Noise-Tolerant Verification of Compositional Boolean Recovery

Pranay Jha [1]

Abstract

A network can be a few hundred bit-flips from the true parity rule yet have an algebraic normal form (ANF) of $24{,}998$ monomials. We call this the *Möbius noise zone*: small residual error catastrophically inflates apparent algebraic complexity, so the natural exact test for compositional recovery (symbolic ANF equality) spuriously rejects networks that behaviorally learned the rule. Accuracy alone errs the opposite way, accepting high-error shortcuts that match the target on easy inputs without computing it. We introduce a noise-tolerant verification framework: Reed–Muller decoding certifies whether the network lies inside the unique-decoding ball of a candidate of known ANF degree, and Walsh-spectrum agreement supplies an exact squared-L_2 spectral distance of 4δ (so L_2 distance $2\sqrt{\delta}$), where δ is the bit-error rate. Inside a small template bank of plausible compositional hypotheses, these checks rank the correct rule top-1 in $22/25$ task/output cells across all three seeds (23, 23, 22 per seed) on parity, modular addition, and modular multiplication; the failures correctly flag bits the network does not robustly recover. Relative to the intended rule, the residual $e = f \oplus h$ is structured: on modular multiplication, a one-line risk score yields a 3–10$\times$ spread between low- and high-risk failure rates (high-risk at 1.4–1.9$\times$ the random baseline), localizing where generalization breaks down.

1. Introduction

Setting. Train a neural network on a Boolean function with known compact structure (parity, bitwise modular addition, or bitwise modular multiplication), and ask whether the network has recovered the underlying *rule*, not just fit the training data. The natural exact test for rule-recovery breaks down under arbitrarily small residual error; we diagnose the failure and replace the exact test with two noise-tolerant checks from classical coding theory and Boolean Fourier analysis.

Compositional Boolean recovery. We use "compositional" here in the restricted algorithmic sense: the output is generated by a compact rule that composes local Boolean operations across input bits (parity over a fixed subset, bitwise addition modulo 2^k, bitwise multiplication modulo 2^k), rather than by a lookup table over the cube. Verification specifies a candidate compositional rule $h : \{0,1\}^n \to \{0,1\}$ and asks whether the network's induced behavior $f : \{0,1\}^n \to \{0,1\}$ (read off as a truth table) agrees with h. Held-out accuracy alone is insufficient: high agreement is consistent with both genuine compositional recovery and high-error shortcut policies that match h on the easy subset of inputs without computing it. The mathematically cleanest exact check is symbolic equality: compute the algebraic normal form (ANF, the unique GF(2) polynomial expansion) of f and compare to the ANF of h. We show that under small residual training error this exact check becomes catastrophically misleading: networks that have compositionally recovered the rule in any reasonable behavioral sense fail it dramatically.

Motivating observation. Train a 3-layer ReLU MLP on parity at $n = 16$. The network classifies 99.31% of the 2^{16} Boolean inputs correctly; behaviorally, it is extremely close to the parity rule. Extract the induced truth table and compute its ANF via the Möbius transform, however, and you find a degree-16 polynomial with $24{,}998$ monomials, where the clean target $x_1 \oplus \cdots \oplus x_n$ has support 16 and ANF degree 1. Exact ANF equality therefore rejects the hypothesis, even though the induced function sits deep inside the Reed–Muller decoding basin of clean parity and Reed's classical majority-logic decoder applied to the truth table yields the clean parity codeword. The instability lives in the exact symbolic readout, not in the recovered behavioral rule.

Recovery under residual error. The ANF instability is not parity-specific; it arises whenever exact symbolic recovery is tested under residual Hamming noise, and is sharpest when the candidate has known ANF degree. We replace exact symbolic equality with two noise-stable behavioral

[1]Independent. Correspondence to: Pranay Jha <pranay.jha@gmail.com>.

Accepted to the 2nd Workshop on Compositional Learning at ICML 2026, Seoul, South Korea. Copyright 2026 by the author(s).

lenses for the same "did the network recover the rule" question. The first is a coding-theoretic certificate. The truth table of a degree-$\leq r$ GF(2) polynomial in n variables is a *Reed–Muller codeword* $h \in \mathrm{RM}(r, n)$, and two distinct codewords differ on at least 2^{n-r} inputs (MacWilliams & Sloane, 1977). Therefore any received word (here, the network's induced truth table) within $2^{n-r-1} - 1$ Hamming flips of one codeword is closer to it than to any other. Given a candidate h of ANF degree r, we accept recovery iff

$$d_H(f, h) \leq 2^{n-r-1} - 1.$$

The second is a spectral measurement. *Walsh-spectrum agreement* reports an exact continuous quantity: by Parseval's theorem, the squared L_2 deviation between the induced and candidate Walsh spectra equals 4δ, where $\delta = \Pr[f(x) \neq h(x)]$ is the bit-error rate. The practitioner version of the question is rarely "does the network match this one rule" but "which compositional rule, if any, in a small bank of plausible candidates does the network recover"; we therefore use these checks inside a template-bank ranking that compares the induced behavior against candidates, reports margins to alternatives, and applies the Reed-radius certificate when a candidate has known ANF degree.

Where compositional recovery fails. A certificate tells us *whether* the induced behavior is close to a candidate compositional rule. It does not tell us *where* that rule fails, i.e., which inputs the network handles compositionally and which it falls back on something more like memorization-flavored exceptions. Relative to a reference candidate h (whether Reed-certified or merely close), the residual

$$e(x) = f(x) \oplus h(x)$$

is itself a Boolean function on the input space. We treat this residual as a second signal rather than as nuisance noise. In our experiments it is structured: for modular multiplication, residual errors concentrate on inputs with high popcount product; for parity, residual errors show seed-dependent Hamming-weight structure. Compared with a matched Bernoulli baseline at the same error rate, these deviations are substantial and actionable: a one-line residual-risk score yields a 3–10$\times$ spread between low-risk and high-risk failure rates on `modmul`. In short, certificates identify when a compositional rule is recovered on the bulk of inputs; residual diagnostics identify the input regimes where compositional generalization breaks down.

Contributions. The mathematics underlying our checks is classical (Reed–Muller codes, Walsh–Hadamard analysis, the parity-bias formula); our contribution is to assemble it into a workflow for evaluating compositional Boolean recovery and to characterize the regime in which it operates. **(1)** *Compositional-recovery benchmark.* We formulate "did the network recover the compositional rule" as ranked template matching of the induced behavior against a small bank of plausible compositional candidates, with margins reported to alternatives and Reed-radius certificates applied when the top candidate has known ANF degree. **(2)** *Failure mode.* Exact symbolic ANF recovery is unstable under small residual Hamming noise: the Möbius transform can amplify a small δ into maximum-degree, random-scale apparent algebraic complexity, even when the induced behavior remains inside the Reed–Muller decoding basin of the target. Symbolic equality therefore conflates behavioral compositional recovery with zero-error symbolic recovery, rejecting any nonzero residual. **(3)** *Stable lenses.* For candidates of known ANF degree, Reed-radius gives a coding-theoretic recovery certificate; Walsh-spectrum agreement gives a continuous spectral measurement of behavioral distance; both are stable under residual Hamming noise. **(4)** *Empirical results.* Across 25 task/output cells at $n = 24$ and three seeds on parity, modular addition, and modular multiplication, the correct compositional rule ranks top-1 in 23/23/22 cells per seed, with $22/25$ robustly top-1 across all three seeds (mean margin $\sim$0.41). Of the rest, two are output bits the network never recovers (correctly rejected in every seed) and one is the cliff-edge bit, certified exactly in the seeds where the network learns it. **(5)** *Residual diagnostics.* Relative to the intended rule, the residual $e = f \oplus h$ separates structured trained-network residuals from matched Bernoulli baselines: high-residual-risk inputs (e.g., high popcount product for `modmul`) predict where the reference rule fails, identifying localized exception regimes relative to a close intended reference rule.

Scope. This is a verification-and-diagnostics framework, not unsupervised discovery: it takes a candidate compositional rule (or a small bank of candidates) as input and certifies behavioral, not mechanism-level, agreement via a coding-theoretic (Reed–Muller) certificate, with the spectral (Walsh) distance quantifying the same behavioral error. We evaluate on parity, modular addition, and modular multiplication at $n \leq 24$, on ReLU MLPs and a small decoder-only transformer; extensions to GF($p > 2$) and automated residual feature discovery are future work.

2. Boolean Compositional Recovery

Setup. A trained network on n Boolean inputs induces a function $f : \{0,1\}^n \to \{0,1\}$. For $n \leq 24$ we enumerate the full input cube; a fraction α of inputs ($\alpha = 0.5$ for the $n = 16$ contrast and the parity scaling, $\alpha = 0.25$ for the $n = 24$ template-bank and residual experiments; full details in Section I) forms the training set and the remainder is held out for evaluation. A compositional hypothesis names a candidate rule $h : \{0,1\}^n \to \{0,1\}$, the *template*. Verification

asks whether f matches h in a sense that tolerates small residual error.

Three notions of recovery. "f matches h" admits three distinct claims, which coincide only at $\delta = 0$ and disagree sharply otherwise:

- *Behavioral recovery:* $\Pr_x[f(x) \neq h(x)] = \delta$ for some small δ.
- *Decoding recovery:* f lies in the unique-decoding ball of h in a Reed–Muller code (§ 3).
- *Symbolic recovery:* exact representation match, e.g. $\widehat{f}_{\mathrm{ANF}}(S) = \widehat{h}_{\mathrm{ANF}}(S)$ for all $S \subseteq [n]$ in the ANF basis.

This paper centers on the regime where behavioral and decoding recovery hold while symbolic recovery fails by orders of magnitude.

Channel-like decomposition. We use a channel-like framing $f = h \oplus e$, where the *residual mask* $e \in \{0,1\}^{2^n}$ indicates inputs on which network and candidate disagree. We do not assume the residual is i.i.d. Bernoulli (§ 6 shows it is structured). Verification becomes a classical decoding question: does f lie in the unique-decoding ball of h, and if so, what does e look like? § 3–5 develop the decoding side; § 6 treats e as a first-class object.

3. Stable Certificates and Brittle Symbolic Recovery

We use three structural representations of f, with very different stability properties under small Hamming noise.

Walsh–Hadamard spectrum (stable measurement). For this paragraph we switch to the standard $\{\pm 1\}$ encoding (where Boolean $0 \mapsto +1$, $1 \mapsto -1$); the rest of the paper uses $\{0,1\}$. The Walsh expansion $f = \sum_S \widehat{f}(S)\chi_S$ on $\{\pm 1\}^n$, with $\chi_S(z) = \prod_{i \in S} z_i$, is the DFT on the Boolean cube; "frequencies" are subsets $S \subseteq [n]$ and basis functions are parity-over-S characters. Walsh stability is exactly Parseval: for $\{\pm 1\}$-valued f, h,

$$\sum_S \big(\widehat{f}(S) - \widehat{h}(S)\big)^2 \;=\; \mathbb{E}_x\big[(f(x) - h(x))^2\big] \;=\; 4\delta, \quad (1)$$

where $\delta = \Pr_x[f(x) \neq h(x)]$. The squared L_2 deviation of induced from target Walsh spectra equals 4δ exactly, a continuous behavioral measurement, no decoding required (Appendix A).

Reed–Muller decoding radius (stable certificate). The Reed–Muller code $\mathrm{RM}(r,n)$ is the set of truth tables of all GF(2) polynomials in n variables of degree $\leq r$ (MacWilliams & Sloane, 1977). Two distinct codewords differ in at least 2^{n-r} inputs, so a received word within $2^{n-r-1} - 1$ flips of one is closer to it than to any other. Given a candidate h of ANF degree r, we accept f as a recovery iff

$$d_H(f,h) \;\leq\; 2^{n-r-1} - 1 \qquad (r < n). \quad (2)$$

The general unique-decoding radius is $\lfloor (2^{n-r} - 1)/2 \rfloor$, which equals $2^{n-r-1} - 1$ for $r < n$ and 0 at $r = n$; all our candidates have $r < n$. Within this radius h is the *unique* degree-$\leq r$ codeword: no other rule of degree $\leq r$ is also within $2^{n-r-1} - 1$ flips of f, so the certificate distinguishes recovery of h from being merely close to some other low-degree rule (Appendix B).

ANF / Möbius (unstable exact baseline). The ANF is the unique GF(2)-polynomial expansion $f(x) = \bigoplus_S \widehat{f}_{\mathrm{ANF}}(S) \prod_{i \in S} x_i$ on $\{0,1\}^n$, with coefficients computed by the Möbius transform $\widehat{f}_{\mathrm{ANF}}(S) = \bigoplus_{T \subseteq S} f(T)$. *ANF equality* ($\widehat{f}_{\mathrm{ANF}}(S) = \widehat{h}_{\mathrm{ANF}}(S)$ for all S) is the precise exact symbolic baseline this paper stress-tests. The instability comes from Möbius non-locality: altering any single truth-table value flips up to 2^n ANF coefficients, so small Hamming noise is amplified into the high-degree ANF strata.

Walsh measures, Reed certifies. Walsh agreement reports the continuous behavioral error 4δ; Reed-radius thresholds the same error against the unique-decoding radius for a degree-r candidate. The two lenses are consistent but serve different roles: Walsh gives magnitude, Reed gives an accept/reject certificate. As a scalar, Walsh L_2 agreement is a reparameterization of Hamming error ($\|\widehat{f} - \widehat{h}\|_2^2 = 4\delta = 4(1 - \mathrm{acc})$ by Parseval), so ranking candidates by Walsh distance coincides with ranking by accuracy; its added value is the spectral, coefficientwise view, not independent statistical evidence. ANF equality is the outlier: it requires $\delta = 0$ exactly.

4. The Möbius Noise Zone

We call a (network, target) cell a *Möbius noise zone* cell if the induced behavior lies inside the Reed–Muller decoding basin of the target ($d_H \leq 2^{n-r-1} - 1$) but ANF equality rejects ($\widehat{f}_{\mathrm{ANF}}(S) \neq \widehat{h}_{\mathrm{ANF}}(S)$ for some S). Empirically, this regime appears repeatedly across our trained networks, tasks, and scales.

The contrast at $n = 16$. Table 1 reports five trained-MLP cells at $n = 16$. For the `parity_16` cell, the same trained

Table 1. Trained ReLU MLPs at $n = 16$, $\alpha = 0.5$, seed 0. `modadd` / `modmul` use $k = 8$, so $n_{\text{in}} = 16$. Walsh 4δ measures behavioral error; Reed accepts iff $d_H \le 2^{n-r-1} - 1$, where r is the *clean* candidate's ANF degree (the "Ind." columns report the network's *induced* degree and support); ANF equality accepts iff $d_H = 0$. Reed-radius accepts $4/5$ cells, ANF equality only the two lossless cells.

Target	Bit acc.	Ind. ANF deg	Ind. supp.	Bit errs	RM radius	4δ	ANF eq?	Reed?
`parity_16`	99.31%	**16**	**24,998**	455	16,383	0.028	no	yes
`modadd_8` bit 0	100.00%	1	2	0	16,383	0	yes	yes
`modadd_8` bit 7	99.99%	**16**	**737**	7	127	4.3×10^{-4}	no	yes
`modmul_8` bit 0	100.00%	2	1	0	8,191	0	yes	yes
`modmul_8` bit 7	70.47%	**16**	**26,653**	19,353	31	1.18	no	no

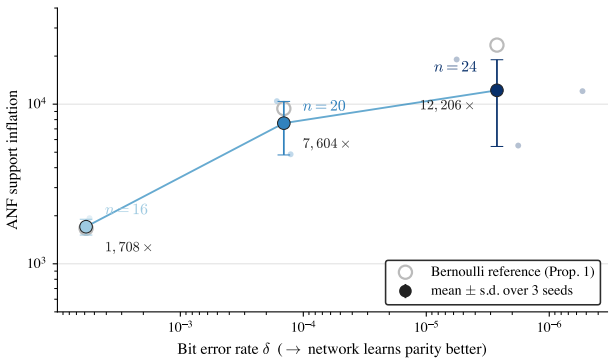


Figure 1. Trained ReLU MLPs on `parity_n` at $n \in \{16, 20, 24\}$, three seeds per scale. As n grows ($16 \to 20 \to 24$), residual error decreases while raw ANF inflation increases (mean factors $1{,}708 \to 7{,}604 \to 12{,}206$); the trend mixes dimension with δ and is not a controlled fixed-n noise sweep. All 9 trained networks lie inside the $\mathrm{RM}(1, n)$ decoding basin. Open circles are the Bernoulli reference from Proposition 4.1.

network gives four different readings. **Truth table:** 99.31% correct (455 errors out of 65,536). **Walsh:** a single coefficient at $\chi_{[n]} \approx 0.987$, the rest at the noise floor ($\le 2.76\%$ spectral-energy leakage outside $\chi_{[n]}$). **Reed:** 455 errors comfortably within the $\mathrm{RM}(1, 16)$ radius of 16,383, so decoding returns the clean codeword. **ANF:** 24,998 monomials at degree 16, a $\sim 1{,}560\times$ inflation. Walsh and Reed identify the same low-error parity solution; ANF equality alone rejects it.

The pattern extends across RM strata: `modadd_8` bit 7 (radius $2^7 - 1 = 127$, only 7 errors) decodes cleanly despite 737 induced monomials; `modmul_8` bit 7 (radius 31, 19,353 errors) is three orders of magnitude out of basin, and Reed correctly rejects. The same MLP gives opposite verdicts at low vs. high RM order, and the Reed certificate tracks basin membership while the Walsh measurement tracks the underlying error, where ANF equality is fooled.

Scaling: error rate down, ANF inflation up. As n grows, residual error shrinks but ANF inflation grows robustly across seeds (Figure 1). All nine trained networks at $n \in \{16, 20, 24\}$ lie inside the $\mathrm{RM}(1, n)$ basin (radius $2^{n-2} - 1$), with mean inflation factors $1{,}708 \to 7{,}604 \to 12{,}206$. The Reed margin and the ANF-based apparent complexity move in opposite directions: as n grows, the network's bit-error rate falls deeper inside the decoding ball while the induced ANF support climbs further into the random-noise range.

Mechanism (informal). A single-input bit-flip at a weight-j input changes 2^{n-j} ANF coefficients (up to 2^n at the origin), so even small δ can flip a large fraction of high-order ANF coefficients. The phenomenon depends only on the Möbius transform and Hamming noise, not on the specific task; `parity`, `modadd`, and `modmul` are illustrative substrates rather than the source of the effect. We make this quantitative as Proposition 4.1 (Appendix C).

Proposition 4.1 (informal; full statement in appendix)**.** *Let $f = h \oplus e$ with $e \sim \mathrm{Bernoulli}(\delta)^{2^n}$ i.i.d. Then for every $S \subseteq [n]$, $\Pr[\widehat{f}_{\mathrm{ANF}}(S) \neq \widehat{h}_{\mathrm{ANF}}(S)] = p_{|S|}$, where $p_k = (1 - (1 - 2\delta)^{2^k})/2$. The expected number of corrupted ANF coefficients is $\Omega(2^n)$ for any fixed $\delta > 0$ as $n \to \infty$.*

The proposition is a reference model: it predicts the shape of ANF inflation under i.i.d. Bernoulli noise, but trained residuals are not Bernoulli (§ 6), and δ varies with n. We use it as a benchmark prediction, not a tight characterization of trained-network behavior.

Architecture transfer. The phenomenon is not MLP-specific. We train a small decoder-only causal transformer (which we refer to as *TinyTransformer*; 2 layers, $d_{\text{model}} = 128$, 4 heads, GELU, pre-LN, $\sim$ 400K parameters, a configuration matching small-transformer testbeds in (Power et al., 2022; Liu et al., 2022; Nanda et al., 2023)) on `modadd_8` and `modmul_8` for three seeds each.[1] The same noise-zone signature appears (Figure 2): low-order `modadd` / `modmul` bits recover with clean ANF; high-order bits inflate to dense, random-scale ANF support; `modmul` cliff bits sit out of basin with $\sim$ 25,000-monomial ANFs, matching the MLP's

[1] We exclude `parity_n` as a design choice: high-dimensional parity is known to be hard for standard self-attention, both in expressivity (Hahn, 2020) and learning dynamics/sensitivity (Bhattamishra et al., 2023). These are asymptotic/empirical hardness results, not a sharp n threshold; since the phenomenon is about *trained-network* representations, we focus on architectures that learn the rule.

26,653.

5. Template-Bank Verification

In practical settings the target is rarely known with certainty as a single function. A practitioner may have a small bank of plausible candidate rules $\mathcal{H} = \{h_1, \ldots, h_m\}$ (correct candidates plus near-misses) and wants to know which best explains the trained network's induced behavior. We extend the known-target check to ranked template matching.

Algorithm. Given the induced behavior f and a finite candidate bank $\mathcal{H} = \{h_1, \ldots, h_m\}$:

1. For each h_j, compute the behavioral error: the exact held-out error $\hat{\delta}_{\text{test}}(h_j)$ when 2^n is enumerable, else the sampled uniform-cube error $\hat{\delta}_{\text{cube}}(h_j) \approx \hat{\delta}_{\text{test}}$ (§ 7). The Reed certificate below instead uses the full-cube Hamming distance $d_H(f, h_j)$.
2. Sort $\mathcal{H}$ ascending in $\hat{\delta}$. Let $h_{(1)}$, $h_{(2)}$ be the top two candidates. Report margin $\hat{\delta}(h_{(2)}) - \hat{\delta}(h_{(1)})$.
3. If $h_{(1)} \in \mathrm{RM}(r, n)$ for known r, report Reed margin $2^{n-r-1} - 1 - d_H(f, h_{(1)})$.
4. Accept $h_{(1)}$ as the behaviorally recovered rule iff *all three* conditions hold:
   (a) *Decisive margin.* In the exhaustive setting, $\hat{\delta}_{\text{test}}(h_{(2)}) - \hat{\delta}_{\text{test}}(h_{(1)}) \geq \tau$ for a fixed $\tau = 0.01$ (one percentage point of the cube). In the sampled setting the top two are data-dependent, so we use a bank-wide simultaneous bound at confidence $1 - \alpha_{\text{conf}}$ ($\alpha_{\text{conf}} = 0.05$): with $\epsilon_{\mathcal{H}} = \sqrt{\log(2|\mathcal{H}|/\alpha_{\text{conf}})/(2m)}$, all $|\mathcal{H}|$ candidate intervals hold simultaneously with probability $\geq 1 - \alpha_{\text{conf}}$, so the data-dependent top two may be compared validly, and we require the margin to exceed $2\epsilon_{\mathcal{H}}$ (Appendix D).
   (b) *Certificate.* When $h_{(1)} \in \mathrm{RM}(r, n)$ for known r, it is Reed-certified: $d_H(f, h_{(1)}) \leq 2^{n-r-1} - 1$ on the full cube, or, when 2^n is not enumerable, by the sampled upper/lower confidence-bound rule of Appendix D.
   (c) *Small behavioral error.* The applicable behavioral-error estimate for $h_{(1)}$ (held-out $\hat{\delta}_{\text{test}}$ when 2^n is enumerable, otherwise the uniform-cube upper confidence bound of Appendix D) is at most $\delta_{\max}$ ($\delta_{\max} = 0.05$). The degree-1 Reed radius is wide (almost $\frac{1}{4}$), so (b) alone gives only *decoding* recovery (the unique closest low-degree rule); this gate is what makes the verdict *behavioral* rather than a network sitting in a wide ball at high error.

   Otherwise abstain. If $h_{(1)}$ is a degenerate candidate (a constant), report that no nontrivial compositional rule is recovered rather than certifying the constant as the recovered rule.

A decisive margin alone is not sufficient: a top candidate with high absolute error abstains (condition (c)) even if it is Reed-certified, since a wide low-degree radius does not imply small behavioral error. In our $n = 24$ cells the empirical margins are either ~ 0.4 (correct cell, far above τ) or produced by a constant winning a non-recovered bit, so the $\tau = 0.01$ threshold cleanly separates recovery from non-recovery; the threshold matters only near the cliff edge.

Empirical results. We construct candidate banks per task (full specification in Appendix G): for `parity_n`, the correct parity plus subsets, complement, shifted bits, majority, and constants; for `modadd` / `modmul` output bit i, the correct bit plus shifted bits, no-carry approximations, single partial products, and complement. Across 25 (task, output bit) cells at $n = 24$ (`parity_24`, `modadd_12` all bits, `modmul_12` all bits) and three seeds, the correct candidate ranks top-1 in 23, 23, and 22 of 25 cells (seeds 0/1/2); $22/25$ cells are top-1-correct in *every* seed, with mean margin ~ 0.41 over those cells (Figure 3 shows seed 0; per-seed breakdown in Appendix G). The cells that are not robustly correct are all `modmul_12` high bits: bits 10 and 11 are rejected in every seed (the network never learns them), and bit 9 is the seed-dependent cliff edge, ranked correct in the two seeds where the network learns it and rejected ("constant 0" wins) in the third. In each rejection the network has collapsed toward majority-class prediction (a class-imbalance shortcut, not a learned compositional rule), and the verifier's refusal to certify the intended rule is the correct verdict. This is the desired behavior: the verifier tracks, per seed and per bit, whether that particular network behaviorally recovered the rule, rather than reporting a single aggregate that a clean target exists for.

6. Reference-Relative Residual Analysis

A certificate tells us *whether* a candidate approximates the network's induced behavior up to residual error. Relative to a reference candidate h (whether Reed-certified or merely close), the residual mask $e = f \oplus h$ tells us *where* that reference explanation fails. The `modmul` cliff cells we analyze below are close to the intended rule but *not* Reed-certified; we use them because the certified cells in our data carry too few residual events (< 100) to expose interpretable structure, whereas the cliff cells supply thousands. This is reference-relative failure analysis, not localization inside a certified recovery. We treat e as an error process and ask two questions: is the residual rate predictable from interpretable input features, and can we use that structure to construct stress tests that amplify the network's failure rate?

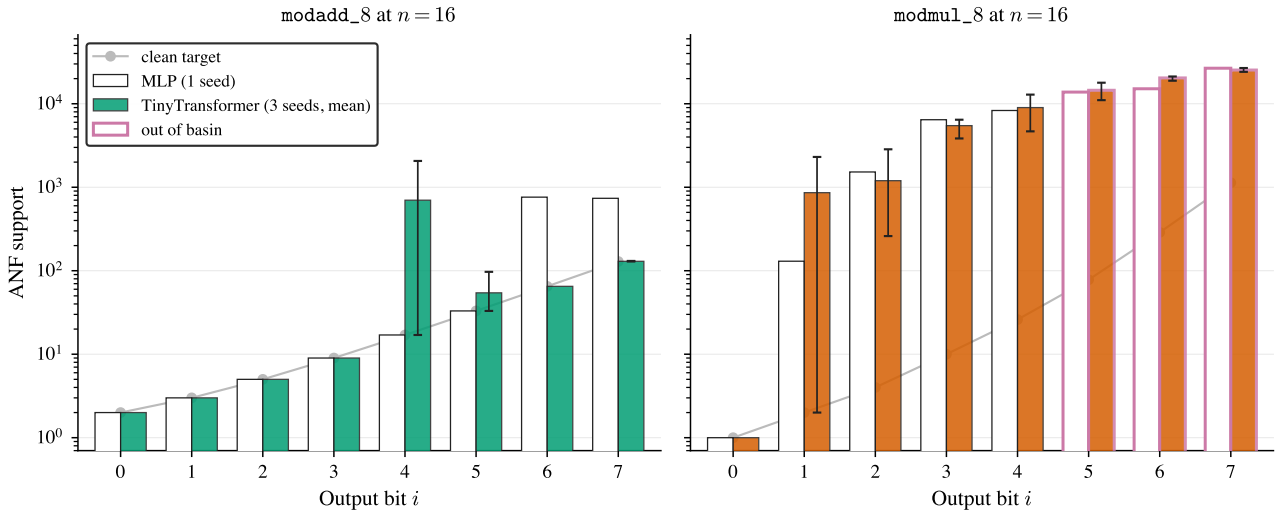


Figure 2. Architecture transfer at $n = 16$: per-output ANF support for MLP (open bars, one seed) and TinyTransformer (filled bars, mean over three seeds, whiskers show min/max) on `modadd_8` (left) and `modmul_8` (right). Grey line traces the clean target ANF support per bit. Out-of-basin bars (Hamming distance from clean target exceeds Reed radius; for the Transformer, at least one seed out) carry an accent edge. Both architectures show the same noise-zone signature.

Bernoulli baseline. For each cell with global error rate δ, we compare empirical conditional residual rates $\Pr[e(x) = 1 \mid z(x)]$ for various input covariates z (full per-task list in Appendix F) to a matched Bernoulli baseline $b \sim \text{Bernoulli}(\delta)$. Under Bernoulli, the conditional rate is uniformly δ (independent of z); deviation from this flat conditional rate indicates residual structure.

`modmul_12` cliff bit. The cliff-edge bit on `modmul_12` is seed-dependent (bits 8/9/7 at $\delta \in [7{\times}10^{-4}, 6{\times}10^{-3}]$), but conditioning on the popcount product $\text{pop}(a) \cdot \text{pop}(b)$ (max $= 144$) produces the same qualitative bell curve in all three seeds (Figure 4): error rate suppressed near the extremes, sharply elevated at mid-range, with peak/baseline ratios 1.9/2.7/2.3$\times$ and several bins outside the Bernoulli $\pm 2\sigma$ band. The drop at high $\text{pop}(a) \cdot \text{pop}(b)$ is partly a sampling artifact: random (a, b) concentrate $\text{pop}(a) \cdot \text{pop}(b)$ near 36, so high-pp bins have far fewer samples and the Bernoulli $\pm 2\sigma$ band widens accordingly: the empirical rate there is not distinguishable from baseline.

`parity_24` weight-dependent residuals. The parity residuals are tiny (89, 212, 73 errors out of $\sim$ 12.6M held-out each; δ_{held} between 6×10^{-6} and 2×10^{-5}; Figure 5). Against the held-out weight baseline (84.6% of inputs already fall in $w \in [5, 14]$), the in-band error share is *not* a robust concentration: it is 100% (seed 0, elevated), 89% (seed 2, mildly elevated), and 72% (seed 1, *below* baseline). The even/odd asymmetry is likewise seed-specific: seed 0 is 5.2$\times$ even-biased, seeds 1 and 2 odd-biased (0.5$\times$ and 0.4$\times$). The extreme-weight bands ($w \leq 3$ or $w \geq 20$) host only $\sim$11,400 held-out inputs and see zero errors, which is unremarkable given <250 total errors. We therefore report the parity weight structure as seed-dependent rather than robust; the residual structure we rely on is the `modmul` partial-product effect below, whose *direction* is robust across seeds.

Residual-guided stress tests. Structured residuals are actionable: they let us select inputs that disproportionately fail. On the `modmul_12` cliff bit (per seed: bit 8 / 9 / 7), sorting held-out inputs by popcount product and binning by quantile thresholds (Figure 6), the high-risk group fails at 3–10$\times$ the rate of the low-risk group across seeds, corresponding to 1.4–1.9$\times$ the random-baseline rate. The risk score is computed directly from the input. The high-risk group (pp $\geq Q_{80}$), which ties make $\approx$24.5% of held-out inputs rather than an exact 20%, captures 34.3% / 44.6% / 47.4% of the residual errors per seed, a 1.40/1.82/1.94$\times$ enrichment over its input share (i.e. the rate ratios above). This operationalizes the residual diagnostic: for a sufficiently close reference candidate with enough residual events, the same analysis predicts where that reference explanation will break, whether or not the candidate is itself Reed-certified.

Where residual analysis applies. Residual structure is informative when the reference candidate is close enough that residual events reflect the network's failure modes rather than gross candidate mismatch, and numerous enough to be interpretable. Cells in clean recovery ($\delta = 0$, e.g. `modadd_12`) have nothing to analyze. The `modmul` cliff cells we use are close to the intended rule but not Reed-

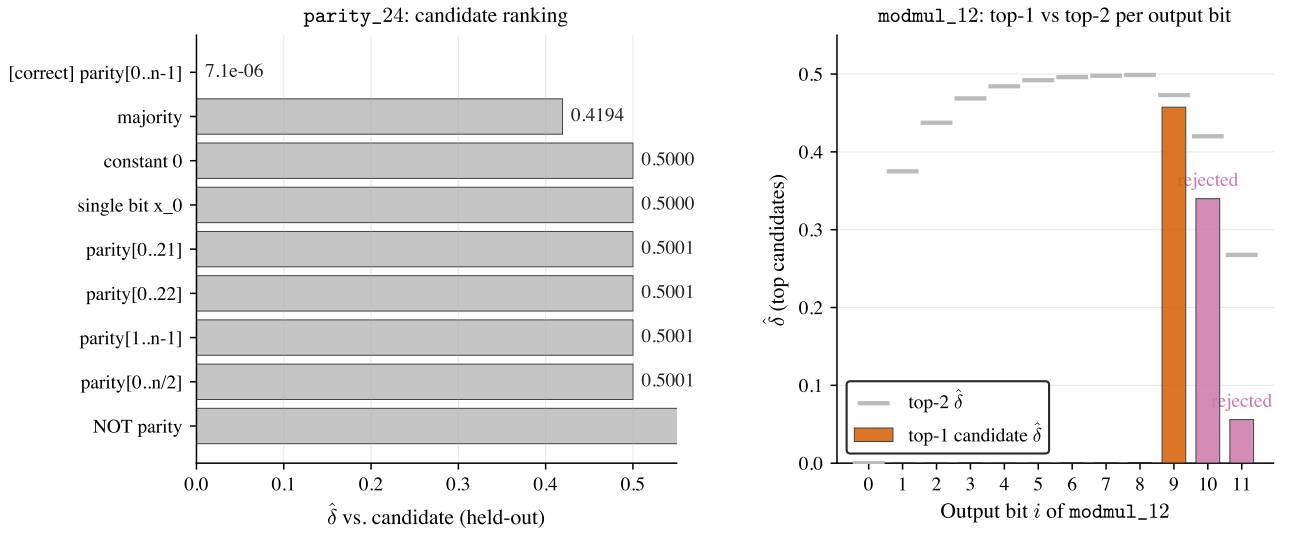


Figure 3. Template-bank ranking. **Left:** `parity_24` candidate ranking (correct in blue; alternatives in grey). Correct sits at $\hat{\delta} = 7 \times 10^{-6}$, the next-best (majority) at 0.42, margin ~ 0.42. **Right:** `modmul_12` per-output-bit top-1 vs. top-2. Bits 0–9 rank correct first with substantial margin; bits 10–11 (where the network failed to learn) are correctly *rejected*: "constant 0" explains the network better than the intended bit, and the framework refuses the false acceptance.

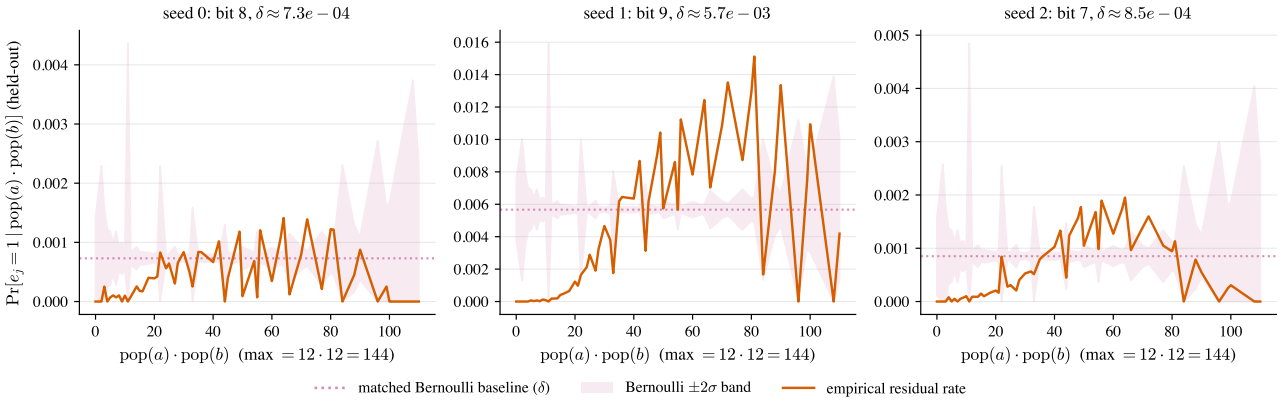


Figure 4. Residual structure on `modmul_12` cliff bit across 3 seeds. The specific cliff bit varies (seed 0: bit 8; seed 1: bit 9; seed 2: bit 7), but in every case the conditional error rate is sharply peaked at mid-range popcount product $\mathrm{pop}(a) \cdot \mathrm{pop}(b)$ (max $144 = 12 \cdot 12$) and falls to zero at the extremes. Several bins per seed fall outside the Bernoulli $\pm 2\sigma$ band; the systematic shape is absent from the matched baseline.

certified, precisely the regime with enough structured residual events to localize failures. Cells *far* from every candidate (residuals dominated by mismatch) should instead be reassigned to a different top template (§ 5) or rejected.

7. Scaling and Candidate Generation

Sample-based verification at large n. At input dimensions where exhaustive truth-table extraction is infeasible ($n \gtrsim 24$), step 1 of the template-bank algorithm is replaced by uniform-sample estimation: $\hat{\delta} = (1/m) \sum_{i=1}^{m} \mathbf{1}[f(x_i) \neq h(x_i)]$ for m samples drawn uniformly from $\{0,1\}^n$. Hoeffding's inequality (Hoeffding, 1963) bounds the verdict-error probability in $m = O(\log(1/p)/\eta^2)$ samples, where η is the desired precision around the fractional Reed threshold 2^{-r-1} (Appendix D). Sampling uniformly from the cube (rather than from the training set) makes $\hat{\delta}$ a behavioral quantity, and in the large-n regime that motivates sampling it guards against pure memorization: when the training set is a vanishing fraction of the cube ($|\mathrm{train}|/2^n \to 0$, as for any poly-size training set at large n), uniform draws are almost all held-out, so a memorizer's chance-level held-out behavior dominates $\hat{\delta}$ and the certificate rejects. At the constant training fractions used for the exhaustive $n \leq 24$ experiments ($\alpha = 0.5$ at $n = 16$, 0.25 at $n = 24$) this guard is weaker (a fraction

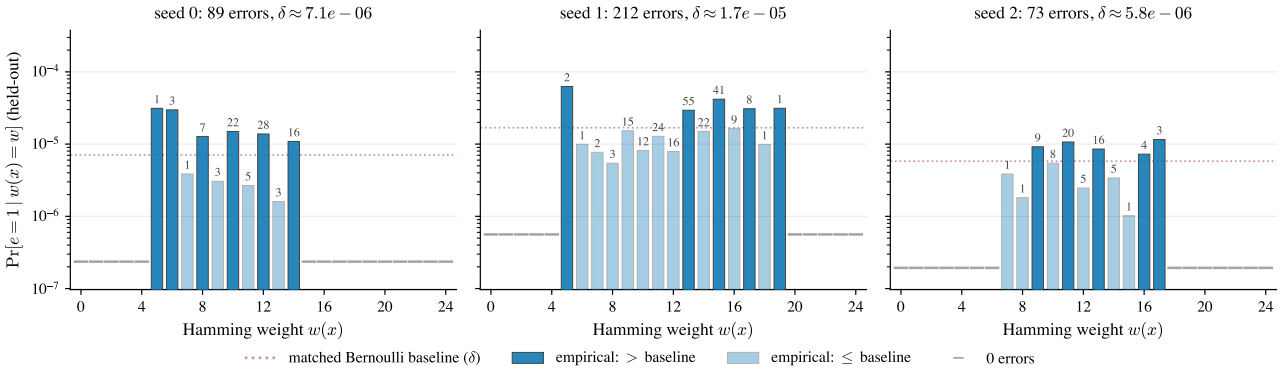


Figure 5. Residual structure on `parity_24` across 3 seeds (89 / 212 / 73 errors out of $\sim$ 12.6M held-out inputs each). In-band ($w \in [5, 14]$) error share is 100% / 72% / 89% against an 84.6% held-out baseline, so the band is elevated for seeds 0 and 2 but *depleted* for seed 1, not a robust concentration. The even-vs-odd asymmetry direction is also *not* robust: seed 0 is 5.2× even-biased while seeds 1 and 2 are odd-biased (0.5×, 0.4×). Annotations show raw error counts per bin.

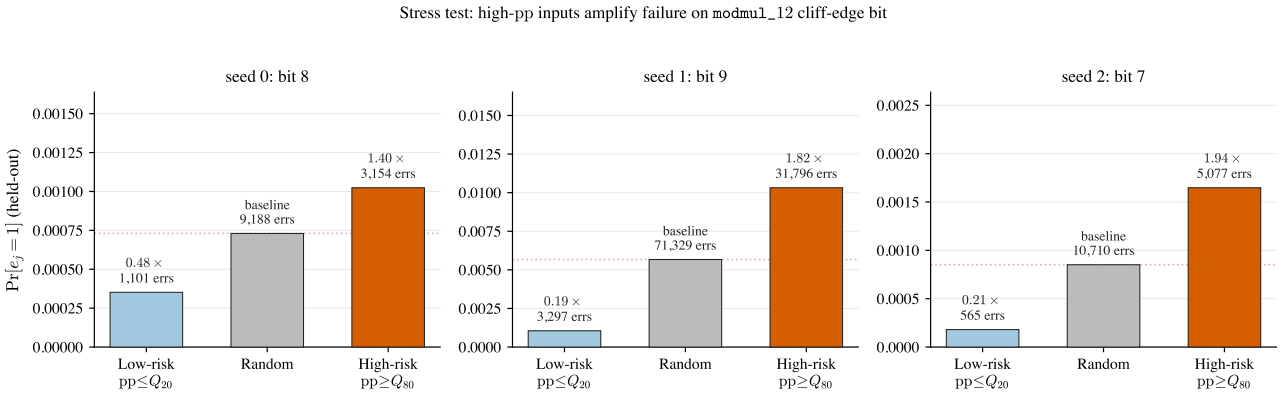


Figure 6. Stress test on `modmul_12` cliff bit across 3 seeds (bit 8 / 9 / 7). Held-out inputs split by popcount-product quantile thresholds (ties make the tails ≈24.5%, not exact quintiles): low-risk (pp $\leq Q_{20}$), random, high-risk (pp $\geq Q_{80}$). High-risk inputs fail at 1.40 / 1.82 / 1.94× the baseline rate; low-risk at 0.48 / 0.19 / 0.21×. The direction of the effect (high-pp amplifies, low-pp suppresses) is robust across seeds.

$\approx\alpha$ of uniform draws are memorized training points), so there we compute δ on the held-out split directly rather than by cube sampling (Appendix D).

Walsh-sparse candidate generation via KM. When no candidate is named but the target is suspected Walsh-sparse (e.g. a small parity-character class), Kushilevitz–Mansour recovery (Kushilevitz & Mansour, 1993; Goldreich & Levin, 1989) proposes candidates in $\mathrm{poly}(n, 1/\theta)$ queries: the recovered heavy set $\mathcal{H}^\star$ defines $\hat{f}_{\mathcal{H}^\star} = \mathrm{sign}(\sum_{S\in\mathcal{H}^\star} \widehat{f}(S)\chi_S)$, whose ANF degree sets the Reed-radius parameter (Walsh and ANF degree differ, with parity at Walsh degree n but ANF degree 1). At enumerable n the degree is read off the cube; at large n this requires an analytically known degree, as for the single parity character of the $n = 40$ demonstration (ANF degree 1 by inspection), and is not available for an arbitrary sparse candidate. A bias-corrected mass estimator (Appendix E) preserves the $\mathrm{poly}(n, 1/\theta)$ scaling. We verify the full loop end-to-end (Appendix H): at one fixed threshold KM proposes a Reed-certified candidate for Walsh-sparse parity (validated on a trained net at $n = 24$, and on a noisy oracle at $n = 40$), while graded (`modadd`) and diffuse (`modmul` cliff) bits correctly fall back or abstain. The targets here are rarely Walsh-sparse, so KM is a convenience for that regime; the general path is verification of *named* candidates, whose Reed certificate and Walsh distance require no Walsh sparsity (§3).

8. Implications for Compositionality Evaluation

Three regimes the workflow distinguishes. For a candidate compositional rule h on input dimensions $n \lesssim 24$ (or larger via uniform-sample $\hat{\delta}$), the verification + residual workflow separates three regimes that accuracy alone or

symbolic equality conflate:

1. *Compositional recovery.* h ranks top-1 in the template bank with significant margin and is Reed-certified, $d_H(f, h) \leq 2^{n-r-1} - 1$ (correspondingly, the Walsh distance $\sqrt{4\delta}$ is small). The network's behavior matches the compositional rule on the bulk of inputs even when symbolic ANF equality fails.
2. *Localized exception regimes.* h is the intended reference rule and the residual $e = f \oplus h$ is structured: a small input regime (e.g., high popcount-product) carries a disproportionate share of the failures, and a one-line risk score amplifies that failure rate at test time. The network agrees with the intended rule on most inputs and fails on a predictable subset, suggestive of shortcut-flavored or memorization-flavored exceptions. We use these descriptors informally: the residual identifies predictable exception regions not captured by the reference rule, but our experiments do not directly distinguish literal training-set memorization from other shortcut-like behavior such as margin failures or missing learned partial-product structure.
3. *No compositional recovery.* The candidate fails to win the bank, wins by a small margin, or wins with high absolute error and no Reed certificate. This is the case our two unrecovered `modmul_12` high bits land in: the verifier correctly refuses to certify h and ranks "constant 0" above the intended rule, because the network has not behaviorally recovered the rule on those output bits.

Three checks, three different mistakes. Held-out behavioral accuracy alone conflates regime 1 with regime 2: the residual structure is invisible to bulk accuracy, so a network that is compositional only on the easy subset of inputs can score similarly to one that is compositional everywhere. Exact symbolic equality makes the opposite mistake: it demands zero-error symbolic recovery and rejects regime 1 even when the recovered rule is certified behaviorally inside the Reed–Muller decoding basin. The lenses here decouple the three: the Reed-radius certifies behavioral recovery up to small residual and the Walsh distance quantifies it; ANF equality tests strict symbolic recovery; residual conditioning locates the exception regimes.

9. Related Work

Compositional generalization on algorithmic tasks. Whether trained networks recover the underlying compositional rule on parity, modular arithmetic, and related Boolean tasks (as opposed to fitting a high-error shortcut) is a recurring question. Grokking work (Power et al., 2022; Liu et al., 2022; Nanda et al., 2023; Furuta et al., 2024) traces phase transitions from memorization to behavioral recovery on modular-arithmetic transformers via cyclic-group Fourier representations; we ask the dual question on the input side, using Walsh–Hadamard on $\{0,1\}^n$ and ANF on the induced function. Two-layer results on staircase and parity (Abbe et al., 2022; Barak et al., 2022) establish when SGD provably learns the rule; we focus on post-training residual structure. Bhattamishra et al. (2023) characterize transformer learnability through *sensitivity*, consistent with our observation that some trained Boolean networks can stay within the Reed decoding basin under residual error. Hahn (2020) establishes parity expressivity limits, informing our exclusion of `parity_`n from architecture transfer. Tian (2024) characterizes the algebraic structure of optimal solutions on modular arithmetic; we instead provide noise-tolerant certificates that a trained network has recovered such a solution.

Boolean Fourier, coding, and circuit discovery. Reed–Muller codes and majority-logic decoding are classical (MacWilliams & Sloane, 1977; Carlet, 2010), used here functionally (network as noisy received word). Walsh–Hadamard analysis is textbook (O'Donnell, 2014); KM (Kushilevitz & Mansour, 1993; Goldreich & Levin, 1989) provides $\mathrm{poly}(n, 1/\theta)$-query scalability. Kang et al. (2024) recover sparse Möbius transforms in $\mathcal{O}(Kn)$ samples when $K \ll 2^n$; gradient-trained networks induce dense Möbius even when the target is sparse, so we run KM on the Walsh side and fall back to template banks when Walsh is diffuse. Mech-interp circuit discovery (Hanna et al., 2023; Conmy et al., 2023; Elhage et al., 2021) is complementary; we add a formal behavioral verification layer.

Property testing, self-correction, and low-degree testing. Our certificates sit downstream of a classical line on testing and correcting proximity to algebraic codes. Low-degree and Reed–Muller testing (Rubinfeld & Sudan, 1996; Alon et al., 2005) decide whether a function is close to *some* low-degree polynomial; the BLR linearity test and self-correction (Blum et al., 1993) locally correct a function near a codeword; and learning polynomials in the highly noisy regime (Goldreich et al., 2000) reconstructs a sparse polynomial from corrupted evaluations. We use these ideas in a narrower, verification-oriented way: rather than testing membership in a code class or reconstructing an unknown polynomial, we are handed a *named* candidate rule h and ask whether a trained network's induced behavior lies in h's unique-decoding ball (a single Hamming-distance comparison), then treat the residual against the reference candidate as a first-class diagnostic. Our contribution is this workflow and its empirical characterization on trained networks, not new testing theory: the unique-decoding radius and the parity-bias formula are classical.

Use of Large Language Models

Large language models were used to assist with code, numerical verification, and LaTeX editing. All experimental results, mathematical claims, and final text were checked and verified by the author, who takes full responsibility for the content.

References

Abbe, E., Boix-Adserà, E., and Misiakiewicz, T. The merged-staircase property: A necessary and nearly sufficient condition for SGD learning of sparse functions on two-layer neural networks. In *Conference on Learning Theory*, pp. 4782–4887, 2022.

Alon, N., Kaufman, T., Krivelevich, M., Litsyn, S., and Ron, D. Testing Reed–Muller codes. *IEEE Transactions on Information Theory*, 51(11):4032–4039, 2005.

Barak, B., Edelman, B. L., Goel, S., Kakade, S. M., Malach, E., and Zhang, C. Hidden progress in deep learning: SGD learns parities near the computational limit. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Bhattamishra, S., Patel, A., Kanade, V., and Blunsom, P. Simplicity bias in transformers and their ability to learn sparse Boolean functions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5767–5791, 2023.

Blum, M., Luby, M., and Rubinfeld, R. Self-testing/correcting with applications to numerical problems. *Journal of Computer and System Sciences*, 47(3): 549–595, 1993. The BLR linearity test, which uses the same parity-bias formula on $\mathbb{F}_2^n$.

Carlet, C. Boolean functions for cryptography and error-correcting codes. In Crama, Y. and Hammer, P. L. (eds.), *Boolean Models and Methods in Mathematics, Computer Science, and Engineering*, pp. 257–397. Cambridge University Press, 2010.

Conmy, A., Mavor-Parker, A. N., Lynch, A., Heimersheim, S., and Garriga-Alonso, A. Towards automated circuit discovery for mechanistic interpretability. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Elhage, N., Nanda, N., Olsson, C., Henighan, T., Joseph, N., Mann, B., Askell, A., Bai, Y., Chen, A., Conerly, T., et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. URL https://transformer-circuits.pub/2021/framework/index.html.

Furuta, H., Minegishi, G., Iwasawa, Y., and Matsuo, Y. Towards empirical interpretation of internal circuits and properties in grokked transformers on modular polynomials. *Transactions on Machine Learning Research*, 2024.

Goldreich, O. and Levin, L. A. A hard-core predicate for all one-way functions. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing (STOC)*, pp. 25–32, 1989.

Goldreich, O., Rubinfeld, R., and Sudan, M. Learning polynomials with queries: The highly noisy case. *SIAM Journal on Discrete Mathematics*, 13(4):535–570, 2000.

Hahn, M. Theoretical limitations of self-attention in neural sequence models. *Transactions of the Association for Computational Linguistics*, 8:156–171, 2020.

Hanna, M., Liu, O., and Variengien, A. How does GPT-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Hoeffding, W. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

Kang, J. S., Erginbas, Y. E., Butler, L., Pedarsani, R., and Ramchandran, K. Learning to understand: Identifying interactions via the Möbius transform. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Kushilevitz, E. and Mansour, Y. Learning decision trees using the fourier spectrum. *SIAM Journal on Computing*, 22(6):1331–1348, 1993.

Liu, Z., Kitouni, O., Nolte, N., Michaud, E. J., Tegmark, M., and Williams, M. Towards understanding grokking: An effective theory of representation learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

MacWilliams, F. J. and Sloane, N. J. A. *The Theory of Error-Correcting Codes*. North-Holland, 1977.

Nanda, N., Chan, L., Lieberum, T., Smith, J., and Steinhardt, J. Progress measures for grokking via mechanistic interpretability. In *International Conference on Learning Representations*, 2023.

O'Donnell, R. *Analysis of Boolean Functions*. Cambridge University Press, 2014.

Power, A., Burda, Y., Edwards, H., Babuschkin, I., and Misra, V. Grokking: Generalization beyond overfitting on small algorithmic datasets, 2022.

Rubinfeld, R. and Sudan, M. Robust characterizations of polynomials with applications to program testing. *SIAM Journal on Computing*, 25(2):252–271, 1996.

Tian, Y. Crafting global optimizers to reasoning tasks via algebraic objects in neural nets. In *NeurIPS Workshop on Compositional Learning*, 2024.

A. Walsh–Hadamard Analysis on the Boolean Cube

This appendix collects the Walsh–Hadamard machinery used in the body. The goal is for a reader unfamiliar with Boolean Fourier to understand why Walsh-spectrum agreement is an *exact* continuous behavioral measurement: the squared L_2 deviation between the induced Walsh spectrum and a candidate's Walsh spectrum equals 4δ, where δ is the bit-error rate. This identity is what lets us replace exact symbolic equality (which fails dramatically under small residual error) with a noise-stable behavioral test for compositional recovery.

The $\{\pm 1\}$ encoding. Throughout the body we work on the Boolean cube $\{0,1\}^n$, but Walsh–Hadamard analysis is cleanest with the $\{\pm 1\}$ encoding $z_i = (-1)^{x_i}$. A Boolean function $f : \{0,1\}^n \to \{0,1\}$ becomes a sign-valued function $\tilde{f} : \{\pm 1\}^n \to \{\pm 1\}$ via $\tilde{f}(z) = (-1)^{f(x)}$. We will write f for $\tilde{f}$ in this section to keep notation simple.

Characters. For each subset $S \subseteq [n] = \{1, \ldots, n\}$, the character $\chi_S : \{\pm 1\}^n \to \{\pm 1\}$ is

$$\chi_S(z) \;=\; \prod_{i \in S} z_i.$$

Each character is a parity-of-S function: $\chi_S(z) = +1$ iff an even number of coordinates indexed by S are -1. There are exactly 2^n characters, indexed by subsets of $[n]$.

Orthonormality. Equip the space of real-valued functions on $\{\pm 1\}^n$ with the inner product

$$\langle f, g \rangle \;=\; \mathbb{E}_z[f(z)g(z)] \;=\; \frac{1}{2^n} \sum_{z \in \{\pm 1\}^n} f(z)g(z).$$

Under uniform z, the characters satisfy $\langle \chi_S, \chi_T \rangle = \mathbf{1}[S = T]$. Proof: $\chi_S(z)\chi_T(z) = \chi_{S \triangle T}(z)$, and $\mathbb{E}_z[\chi_U(z)] = \prod_{i \in U} \mathbb{E}[z_i] = 0$ for any nonempty U (by independence and $\mathbb{E}[z_i] = 0$); when $S = T$ we get $U = \emptyset$ and $\chi_\emptyset \equiv 1$. So the 2^n characters form an orthonormal basis for the space of functions on the cube.

Walsh expansion and Walsh coefficients. Every function $f : \{\pm 1\}^n \to \mathbb{R}$ admits a unique expansion

$$f(z) \;=\; \sum_{S \subseteq [n]} \widehat{f}(S)\, \chi_S(z), \qquad \widehat{f}(S) \;=\; \mathbb{E}_z[f(z)\chi_S(z)]. \tag{3}$$

The coefficients $\widehat{f}(S)$ are the *Walsh–Hadamard coefficients* of f. Because the basis is orthonormal, computing $\widehat{f}(S)$ is the correlation of f with the parity-of-S character.

Parseval. Orthonormality gives Parseval's identity:

$$\sum_{S \subseteq [n]} \widehat{f}(S)^2 \;=\; \mathbb{E}_z[f(z)^2]. \tag{4}$$

For $\{\pm 1\}$-valued f this becomes $\sum_S \widehat{f}(S)^2 = 1$.

The 4δ identity. This is the identity used in the body. Let $f, h : \{\pm 1\}^n \to \{\pm 1\}$ be Boolean (sign-valued), and let $\delta = \Pr_z[f(z) \neq h(z)]$ be the bit-error rate (equivalently, the fraction of inputs on which the network disagrees with the candidate compositional rule). Apply Parseval to $f - h$:

$$\begin{aligned} \sum_S \big(\widehat{f}(S) - \widehat{h}(S)\big)^2 \;&=\; \mathbb{E}_z\big[(f(z) - h(z))^2\big] \\ &=\; \mathbb{E}_z[\,4 \cdot \mathbf{1}\{f(z) \neq h(z)\}\,] \\ &=\; 4\delta, \end{aligned}$$

where the second line uses $(f - h)^2 \in \{0, 4\}$ since $f, h \in \{\pm 1\}$. So the squared L_2 deviation between the two Walsh spectra is exactly four times the bit-error rate. Walsh distance is therefore a continuous, exact measurement of behavioral disagreement; small residual training error produces small Walsh-spectrum displacement, no decoding required.

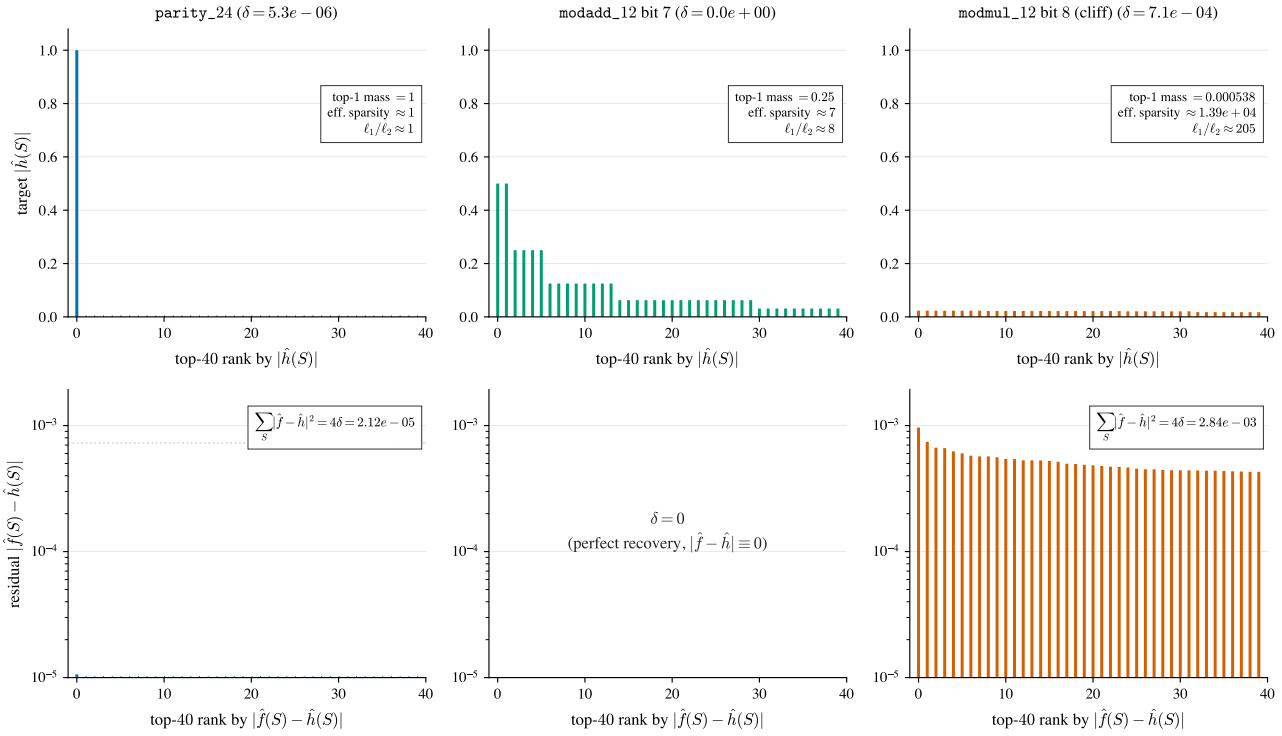


Figure 7. Walsh spectra under $\{\pm 1\}$ encoding, computed from full 2^{24}-entry truth tables of the trained network and the clean target. **Top row:** target spectra $|\widehat{h}(S)|$, top-40 ranked by magnitude, showing three distinct concentration regimes. `parity_24` is perfectly sparse (one coefficient at $S = [n]$, top-1 mass $= 1$); `modadd_12` bit 7 is graded (top-1 mass $= 0.25$, effective sparsity ≈ 7); the `modmul_12` cliff bit (bit 8, seed 0) is diffuse, with energy spread across $\sim 14{,}000$ effective coefficients, exactly the regime where KM-based candidate generation cannot help and the framework falls back to template-bank ranking. **Bottom row:** residual spectra $|\widehat{f}(S) - \widehat{h}(S)|$, top-40 ranked by residual magnitude (log scale, floor 10^{-5}). For `parity_24`, $4\delta \approx 2.1 \times 10^{-5}$ is spread across 2^{24} coefficients, so almost all individual residuals fall below the visible floor; `modadd_12` bit 7 has $\delta = 0$ (perfect recovery, residual identically zero); the `modmul_12` cliff bit shows visible structured residual at every plotted rank, consistent with the conditional-rate amplification documented in §6. Annotations report top-1 mass $\max_S |\widehat{h}(S)|^2$, effective sparsity $1/\sum_S |\widehat{h}(S)|^4$, and ℓ_1/ℓ_2 ratio $\sum_S |\widehat{h}(S)| / \sqrt{\sum_S |\widehat{h}(S)|^2}$ for the target spectrum, and total residual mass $\sum_S |\widehat{f} - \widehat{h}|^2 = 4\delta$ in the residual panels.

Consequence for compositional recovery. If the candidate compositional rule h is Walsh-sparse (parity, for example, has $\widehat{h}(S) = \mathbf{1}[S = [n]]$), its known signature makes the L_2 distance $\|\widehat{f} - \widehat{h}\|_2$ cheap to read off, and by Parseval this distance is $\sqrt{4\delta}$: a small Walsh distance means a small behavioral error δ against h. Walsh agreement is therefore a natural, stable way to *measure* behavioral closeness to any candidate with a known Walsh signature (the accept/reject decision remains the Reed-radius certificate of §3), and unlike ANF equality (§C) it does not amplify small residual error into a dense, random-scale ANF.

B. Reed–Muller Codes and the Decoding Radius

The Reed–Muller decoding radius $2^{n-r-1} - 1$ is the quantity the body uses as a coding-theoretic certificate of compositional recovery: if the network's induced truth table is within that many bit-flips of a candidate of ANF degree r, it is closer to that candidate than to any other codeword in $\mathrm{RM}(r, n)$. This appendix derives the radius and notes how it is used.

The code. The Reed–Muller code $\mathrm{RM}(r, n)$ is the set of truth tables of all GF(2) polynomials in n variables of degree at most r. Concretely, fix the 2^n inputs $x \in \{0,1\}^n$ in lexicographic order; each polynomial of degree $\leq r$ produces a codeword $c \in \{0,1\}^{2^n}$ by evaluation. The dimension of $\mathrm{RM}(r, n)$ is $\sum_{k=0}^{r} \binom{n}{k}$ (the number of distinct monomials of degree $\leq r$).

Minimum distance. A central result (MacWilliams & Sloane, 1977): the minimum Hamming distance between distinct codewords of $\mathrm{RM}(r, n)$ is exactly 2^{n-r}. Sketch: any nonzero degree-$\leq r$ polynomial has Hamming weight at least 2^{n-r}; applying this to the difference of two codewords (itself a degree-$\leq r$ polynomial) gives minimum distance 2^{n-r}.

Unique decoding radius. A standard fact for any binary code with minimum distance d: the unique-decoding radius is $\lfloor (d-1)/2 \rfloor$. For $\mathrm{RM}(r, n)$ with $d = 2^{n-r}$, this is $\lfloor (2^{n-r} - 1)/2 \rfloor = 2^{n-r-1} - 1$ for $r < n$. Therefore, given a received word f (here, the network's induced truth table) and a candidate codeword h of ANF degree $\leq r$, the inequality

$$d_H(f, h) \;\leq\; 2^{n-r-1} - 1$$

implies h is the unique closest codeword in $\mathrm{RM}(r, n)$ to f.

Use in the body. We treat the trained network's induced truth table as a received word and the candidate compositional rule as a codeword. If the rule has known ANF degree r and the Hamming distance from the network's induced truth table to the rule is below 2^{n-r-1}, the network is provably closer to the candidate than to any other rule in $\mathrm{RM}(r, n)$, a coding-theoretic certificate of behavioral compositional recovery, regardless of what the network's apparent ANF expansion looks like in the symbolic basis.

Radius shrinkage at high degree. The radius decays geometrically with r: at $n = 24$, a degree-1 candidate (parity) has radius $2^{22} - 1 \approx 4.2 \times 10^6$; a degree-8 candidate has radius $2^{15} - 1 \approx 3.3 \times 10^4$. So the certificate is informative for low-ANF-degree compositional rules and tight for high-degree ones. Modular arithmetic bits sit at varying ANF degrees (low bits low-degree, high bits maximum-degree), and Reed-radius certificates apply correspondingly.

C. The Möbius Transform and the Parity-Bias Formula

This appendix gives the Möbius transform on $\{0, 1\}^n$ and derives Proposition 4.1 of the body: under small Bernoulli residual noise, the expected ANF mass at each subset is pushed toward $1/2$, so the expected number of nonzero ANF monomials approaches $\sim 2^{n-1}$, half the maximum. This is why exact symbolic ANF equality is brittle: behavioral compositional recovery up to small δ is consistent with dense, random-scale apparent algebraic complexity.

The Möbius transform / ANF coefficient map. Every Boolean function $f : \{0, 1\}^n \to \{0, 1\}$ has a unique algebraic normal form (ANF) representation as a polynomial over GF(2):

$$f(x) \;=\; \bigoplus_{S \subseteq [n]} \widehat{f}_{\mathrm{ANF}}(S) \prod_{i \in S} x_i, \qquad \widehat{f}_{\mathrm{ANF}}(S) \in \{0, 1\}.$$

The map $f \mapsto \widehat{f}_{\mathrm{ANF}}$ is the *Möbius transform*, and it is given by Möbius inversion on the Boolean lattice:

$$\widehat{f}_{\mathrm{ANF}}(S) \;=\; \bigoplus_{T \subseteq S} f(T), \qquad f(S) \;=\; \bigoplus_{T \subseteq S} \widehat{f}_{\mathrm{ANF}}(T), \tag{5}$$

where we identify $S \subseteq [n]$ with its indicator vector. The ANF basis is sparse for compositional rules with low GF(2) algebraic structure: parity $x_1 \oplus \cdots \oplus x_n$ has ANF support $\{\{1\}, \ldots, \{n\}\}$ (n degree-1 monomials, ANF degree 1), modular addition's i-th output bit has support on monomials of degree at most $i + 1$, and so on. (Parity's *Walsh* support, by contrast, is the single coefficient at $S = [n]$; the Möbius and Walsh transforms place parity's mass in very different strata (ANF degree 1 versus Walsh degree n), which is why the Reed-radius parameter is read from the ANF side, not the Walsh side, in §7.)

Bernoulli residual model. Suppose the network's induced function is $f = h \oplus e$, where h is the clean candidate compositional rule and $e \in \{0, 1\}^{2^n}$ is the residual mask, drawn as $e(x) \overset{\text{i.i.d.}}{\sim} \mathrm{Bernoulli}(\delta)$ across inputs. The Bernoulli model is a reference model for ANF instability: it shows that even unstructured residual noise at small rate can produce near-random high-degree ANF support. The trained residuals we find empirically are structured (§ 6), so this model is a benchmark prediction rather than a worst-case characterization.

Expected ANF mass under Bernoulli noise. By Möbius inversion (5), at any subset $S \subseteq [n]$,

$$\widehat{f}_{\mathrm{ANF}}(S) \;=\; \bigoplus_{T\subseteq S} f(T) \;=\; \bigoplus_{T\subseteq S} (h(T)\oplus e(T)) \;=\; \widehat{h}_{\mathrm{ANF}}(S) \;\oplus\; \bigoplus_{T\subseteq S} e(T).$$

Let $E_S = \bigoplus_{T\subseteq S} e(T)$; this is the parity of $2^{|S|}$ i.i.d. Bernoulli(δ) random variables. The probability that a sum of m i.i.d. Bernoulli(δ) variables has odd parity is

$$\Pr\big[\textstyle\bigoplus_{i=1}^{m} b_i = 1\big] \;=\; \frac{1-(1-2\delta)^m}{2}. \tag{6}$$

This is the parity-bias formula: it is a one-line calculation by induction on m, using the fact that $\Pr[b_1 \oplus b_2 = 1] = 2\delta(1-\delta)$ and $1 - 2\cdot 2\delta(1-\delta) = (1-2\delta)^2$. With $m = 2^{|S|}$:

$$\Pr\big[E_S = 1\big] \;=\; \frac{1-(1-2\delta)^{2^{|S|}}}{2}. \tag{7}$$

Consequence: ANF mass approaches $1/2$. For any subset S with $|S| \geq \log_2(1/\delta) + O(1)$, $(1-2\delta)^{2^{|S|}}$ is exponentially small, so $\Pr[E_S = 1] \to 1/2$. The ANF coefficient at such an S is therefore equally likely to be flipped as not flipped relative to the clean candidate's ANF. Summing the per-coefficient flip probabilities $\Pr[E_S = 1] = p_{|S|}$ over all S gives, *by linearity of expectation*, the expected number of *nonzero* coefficients of f, $\mathbb{E}[\,|\mathrm{supp}\,\widehat{f}_{\mathrm{ANF}}|\,] = \sum_S \big[\widehat{h}_{\mathrm{ANF}}(S) + (1 - 2\widehat{h}_{\mathrm{ANF}}(S))\, p_{|S|}\big]$, which for a sparse candidate ($|\mathrm{supp}\,\widehat{h}_{\mathrm{ANF}}|$ small, as for parity) is $\sum_S p_{|S|} + O(|\mathrm{supp}\,\widehat{h}_{\mathrm{ANF}}|)$, i.e. $\sum_S p_{|S|}$ to leading order. Since there are $\binom{n}{k}$ subsets of size k, most subsets have size near $n/2$, and $p_k \to 1/2$ for $k \gtrsim \log_2(1/\delta)$, this expectation approaches 2^{n-1} (half of the maximum) under any nonvanishing δ. We claim only this expectation: the events $\{E_S = 1\}$ at different S are *not* independent (the E_S share underlying residual entries $e(T)$), so we make no concentration statement about the support count itself.

Numerical example matching the body. At $n = 16, \delta = 0.0069$ (the 99.31%-accurate parity network), $(1-2\delta)^{2^{|S|}}$ is already negligible by $|S| \geq 8$: $\log(1-2\delta) \approx -0.014$, so for $|S| = 8$ we have $(1-2\delta)^{256} \approx e^{-3.6} \approx 0.027$, giving $\Pr[E_S = 1] \approx 0.49$. For $|S| \geq 10$ the flip probability is essentially $1/2$. The expected ANF support of f is therefore dominated by the $\sum_{k=8}^{16}\binom{16}{k} \approx 2^{15}$ medium-to-large subsets, each contributing flip probability $\approx 1/2$; the empirical 24,998 monomials are consistent with the expected support $\sum_S p_{|S|}$ predicted by (7). (As above, this is a statement about the expectation only; the per-coefficient flips are correlated, so we do not assert a concentration bound around 24,998.)

Why this matters for compositional recovery. A network can be very close in Hamming distance to a clean compositional rule and still have an ANF whose support appears random-like. Symbolic ANF equality *rejects* such networks. The Reed radius *accepts* them and the Walsh distance quantifies the small deviation: the Hamming distance and the *squared* Walsh L_2 deviation both scale linearly in δ (the L_2 distance itself scales as $2\sqrt{\delta}$), rather than blowing up under the Möbius transform.

D. Sample-Based Verification at Large n

For input dimensions where exhaustive truth-table extraction is infeasible, the body replaces step 1 of the template-bank algorithm (compute $\delta = \Pr_x[f(x) \neq h(x)]$) with the uniform-sample estimator $\hat{\delta} = \frac{1}{m}\sum_{i=1}^{m} \mathbf{1}[f(x_i) \neq h(x_i)]$, where $x_1, \ldots, x_m$ are drawn i.i.d. uniformly from $\{0,1\}^n$. The Bernoulli indicators $\mathbf{1}[f(x_i) \neq h(x_i)] \in \{0,1\}$ satisfy the conditions of Hoeffding's inequality (Hoeffding, 1963), which gives

$$\Pr\big[\,|\hat{\delta} - \delta| > \eta\,\big] \;\leq\; 2\exp(-2m\eta^2)$$

for any $\eta > 0$. To reach confidence $1-p$ that $|\hat{\delta} - \delta| \leq \eta$, it therefore suffices to draw

$$m \;\geq\; \frac{\log(2/p)}{2\eta^2} \;=\; O\!\left(\frac{\log(1/p)}{\eta^2}\right)$$

samples, independently of the input dimension n.

Choosing η. The Reed-radius certificate accepts iff $d_H(f,h) \le 2^{n-r-1}-1$, equivalently $\delta \le 2^{-r-1}-2^{-n}$. For verdict stability, η should be a fraction of the fractional Reed threshold 2^{-r-1}: setting $\eta = \frac{1}{2}\cdot 2^{-r-1}$ gives a verdict error iff $\hat{\delta}$ falls on the wrong side of the threshold by more than half its width, and $m = O(2^{2r}\log(1/p))$ suffices. The bound is degree-dependent, not dimension-dependent: high-degree candidates (large r) need exponentially more samples in r, but there is no penalty for large n at fixed r. This presumes a promise gap: the verdict is stable only when the true δ is at least η from the threshold $2^{-r-1}-2^{-n}$. In the sampled regime we therefore certify by the Hoeffding *upper* confidence bound ($\hat{\delta}+\eta \le 2^{-r-1}-2^{-n}$ to accept, $\hat{\delta}-\eta > 2^{-r-1}-2^{-n}$ to reject) and abstain in the boundary band, rather than thresholding the point estimate $\hat{\delta}$.

Memorization and the training fraction. Sampling uniformly from the cube (rather than from the training set) guards against pure memorization *in the large-n regime where sampling is used*: when the training set is a vanishing fraction of the cube (as for any poly-size training set at large n), uniform draws are almost entirely held-out, so a memorizer (high training accuracy, chance-level held-out behavior) yields $\hat{\delta}$ near $1/2$ and the certificate rejects. The guarantee is asymptotic in $|\text{train}|/2^n \to 0$. At a constant training fraction α, cube sampling mixes the two splits, $\delta_{\text{cube}} = \alpha\,\delta_{\text{train}} + (1-\alpha)\,\delta_{\text{test}}$, so it sits below the held-out error by $\approx\alpha(\delta_{\text{test}}-\delta_{\text{train}})$; a degree-1 candidate has fractional threshold $2^{-2}=0.25$, so at $\alpha=0.5$ a pure memorizer (held-out error $\frac{1}{2}$) gives $\hat{\delta}\approx 0.25$, asymptotically at the $1/4$ boundary (the finite-n threshold is $1/4-2^{-n}$). For this reason, in the exhaustive $n \le 24$ experiments we estimate δ on the held-out split directly; cube-sampling is the large-n tool, where the vanishing-fraction condition holds.

E. KM Bias Correction

The Kushilevitz–Mansour (KM) algorithm (Kushilevitz & Mansour, 1993; Goldreich & Levin, 1989) recovers the heavy Walsh coefficients of a Boolean function in $\text{poly}(n, 1/\theta)$ *membership queries*: it queries the network at *chosen* inputs, which a trained model supplies. The body uses KM to generate candidates for the template bank when the input dimension is too large for full truth-table extraction. This appendix records the bias correction we use to make KM's squared-mass estimates unbiased at finite query budgets.

Setup. KM is a recursive search over Walsh-coefficient prefixes. At a prefix fixing membership for the first d coordinates, it estimates the *prefix mass*

$$M(\text{prefix}) = \sum_{S \text{ extending prefix}} \widehat{f}(S)^2 = \mathbb{E}_z\big[g(z)^2\big], \quad g(z) = \mathbb{E}_y\big[\chi_{\text{prefix}}(y)\, f(y,z)\big],$$

where $g(z)$ is a partial Walsh projection over a split (y,z) of the input (this uses correlated membership queries sharing z). KM prunes the subtree when $M < \theta^2/2$, recurses on both children, and at a leaf estimates a single coefficient $\widehat{f}(S) = \mathbb{E}_x[f(x)\chi_S(x)]$. Both the per-$z$ projection $\hat{g}(z)$ and the leaf $\hat{g}(S)$ are sample means; KM thresholds on their *squares*, which carry a finite-sample bias.

The bias. $\hat{g}(z)^2$ is a biased estimator of $g(z)^2$:

$$\mathbb{E}[\hat{g}(z)^2] \;=\; g(z)^2 + \text{Var}[\hat{g}(z)] \;=\; g(z)^2 + \frac{1-g(z)^2}{m},$$

using that the per-sample variance is $1-g(z)^2$ for $\{\pm 1\}$-valued products. At small sample sizes the $1/m$ term inflates apparent mass, producing spurious candidates that pass KM's threshold despite having near-zero true mass.

Unbiased mass estimator. Solving for $g(z)^2$ gives

$$\widetilde{g}(z)^2 \;=\; \frac{m\,\hat{g}(z)^2 - 1}{m-1}. \tag{8}$$

A short calculation confirms $\mathbb{E}[\widetilde{g}(z)^2] = g(z)^2$. We apply this correction to both the prefix-mass projections $g(z)^2$ and the leaf coefficients $\widehat{f}(S)^2$ before thresholding, which preserves the $\text{poly}(n, 1/\theta)$ scaling at query budgets a practitioner can afford. When the spectrum is diffuse (no heavy coefficients survive thresholding under $\widetilde{g}$), the KM stage abstains and the template-bank verifier handles practitioner-proposed candidate rules directly.

F. Residual Covariates

The body conditions the residual error rate $\Pr[e(x) = 1]$ on a small set of human-designed covariates of x. This appendix lists the covariates used for each task. The choice is task-specific: each covariate corresponds to a natural notion of input complexity for the intended compositional rule.

Parity (`parity_`n). The only covariate is the input Hamming weight $w(x) = \sum_{i=1}^{n} x_i$. Parity at $n = 24$ has 25 weight bins ($w = 0, 1, \ldots, 24$), with bin sizes $\binom{n}{w}$. The body reports seed-dependent Hamming-weight profiles and does *not* claim robust mid-weight concentration (the band $w \in [5, 14]$ already holds 84.6% of inputs).

Modular addition (`modadd_`k). Inputs are $(a, b) \in \{0,1\}^k \times \{0,1\}^k$ encoding two k-bit integers. The output bit at position i depends on a, b through the carry chain. We use:

- *Total carries:* the number of bit positions j at which the carry-out is 1 during $a + b \mod 2^k$.
- *Maximum carry-chain length:* the longest run of consecutive carry-out positions. Carry-chain length is the natural complexity proxy for modular addition: when no carries fire, the i-th output bit is just $a_i \oplus b_i$; long carry chains mean output bit i depends on many input bits below.

Carry features are computed by simulating the standard ripple-carry adder $c_{j+1} = (a_j \wedge b_j) \vee (a_j \oplus b_j) \wedge c_j$ with $c_0 = 0$.

Modular multiplication (`modmul_`k). Inputs are $(a, b) \in \{0,1\}^k \times \{0,1\}^k$. The relevant complexity measures are

- $\mathrm{popcount}(a)$ and $\mathrm{popcount}(b)$, the operand Hamming weights.
- $\mathrm{popcount}(a) \cdot \mathrm{popcount}(b)$, an operand-density proxy for partial-product load. It counts *all* nonzero (a_i, b_j) pairs in the $k \times k$ multiplication grid; this over-counts the partial products that actually affect a given output bit modulo 2^k (pairs with $i + j \geq k$ are discarded, and only pairs feeding that bit's column matter), so we use it as a correlate of partial-product load, not an exact count. The body finds the residual rate peaked at mid-range values of this proxy.

G. Template-Bank Construction

This appendix specifies the per-task candidate banks used in the body. Each bank contains the canonical compositional rule plus two kinds of alternatives:

- *Plausible failure modes*: functions a trained network could realistically converge to instead of the rule. These include constants (degenerate solutions exploiting class imbalance) and low-degree partial implementations (e.g. "no-carry" substitutes for `modadd`, single partial products for `modmul`).
- *Structural distractors*: near-misses not expected to survive to low training loss under correctly specified training, but which are included so that the correct rule's margin to alternatives is informative and as a pipeline sanity check. Off-by-one shifted bits and the complement $1 \oplus h$ fall in this class; the latter also catches label-inversion bugs.

Multi-seed outcome. We run the identical ranking procedure across three seeds at $n = 24$; no retraining is involved, since the ranking reads the saved residual masks. Table 2 summarizes. The correct rule ranks top-1 in 23/23/22 of 25 cells (seeds 0/1/2). Of the 25 cells, 22 are top-1-correct in *every* seed (mean margin 0.41), two (`modmul_12` bits 10 and 11) are rejected in every seed (the network never learns these high bits, and "constant 0" explains it better), and one (`modmul_12` bit 9) is the seed-dependent cliff edge. No `parity_24` or `modadd_12` cell ever fails. The seed-dependence of bit 9 mirrors the cliff-bit migration documented in §6 (the cliff sits at bit 8/9/7 for seeds 0/1/2): the verifier certifies recovery exactly on the bits each individual network learned, which is the intended behavior rather than a failure of the headline result.

Parity bank (`parity_`n). Single-output task. The bank contains: parity $x_1 \oplus \cdots \oplus x_n$; its complement $1 \oplus x_1 \oplus \cdots \oplus x_n$; parity over three strict subsets ($x_1 \oplus \cdots \oplus x_{n-1}$, $x_1 \oplus \cdots \oplus x_{n-2}$, and $x_1 \oplus \cdots \oplus x_{n/2}$); shifted parity $x_2 \oplus \cdots \oplus x_n$; majority over $[n]$; the single bit x_1; and constant 0. ANF degree is 1 for the parity-family candidates and the single bit, 0 for the constant, and 16 for majority at $n = 24$ (its nonzero ANF levels are 13–16); so the Reed-radius certificate is informative only for the low-degree candidates.

Table 2. Multi-seed template-bank outcome at $n = 24$ over 25 (task, output) cells. "Robust" = same verdict in all three seeds.

Outcome	Cells
Top-1 correct per seed (seeds 0/1/2)	23 / 23 / 22
Robustly top-1 correct (all seeds)	22
Robustly rejected (all seeds; `modmul` bits 10, 11)	2
Seed-dependent (cliff bit; `modmul` bit 9)	1
Mean margin over robustly-correct cells	0.41

Modular addition bank (`modadd`_k, per output bit i). The intended rule for output bit i is the bit-i function of $a + b \mod 2^k$, which has ANF degree $i + 1$. The bank contains:

- $a_i \oplus b_i$ (degree 1, the "no-carry" substitute);
- $a_i \oplus b_i \oplus (a_{i-1} \wedge b_{i-1})$ (degree 2, the "one-position carry" substitute);
- the full bit-i adder function (degree $i + 1$, the intended rule);
- shifted bits $a_{i\pm1} \oplus b_{i\pm1}$ (off-by-one wiring errors);
- the complement $1\oplus$ (intended rule);
- constant 0, constant 1.

Reed certificates apply at all degrees $\leq k$.

Modular multiplication bank (`modmul`_k, per output bit i). The intended rule for output bit i is the bit-i function of $a \cdot b \mod 2^k$. The Reed radius uses each candidate's ANF degree read off its truth table; for `modmul_12` these degrees are $2, 2, 4, 5, 6, 7, 8, 10, 12, 14, 16, 18$ for bits 0–11, exceeding k at the high bits (unlike modular addition, where bit i has degree $i{+}1$). The bank contains: the intended rule; the bit-i XOR partial-product truncation (degree 2); single partial products $a_p \wedge b_q$ for various (p, q); shifted bits $a_{i\pm1} \cdot b$-type variants (off-by-one wiring errors); the complement $1\oplus$ (intended rule); constant 0, constant 1. The two failures the verifier surfaces in the body (the unrecovered `modmul_12` high bits) are output bits where constant 0 ranks above the intended rule, correctly indicating that the network has not behaviorally recovered the rule on those bits. Because the reported margins depend on the exact bank, the precise enumerated candidates (including the (p, q) pairs and the boundary conventions at $i = 0$ and $i = k - 1$) are fixed by the bank-construction procedure and are available from the author.

H. End-to-End KM Candidate Generation and Verification

This appendix closes the KM loop of §7: with the target *unknown* to KM (membership-query access only, i.e. the network may be queried at chosen inputs but its rule is not given), KM proposes a heavy Walsh set $\mathcal{H}^\star$, we build $\hat{f}_{\mathcal{H}^\star} = \mathrm{sign}(\sum_{S\in\mathcal{H}^\star} \widehat{f}(S)\chi_S)$, read off its ANF degree r on the cube, and apply the Reed-radius certificate $d_H \leq 2^{n-r-1}-1$ via a sampled $\hat{\delta}$. We use a *single* fixed threshold $\theta = 0.3$ throughout, so the outcome is driven by the target's Walsh spectrum, not by per-task tuning.

Hidden sparse-parity discovery (where KM earns its keep). KM's non-trivial use is not confirming parity over all n bits (whose support is the obvious guess), but *discovering* an unknown sparse support: the noise-tolerant junta / learning-parity-with-noise setting. We test both a real trained network and large n (Table 3). **Trained net.** We train an MLP on a hidden 6-junta parity at $n = 24$ (parity over a secret size-6 subset, one of $\binom{24}{6} = 134{,}596$). The net learns it exactly ($\delta = 0$ held-out), and KM (given only membership-query access and *not* told the support) recovers the exact secret subset as its single heavy character, yielding a degree-1 candidate the certificate accepts. This is discovery from a real network, not confirmation of a known rule. **Scale.** Lacking a net that learns sparse parity at $n > 24$, we use a controlled noisy oracle (parity over a secret subset with a fixed Bernoulli(δ_0=0.01) residual). At $n = 40$ with secret size $k \in \{8, 12\}$, KM recovers the exact hidden support in ≈1.5M queries and certifies with sampled $\hat{\delta} \approx 0.01 \ll 2^{-2}$ from $m = 50{,}000$ samples independent of n, while a brute-force Walsh transform over $2^{40} \approx 10^{12}$ coefficients is infeasible. This is the one regime where KM adds value: noise-tolerant recovery of an *unknown* sparse support at a scale exhaustive methods cannot reach.

Table 3. Hidden sparse-parity discovery: KM recovers an unknown secret support (a k-junta), not parity over all bits. "Found" = recovered support equals the secret. Trained: an MLP on a secret 6-junta at $n = 24$; Oracle: noisy parity ($\delta_0 = 0.01$) over a secret subset.

Source	n	k	Found	deg	$\hat{\delta}$	Verdict
Trained MLP	24	6	yes	1	0.000	**accept**
Oracle	40	8	yes	1	0.011	**accept**
Oracle	40	12	yes	1	0.010	**accept**

The boundary: only Walsh-sparse targets. Crucially, the success in Table 3 is a property of the *target*, not of the pipeline. Running the *identical* closed loop on three trained-MLP cells at $n = 24$ (induced truth tables reconstructed from the saved residual masks; no retraining) gives sharply different outcomes (Table 4), tracking the Walsh concentration of Figure 7: `parity_24` is Walsh-sparse, so KM returns a single character, the candidate is degree-1 parity, and the certificate accepts; the graded `modadd_12` bit 7 and diffuse `modmul_12` cliff bit yield no certifiable compact candidate, and the framework correctly falls back or abstains. The reason is structural: a parity (even a hidden one) *is* a single Walsh character of ANF degree 1, exactly what KM recovers and the wide degree-1 Reed radius admits; `modadd` bit i has ANF degree $i+1$ with a long Walsh tail and `modmul` cliff bits are highly diffuse, so there is no sparse parity inside them to find. KM discovers sparse structure where it exists; it cannot manufacture it.

Table 4. End-to-end KM at a single fixed $\theta = 0.3$, $n = 24$. "Heavy" is the size of KM's recovered set; degree is the reconstructed candidate's ANF degree; $\hat{\delta}$ is the sampled bit-error rate of the candidate against the trained net.

Cell	Walsh	Heavy	deg	$\hat{\delta}$	Verdict
`parity_24`	sparse	1	1	0.000	**accept**
`modadd_12` b7	graded	2	2	0.250	fall back
`modmul_12` cliff	diffuse	0	n/a	n/a	abstain

Why magnitude-truncated reconstruction does not rescue the graded case. One might hope to recover the heaviest Walsh coefficients and Reed-certify the reconstruction. The exact spectrum of `modadd_12` bit 7 has 382 nonzero coefficients in tied magnitude levels (2 at $\frac{1}{2}$, 4 at $\frac{1}{4}$, …, 256 at $\frac{1}{128}$), so "top-k" is ill-defined when k cuts through a tie block; we therefore truncate by magnitude level (all or none of each level). Table 5 reports the cumulative truncations using the *exact* Walsh coefficients (no KM sampling noise, the best case). Each proper truncation captures more mass and lowers $\hat{\delta}$, but its ANF degree climbs in lockstep, so the Reed threshold 2^{-r-1} shrinks just as fast: every truncation *below* the full spectrum fails the certificate. Only including the final ($\frac{1}{128}$) level reconstructs the function exactly ($\hat{\delta} = 0$, degree 8), which then certifies trivially. Faithful certification of this graded target therefore requires essentially its *entire* spectrum: exactly what is infeasible to recover when the spectrum is not sparse. Notably the template bank (§5) *does* certify the `modadd` bit (and the recovered `modmul` bits) via the named candidate at its known degree, the trained net's true δ sitting inside the radius even though no sparse Walsh reconstruction does; the `modmul` cliff cell, by contrast, stays outside the Reed radius and is correctly rejected. The obstruction is candidate *generation* from a non-sparse spectrum, not verification. KM's reconstruction is in fact a strictly worse candidate than the named rule, so KM pays off only when the target is both unnameable in advance and Walsh-sparse, the hidden-parity regime above.

Table 5. Exact magnitude-level Walsh truncations of `modadd_12` bit 7 (all 382 nonzero coefficients lie in tied levels). Each row includes every coefficient down to the stated magnitude; k is the cumulative count. ANF degree and the Reed threshold 2^{-r-1} move against $\hat{\delta}$; only the full spectrum certifies, at $\hat{\delta} = 0$.

level $\leq$	k	ANF deg	$\hat{\delta}$	2^{-r-1}	cert.?
$1/2$	2	2	0.250	0.125	no
$1/4$	6	3	0.125	0.062	no
$1/8$	14	4	0.062	0.031	no
$1/16$	30	5	0.031	0.016	no
$1/32$	62	6	0.016	0.008	no
$1/64$	126	7	0.008	0.004	no
$1/128$	382	8	0.000	0.002	**yes**

Takeaway. KM is not a general candidate generator: it earns its keep only on Walsh-sparse targets, where it performs noise-tolerant discovery of an unknown sparse support (validated on a real trained net and at $n = 40$ beyond exhaustive reach, Table 3), and declines (falls back or abstains) on the graded and diffuse cases we test. For targets that are not sufficiently Walsh-sparse, the general path is verification of *named* candidates at their known ANF degree, whose Reed certificate and Walsh distance require no Walsh sparsity.

I. Experimental and Implementation Details

We record the configuration behind every table and figure. Per-run metadata (split sizes, width, convergence, error counts) is logged with each result.

ReLU MLP. Three hidden layers with ReLU, hidden width $\max(128, 16n)$ (so $256/320/384$ at $n = 16/20/24$), and a linear head producing one logit per output bit. Bits are fed in the $\{\pm 1\}$ encoding; the induced Boolean value is the logit thresholded at 0 (logit $> 0 \Rightarrow 1$). Training uses Adam (learning rate 10^{-3}) on `BCEWithLogits` loss for at most 15,000 epochs, with early stopping after 50 consecutive epochs at zero training-bit error. Cells with $n < 24$ train full-batch; the $n = 24$ cells use minibatches of 2^{18}. All cells use seeds $\{0, 1, 2\}$.

Train/held-out split. A fraction α of the 2^n cube is drawn without replacement for training (seed-dependent, so the split varies across seeds) and the remainder is held out. The $n = 16$ contrast (Table 1) and the parity scaling (Figure 1) use $\alpha = 0.5$; the $n = 24$ template-bank (§5) and residual (§6) experiments use $\alpha = 0.25$ ($n_{\text{train}} = 4{,}194{,}304$, $n_{\text{held}} = 12{,}582{,}912$). Held-out $\hat{\delta}$ is computed on the held-out split; Reed-radius, ANF, and Walsh quantities use the full 2^n truth table.

TinyTransformer. Decoder-only, causal, pre-LayerNorm, 2 blocks, $d_{\text{model}} = 128$, 4 heads ($d_{\text{head}} = 32$), GELU MLP with $d_{\text{ff}} = 4d_{\text{model}}$ ($\sim$400K parameters). Each input bit is one token (embedding over $\{0, 1\}$) plus a learned positional embedding; the output bit is read from the *last* token's hidden state through a linear head, thresholded at 0. Trained with Adam (10^{-3}) for at most 3,000 epochs (same early-stopping rule), $\alpha = 0.5$, seeds $\{0, 1, 2\}$, on `modadd_8` and `modmul_8`.

KM candidate generation. KM uses a single fixed magnitude threshold $\theta = 0.3$ throughout (§7, Section H); the candidate bit-error rate $\hat{\delta}$ is estimated from up to 200,000 uniform samples (50,000 for the $n = 40$ oracle). The hidden-parity experiments train an MLP on a secret k-junta ($n = 24$, $k = 6$) and run a noisy parity oracle ($n = 40$, $k \in \{8, 12\}$, with a fixed $\text{Bernoulli}(0.01)$ residual). Candidate banks are specified in Section G. Code and the saved per-seed residual masks that regenerate all tables and figures are available from the author.

J. Limitations

Our experiments use synthetic Boolean tasks (parity, modular addition, modular multiplication) at $n \leq 24$ on ReLU MLPs and a small transformer; a case study on a real trained circuit is the natural next step. The certificate reports *behavioral* agreement with a named rule, not internal implementation, and is complementary to causal localization. The Reed radius shrinks geometrically with ANF degree, so the certificate is most informative for low-degree candidates. Most importantly, *automated* candidate generation via KM applies only when the target is Walsh-sparse. Within that regime KM does genuine work (it recovers an unknown sparse-parity support on a real trained net and at $n = 40$ under noise, Appendix H), but the regime is narrow: the `modadd`/`modmul` targets we test are not sufficiently Walsh-sparse at the chosen threshold, and we show that no magnitude-truncated Walsh reconstruction below the full spectrum is Reed-certified for the graded target we examine. So in practice the framework relies on practitioner-supplied named candidates verified by the Reed certificate and Walsh distance, which require no Walsh sparsity, rather than on unsupervised discovery. The residual covariates we use are task-specific and human-designed; a general residual-feature-discovery procedure, and a GF($p > 2$)-native treatment, are future work.