-x$; equivalence: $|x-y|-1$; conjunction: $\min(1,x+y+1)-1$; etc.) and the result mapped to the navy–white–gold colour scale. The tent map iteration (Figure 8) uses 10,000 iterations from initial condition $x_0 = 0.1$, with cobweb paths drawn geometrically.

Software and availability. All figure generation scripts are written in Python 3.12 using NumPy (array operations), Matplotlib (rendering), and Pillow (PNG output). The document build system uses Node.js with the docx library for DOCX generation and LibreOffice for PDF conversion, with Unicode post-processing for consistent f̂ notation. The complete build system including all figure generators is available at the IFS Mega Heatmap Survey repository (tltmedia.github.io/ThinkingMattersExercises/mathFun/ifs_mega_heatmap_survey.html); the paper build scripts will be released alongside publication.

The spectral structure of propositional logic established in this paper surfaces in several other domains. Most directly in logic: Kalai (2002) showed that Arrow's Impossibility Theorem admits a Fourier-analytic treatment in which each voter's influence is a coordinate in $\{-1,1\}^n$, and any noise-stable social choice function is provably close to a dictator. This is a quantitative Fourier-analytic strengthening of Arrow — it makes the structure of the impossibility geometrically precise — rather than a direct application of the FKN theorem, which requires a balance condition that social choice functions do not always satisfy. The shared ingredient is the Walsh-Hadamard spectral decomposition: the same framework that makes the 16-tile attractors visible also makes Arrow's result computable.

Beyond logic: Hadamard conjugation — the direct analogue of the Walsh-Hadamard Transform over quaternary alphabets — is used in phylogenetics to decompose fitness landscapes into additive and epistatic components. The {-1, 1} symmetry that makes Boolean Fourier analysis clean extends naturally to these settings. These connections are noted as evidence that the spectral structure of the tile basis is not particular to propositional logic but reflects something deeper about the mathematics of binary and near-binary structures.

22. Conclusion

The Boolean Tile Basis begins with two propositional variables P and Q. They generate exactly 16 distinct truth functions — the 16 tiles. These tiles form a self-contained alphabet: P and Q are themselves tiles (Indices 3 and 5), not external primitives. NOR is the sole generator from which all 15 others are constructible. This is the flat ontology: operators, variables, and constants inhabit one categorical level.

Each tile generates a unique IFS fractal attractor under recursive self-substitution. The fractal images are not illustrations of logic — they *are* the logic, rendered spatially. XOR generates the Sierpiński triangle at Hausdorff dimension $\log(3)/\log(2) \approx 1.585$, a dimension determined by

XOR's Fourier structure: one dominant coefficient maps to one dominant void pattern. The {-1, 1} basis makes this connection algebraically exact.

Extending to many-valued logic, the Łukasiewicz operators translate into {-1, 1} in unusually clean form. Negation becomes pure reflection $x \mapsto -x$. Strong disjunction and conjunction become symmetric clamped sums. And the self-equivalence paradox $p \leftrightarrow \neg p$ becomes the map $f(x) = 2|x| - 1$: a provably chaotic system with Lyapunov exponent log 2 and no stable truth value. Logical self-reference generates deterministic chaos. This is not a metaphor; it is a theorem.

Logical quantification maps exactly onto the geometry. ∀x is AND-pooling along the variable axis. ∃x is OR-pooling. Each quantifier reduces the Bit Budget $B = n \cdot m + W$ by 1, collapsing one degree of freedom of the logical space. The three orders of quantification — first (over individuals), second (over predicates), third (over predicate sequences) — correspond to the three axes n, m, W. The hierarchy is exhausted at third order because logical space has exactly three degrees of freedom. Parseval's Identity, $\Sigma\_S \hat{f}(S)^2 = 1$, is the spectral form of full third-order universal quantification.

After any sequence of quantifier applications, the residual logical space is 3-dimensional — the natural fixed point of the quantification process, where the tile alphabet reaches 256 elements, matching the byte. The logical space is inexhaustible: at any finite depth, the level-varying encoding sees a structured slice of a space extending infinitely deeper. The NOR fractal at infinite depth contains everything. There is always more at the next level of resolution.

The 1997 paper established that formal systems have fractal images. This paper establishes *why* they must: because a 2×2 tile is a quadtree node, a quadtree generates Morton path codes, and a Morton-addressed quadtree under a Boolean rule is an IFS. The fractal images are not found — they are derived. The Sierpiński triangle is the carry-free locus of XOR, proven by Lucas's theorem. The tent map is the self-referential fixed point of Łukasiewicz logic. The quantifier hierarchy is the axis-collapse structure of the quadtree. The space of truth functions is inexhaustible in exactly the sense that the fractal is inexhaustible: always more detail at the next depth, always the same structure at every scale. And the unique privileged dimension is 3, where the XOR fractal becomes a complete surface, the tile alphabet matches the byte, and the Sierpiński voids are algebraically closed by the XNOR third coordinate.

Appendix A. Polynomial Derivations Under Convention A

This appendix derives the {-1, 1} multilinear polynomial for three representative operators — AND, OR, and XOR — directly from their truth tables, step by step. The purpose is to make the sign convention concrete and prevent reader confusion when working with the {-1, 1} basis for the first time. The full table of all 16 operators appears as Table 1 in Section 3.3.

Convention A (recap). True = −1, False = +1. The bijection is $x = 1 - 2b$ where $b \in \{0, 1\}$ is the classical bit. Inputs $A, B \in \{-1,+1\}$; outputs $f(A,B) \in \{-1,+1\}$.

General method. Any function $f: \{-1,+1\}^2 \to \{-1,+1\}$ has a unique multilinear polynomial representation $f(A,B) = c_0 + c_1A + c_2B + c_3AB$. The coefficients are computed by the Fourier formula c_S = (1/4) Σ_{A,B} f(A,B) · A^{1_{A ∈ S}} · B^{1_{B ∈ S}} over the four input pairs $(A,B) \in \{(-1,-1),(-1,+1),(+1,-1),(+1,+1)\}$.

A.1 AND (Index 1, binary 0001)

Truth table under Convention A:

```
A=-1, B=-1: AND(T,T)=T  →  f = -1
A=-1, B=+1: AND(T,F)=F  →  f = +1
A=+1, B=-1: AND(F,T)=F  →  f = +1
A=+1, B=+1: AND(F,F)=F  →  f = +1
```

The native polynomial for AND is obtained by noting that AND(A,B) = True only when both $A = -1$ and $B = -1$. Using the projector $\frac{1}{2}(1 - A)$ which equals 1 when $A = -1$ and 0 when $A = +1$, and likewise for B:

$$\mathrm{AND}(A,B) = -1 \cdot \tfrac{1}{2}(1-A) \cdot \tfrac{1}{2}(1-B) + (+1) \cdot [1 - \tfrac{1}{2}(1-A)\cdot\tfrac{1}{2}(1-B)]$$

$$= 1 + A/2 + B/2 - AB/2$$

Verification: $\mathrm{AND}(-1,-1) = 1 -\frac{1}{2} -\frac{1}{2} -\frac{1}{2} = -1$ ✓; $\mathrm{AND}(-1,+1) = 1 -\frac{1}{2}+\frac{1}{2} +\frac{1}{2} = +1$ ✓; $\mathrm{AND}(+1,+1) = 1+\frac{1}{2}+\frac{1}{2} -\frac{1}{2} = +1$ ✓. This is the H_1 matrix: the constant term ½ and degree-1 coefficients ½, ½, −½ give the row (1, +1/2, +1/2, -1/2) of the Walsh-Hadamard Transform, confirming AND = H_1 (Section 7.1).

A.2 OR (Index 7, binary 0111)

Truth table under Convention A: OR is True if either input is True.

```
A=-1, B=-1: OR(T,T)=T  →  f = -1
A=-1, B=+1: OR(T,F)=T  →  f = -1
A=+1, B=-1: OR(F,T)=T  →  f = -1
A=+1, B=+1: OR(F,F)=F  →  f = +1
```

OR is True everywhere except (F,F). Using the projector for the False-False pair $\frac{1}{2}(1+A)\cdot\frac{1}{2}(1+B)$:

$$\mathrm{OR}(A,B) = -1 + (1 - \tfrac{1}{2}(1+A)\cdot\tfrac{1}{2}(1+B))\cdot 2 = -1 + A/2 + B/2 + AB/2$$

Verification: $\mathrm{OR}(+1,+1) = -1+\frac{1}{2}+\frac{1}{2}+\frac{1}{2} = +1$ ✓; $\mathrm{OR}(-1,+1) = -1 -\frac{1}{2}+\frac{1}{2} -\frac{1}{2} = -1$ ✓. The degree-2 coefficient is +½, opposite in sign to AND's −½. This reflects the De Morgan relationship: $\mathrm{OR}(A,B) = -\mathrm{AND}(-A,-B)$, and negating both inputs flips the sign of all odd-degree terms while leaving the degree-2 term unchanged in magnitude but flipped in the constant term.

A.3 XOR (Index 6, binary 0110)

Truth table under Convention A: XOR is True when inputs differ.

```
A=-1, B=-1: XOR(T,T)=F  →  f = +1
A=-1, B=+1: XOR(T,F)=T  →  f = -1
A=+1, B=-1: XOR(F,T)=T  →  f = -1
A=+1, B=+1: XOR(F,F)=F  →  f = +1
```

XOR is the parity function: True iff $A \neq B$. In the {-1,+1} basis, parity is multiplication: $\mathrm{XOR}(A,B) = AB$. Verification: $(-1)(-1) = +1 = F$ ✓ (two Trues give False); $(-1)(+1) = -1 = T$ ✓ (True XOR False gives True). The polynomial has zero constant term, zero degree-1 terms, and degree-2 coefficient exactly 1. This means the entire spectral weight falls on the {A,B} Fourier coefficient: $\hat{f}(\{A,B\}) = 1$, all others zero. By the Spectral-Geometric Law (Section 7.2a), concentrated degree-2 weight corresponds to the anti-diagonal Cantor structure (Hausdorff dim = 1), and the carry-free isomorphism (Theorem 5.1) ties this to the Sierpiński gasket via the zero-set family.

The three polynomials illustrate the sign-convention pattern: all three operators evaluate to (False = +1) at the all-False input ($A = B = +1$), so the constant term c_0 records this base value, and the degree-1 and degree-2 terms encode how the function departs from it as inputs move toward

True (−1). The degree-2 term ·AB measures whether the two inputs cooperate (positive: AND-like) or anti-cooperate (negative: XOR-like) in producing True outputs.

References

1. St. Denis, P. and Grim, P. “Fractal Images of Formal Systems.” Journal of Philosophical Logic, 26(2): 181–222, 1997. DOI: 10.1023/A:1004280900954.
2. Mar, G. and St. Denis, P. “What the Liar Taught Achilles.” Journal of Philosophical Logic, 28(1): 29–46, 1999. (Establishes isomorphisms between Zeno’s paradoxes and the Liar in infinite-valued logic; introduces the Chaotic Liar dynamical system and the Sisyphean fractal.)
3. Grim, P., Mar, G. and St. Denis, P. The Philosophical Computer: Exploratory Essays in Philosophical Computer Modeling. MIT Press, 1998.
4. St. Denis, P. The Boolean Tile Basis for Binary Surfaces: A Quadtree Unification of Logic, Geometry, and Dimension. Preprint, 2025.
5. O’Donnell, R. Analysis of Boolean Functions. Cambridge University Press, 2014.
6. O’Donnell, R. Some Topics in Analysis of Boolean Functions. CMU. https://www.cs.cmu.edu/~odonnell/papers/analysis-survey.pdf
7. Filmus, Y. Analysis of Boolean Functions — Lecture Notes. Technion, 2015. https://yuvalfilmus.cs.technion.ac.il/Courses/BooleanFunctionAnalysis/2015/LectureNotes.pdf
8. Kahn, J., Kalai, G. and Linial, N. The influence of variables on Boolean functions. Proc. 29th FOCS, pp. 68–80, 1988.
9. Kalai, G. A Fourier-theoretic perspective on Arrow’s theorem. Advances in Applied Mathematics, 29(1): 232–246, 2002.
10. Spiro, S. Fourier Analysis of Boolean Functions. University of Chicago REU, 2016. https://math.uchicago.edu/~may/REU2016/REUPapers/Spiro.pdf
11. Hutchinson, J. E. Fractals and self-similarity. Indiana University Mathematics Journal, 30(5): 713–747, 1981.
12. Barnsley, M. F. Fractals Everywhere. Academic Press, 1988.
13. Falconer, K. Fractal Geometry: Mathematical Foundations and Applications. Wiley, 1990. (Standard reference for Hausdorff dimension and IFS attractor theory.)
14. Wong, J. Space-filling curves and Hausdorff dimensions. NSYSU Mathematics. https://www-math.nsysu.edu.tw/~wong/papers/sfc_f.pdf
15. Lutz, J. H. The dimensions of individual strings and sequences. Information and Computation, 187(1): 49–79, 2003. (Resource-bounded dimension; links Hausdorff dimension to polynomial-size Boolean circuit complexity via effectivized gale-martingales.)
16. Ewert, U. Complexity of Elementary Cellular Automata. International Journal of Unconventional Computing, 13: 113–145, 2017. (Defines D_{ASS} complexity measure from minimised Boolean transition functions; fractal dimension of algorithmic state spaces aligns directly with the RBHM framework.)
17. Alber, J. and Niedermeier, R. On multidimensional curves with Hilbert property. Theory of Computing Systems, 33(4): 295–312, 2000. (Establishes that worst-case L_∞

adjacency distance for multidimensional Hilbert curves scales with dimension n; resolves the Hilbert locality question negatively.)

18. Mandelbrot, B. B. Intermittent turbulence in self-similar cascades: divergence of high moments and dimension of the carrier. Journal of Fluid Mechanics, 62: 331–358, 1974. (Fractal percolation — random Cantor sets by independent sub-square retention. The Boolean tile attractors studied here are the deterministic counterpart.)
19. Chayes, J. T., Chayes, L. and Durrett, R. Connectivity properties of Mandelbrot's percolation process. Probability Theory and Related Fields, 77(3): 307–324, 1988. (Phase transitions in fractal percolation; connectivity thresholds for self-similar random sets.)
20. Mandelbrot, B. B. The Fractal Geometry of Nature. W. H. Freeman, 1982. (Fractals as spatial modelling primitives; multiscale self-similar structure in natural surfaces and engineered designs.)
21. Morton, G. M. A computer oriented geodetic data base. IBM Technical Report, 1966. (The literature name for the quadtree path address structure — native to the RBHM, independently identified by Morton for spatial databases.)
22. Samet, H. The Design and Analysis of Spatial Data Structures. Addison-Wesley, 1990. (Quadtrees and spatial indexing.)
23. Post, E. L. The two-valued iterative systems of mathematical logic. Princeton University Press, 1941. (Functional completeness of NOR/NAND.)
24. Hájek, P. Metamathematics of Fuzzy Logic. Kluwer Academic, 1998. (Łukasiewicz infinite-valued logic.)
25. Friedgut, E., Kalai, G. and Naor, A. Boolean functions whose Fourier transform is concentrated on the first two levels. Advances in Applied Mathematics, 29(3): 427–437, 2002. DOI: 10.1016/S0196-8858(02)00024-6. (The FKN theorem; the balance condition is required for the dictator conclusion.)
26. Cipra, B. A. An introduction to the Ising model. The American Mathematical Monthly, 94(10): 937–959, 1987. DOI: 10.1080/00029890.1987.12000742.
27. Lucas, A. Ising formulations of many NP problems. Frontiers in Physics, 2: 5, 2014. DOI: 10.3389/fphy.2014.00005. (NP-hardness of Ising ground state and QUBO equivalence for Boolean satisfiability.)
28. Tsuiki, H. Real Number Computation through Gray Code Embedding. Theoretical Computer Science, 284(2): 467–485, 2002. (Gray-code topology for real number computation; the Sisyphean fractal is the 2D carry-free locus in Tsuiki's coordinate system.)
29. Conway, J. H. Mathematical Games: The fantastic combinations of John Conway's new solitaire game 'Life'. Scientific American, 223: 120–123, 1970.
30. Rothaus, O. S. On bent functions. Journal of Combinatorial Theory A, 20(3): 300–305, 1976. (Background on nonlinearity; bent functions noted but not the subject of a conjecture in this paper.)
31. IFS Mega Heatmap Survey (interactive visualisation, P. St. Denis). https://tltmedia.github.io/ThinkingMattersExercises/mathFun/ifs_mega_heatmap_survey.html