Correct Chains, Wrong Answers: Dissociating Reasoning from Output in LLM Logic

Abinav Rao Sujan Rachuri Nikhil Vemuri

Abstract

We introduce the Novel Operator Test, a contamination-proof benchmark that separates operator logic from operator name: because operators use novel names and are defined solely by truth tables, models cannot rely on memorized associations, enabling rigorous distinction between genuine reasoning and pattern retrieval. Evaluating nine Boolean operators across depths 1–10 on five models (up to 8,100 problems each), we uncover a reasoning–output dissociation invisible to existing benchmarks: at Claude Sonnet 4's depth 7, all 31 errors have verifiably correct chain-of-thought yet wrong declared answers. Mechanistic probes reveal the model is 99.99% confident even when wrong and pre-commits to incorrect answers before generating correct reasoning. This differs from CoT unfaithfulness (Turpin et al., 2024), where reasoning diverges from the actual process; here, reasoning is correct yet the answer is wrong. An arithmetic domain replication (modular operators mod 5) confirms the novelty gap generalizes but shows lookup errors rather than dissociation, localizing the phenomenon as specific to Boolean semantics. The benchmark reveals two failure types: strategy failures at depth 2 (+62pp from scaffolding) and content failures at depth 7 (+8–30pp, 0/300 errors post-intervention).

1 Introduction

Chain-of-thought (CoT) reasoning is widely used to improve and verify LLM outputs on logical tasks (Wei et al., 2022). A natural assumption follows: if every reasoning step is correct, the final answer should be too. We show this assumption is false.

Existing logic benchmarks use standard connectives from pretraining data, confounding reasoning with retrieval. The Novel Operator Test separates these: operators are defined solely by truth tables under unfamiliar names, so models must reason from definitions. We test nine operators: four standard (AND, OR, XOR, IF-THEN), four novel-named (BLIF, TARN, QUEX, DREM), and one Trojan (ZENT = XOR under a novel name), across depths 1–10 on five models. Key findings:

1. Reasoning–Output Dissociation: All 31 of Claude's depth-7 errors have correct CoT yet wrong answers; 17/19 mixed-operator chain errors show the same pattern. Mechanistic probes show 99.99% model confidence on wrong answers and only 16% recovery from answer-first prompting, indicating pre-committed errors.
2. Two Failure Modes: Strategy failures at depth 2 (terse retrieval; +62pp from ETT scaffolding) and content failures at depth 7 (full CoT but systematic errors; +8–30pp from ETT, 0/300 post-intervention).
3. Processing Strategy Divergence: GPT-4o uses 141 tokens for standard operators vs. 343 for novel (2.4×) at identical accuracy (99.5%), with AND/OR answered in 1 token. Llama's novelty gap widens to 28pp at depth 8–9 while the Trojan stays at 92–100%.
4. Cross-Domain Generalization: Arithmetic replication (modular operators mod 5) reproduces the novelty gap (6pp at depth 7) but shows lookup errors, not dissociation—the Boolean dissociation is domain-specific, not a generic depth-scaling artifact.

2 Related Work

Reasoning versus memorization. McCoy et al. (2024) show LLM behavior reflects statistical training patterns, with performance degrading on atypical examples. Mirzadeh et al. (2025) find that symbolic perturbations to GSM8K cause accuracy drops up to 65%. Zhao et al. (2024) demonstrate that semantically irrelevant token changes cause large accuracy drops via token-level biases. Our work provides a complementary finding: accuracy on novel operators matches familiar ones when given sufficient generation budget, but the behavioral strategy diverges sharply, a distinction that accuracy-based evaluation alone cannot detect.

Chain-of-thought and dual-process analogies. Turpin et al. (2024) show that chain-of-thought explanations can be unfaithful to the model's actual reasoning process; Lanham et al. (2024) further characterize when CoT influences model predictions. Our finding is complementary: we identify cases where CoT is faithful and correct yet the final answer is wrong. Hagendorff et al. (2023) find LLMs exhibit human-like intuitive biases paralleling dual-process theory (Kahneman, 2011); our response length analysis provides consistent behavioral evidence.

Logical reasoning benchmarks. ProntoQA (Saparov & He, 2023) tests syllogistic reasoning with fictional entities; LogicBench (Parmar et al., 2024) evaluates 25 reasoning patterns; ProofWriter (Tafjord et al., 2021) generates proofs over natural language rules; FLD (Morishita et al., 2024) provides formal deduction benchmarks. These benchmarks isolate content novelty (new entities) while retaining standard logical connectives. We isolate rule-name novelty.

Counterfactual and rule learning. Wu et al. (2024) test LLMs on counterfactual task variants and find substantial drops; our novel operator names avoid triggering preexisting associations. This complements the reversal curse (Berglund et al., 2024); our Trojan tests whether models can apply a known function under a new label. Dziri et al. (2023) show transformers solve compositional tasks via linearized subgraph matching; Dasgupta et al. (2024) demonstrate content effects paralleling human cognitive biases.

3 The Novel Operator Benchmark

3.1 Operator Design

The benchmark is contamination-proof by construction: novel names do not appear in training corpora as logical connectives, and operators are defined solely by truth tables in the prompt, forcing models to reason from definitions rather than retrieve memorized associations.

There are 16 binary Boolean functions. After removing six trivial functions (two constants, two projections, and two negated projections), ten non-trivial operators remain. We select nine, omitting NAND (to keep Group A at four operators) and converse nonimplication ($B \wedge \neg A$, which mirrors BLIF's structure).

Group A (Standard) contains AND, OR, XOR, and IF-THEN, which appear extensively in pretraining corpora.

Group B (Novel-named) contains four operators given unfamiliar names: BLIF (inhibition, $A \wedge \neg B$), TARN (NOR, $\neg(A \vee B)$), QUEX (converse implication, $B \rightarrow A$), and DREM (XNOR, $A \equiv B$). These sample the structural space along symmetry and sparsity (number of True rows).

Group C (Trojan) contains ZENT, which has XOR's exact truth table under a novel name. If ZENT performance matches XOR, the model applies operators from definitions regardless of name.

Table 1: Truth tables for all nine operators. Group B operators use novel names. ZENT (Group C) is XOR's truth table under a novel name, serving as the Trojan control.

Group	Name	TT	TF	FT	FF	Sym	#T
A (Standard)	AND	T	F	F	F	Y	1
	OR	T	T	T	F	Y	3
	XOR	F	T	T	F	Y	2
	IF-THEN	T	F	T	T	N	3
B (Novel)	BLIF	F	T	F	F	N	1
	TARN	F	F	F	T	Y	1
	QUEX	T	T	F	T	N	3
	DREM	T	F	F	T	Y	2
C (Trojan)	ZENT	F	T	T	F	Y	2

3.2 Problem Generation

Each problem defines the relevant operator via truth table, specifies Boolean variable assignments, and asks the model to evaluate a left-associated chain expression.

Depth 1 (single-step): evaluate A OP B given values of A and B. Depths 2–10: evaluate left-associated homogeneous chains, e.g., $((A \text{ OP } B) \text{ OP } C) \text{ OP } D$ at depth 3. All variable assignments are sampled uniformly; we generate 50 instances per (operator, depth) condition.

The core benchmark contains 2,250 problems (9 operators $\times$ 5 depths $\times$ 50 instances) at depths 1–5, plus 2,250 extended problems at depths 6–10 (Section 4.7), 1,200 problems testing representation format effects (Appendix B), 1,200 testing ETT prompting at depths 2–5, and 600 testing ETT at depth 7 (Section 4.6), plus 600 mixed-operator chain problems (Section 5), totaling up to 8,100 problems per model depending on applicable interventions (reasoning models o3-mini and QwQ-32B do not receive ETT or representation experiments; QwQ-32B is tested at depths 7 and 10 only in the extended experiment, yielding $\sim$3,150 problems).

3.3 Prompting Interventions

Explicit Truth-Table Tracing (ETT). A prompting intervention that forces step-by-step truth table lookup at each chain step, testing whether structured scaffolding improves performance on novel operators.

4 Results

We evaluate five models: GPT-4o-2024-11-20, Claude Sonnet 4, Llama 3.1 70B Instruct, o3-mini-2025-01-31, and QwQ-32B, each completing up to 8,100 problems. Standard models use temperature 0.0 and max_tokens=2048 (4,096 for depths 6–10); reasoning models use default temperature with max_tokens=4,096 (8,192 for depths 6–10).

4.1 Overview: Accuracy by Operator Group and Depth

All models achieve $\geq$96% at depth 1. At depth 5, GPT-4o and Claude Sonnet 4 achieve $\geq$96% across all groups; Llama 3.1 70B shows a modest novelty gap. A notable exception: Llama's Group A accuracy dips to 91% at depth 4, driven by IF-THEN at 68%.

The novelty gap (Group A $-$ Group B) is negligible for GPT-4o (0pp at depth 5) and Claude Sonnet 4 ($-$1pp), but present for Llama (11pp, $p < 0.01$, Welch's t-test) and moderate for QwQ-32B (6pp). Per-operator analysis (Appendix C) shows Llama's gap concentrates on QUEX (78%) and DREM (86%), confirming that structural properties influence difficulty independently of familiarity.

Table 2: Accuracy (%) by operator group and chain depth. GPT-4o, Claude Sonnet 4, and QwQ-32B show no meaningful novelty gap at depth 5 ($|\text{gap}| \leq 6$pp). Llama 3.1 70B shows an 11pp gap at depth 5, concentrated on structurally complex novel operators. $n = 50$ instances per cell.

		Accuracy (%)					Novelty Gap (A−B, pp)			
Model	Group	d=1	d=2	d=3	d=4	d=5	d=2	d=3	d=4	d=5
GPT-4o	A (Std)	100	81.0	98.0	97.5	99.5				
	B (Novel)	100	69.0	99.0	99.5	99.5	12.0	−1.0	−2.0	0
	C (Trojan)	100	82.0	98.0	100	100				
Claude Sonnet 4	A (Std)	100	95.0	92.5	96.5	97.5				
	B (Novel)	100	93.0	94.0	94.0	98.5	2.0	−1.5	2.5	−1.0
	C (Trojan)	100	100	94.0	94.0	96.0				
Llama 3.1 70B	A (Std)	100	98.5	100	91.0	98.5				
	B (Novel)	100	91.5	91.5	93.5	87.5	7.0	8.5	−2.5	11.0
	C (Trojan)	100	88.0	92.0	96.0	96.0				
o3-mini	A (Std)	100	100	100	100	100				
	B (Novel)	100	100	100	100	100	0	0	0	0
	C (Trojan)	100	100	100	100	98.0				
QwQ-32B	A (Std)	100	99.5	100	94.5	100				
	B (Novel)	99.0	95.5	93.5	93.0	94.0	4.0	6.5	1.5	6.0
	C (Trojan)	96.0	74.0	82.0	100	100				

Table 3: XOR vs. ZENT accuracy (%) by depth ($n = 50$ per cell). No model shows a significant name effect at depth 5 (Fisher's exact $p \geq 0.49$). 95% binomial CIs shown for depth 5; Fisher's p in rightmost column.

	d=1		d=2		d=3		d=4		d=5		
Model	XOR	ZENT	XOR	ZENT	XOR	ZENT	XOR	ZENT	XOR	ZENT	p
GPT-4o	100	100	62	82	92	98	90	100	98±4	100±0	1.0
Claude Sonnet 4	100	100	98	100	86	94	100	94	100±0	96±5	0.49
Llama 3.1 70B	100	100	100	88	100	92	96	96	98±4	96±5	1.0
o3-mini	100	100	100	100	100	100	100	100	100±0	98±4	1.0
QwQ-32B	100	96	98	74	100	82	92	100	100±0	100±0	1.0

4.2 The Trojan Operator: XOR vs. ZENT

ZENT has XOR's exact truth table under a novel name, differing only in the operator name ($n = 50$ per cell, independently sampled).

Table 3 presents the central result: at depth 5, no model shows a significant XOR–ZENT gap (Fisher's exact $p \geq 0.49$ for all). At depth 2, GPT-4o shows a reversal: ZENT (82%) outperforms XOR (62%), suggesting name familiarity can hurt at the strategy–reasoning transition.

4.3 The Depth-2 Reasoning Mode Transition

GPT-4o's accuracy follows a non-monotonic trajectory: 100% at depth 1, a sharp dip at depth 2, then recovery by depth 3. Response length reveals the mechanism (Figure 2): at depth 2, the model still attempts brief responses (XOR: 1 token, 62%; TARN: 5 tokens, 36%), but by depth 3 commits to full CoT (TARN: 321 tokens, 98%). This reasoning mode transition is operator-dependent: BLIF achieves 96% at depth 2 (one True row yields predictable chains), while balanced operators suffer sharply. ETT scaffolding has its largest effect at this transition point (Section 4.6).

4.4 Behavioral Strategy Divergence: Response Length Analysis

We measure response length (completion tokens) as a behavioral proxy for whether models answer via terse retrieval or explicit chain-of-thought.

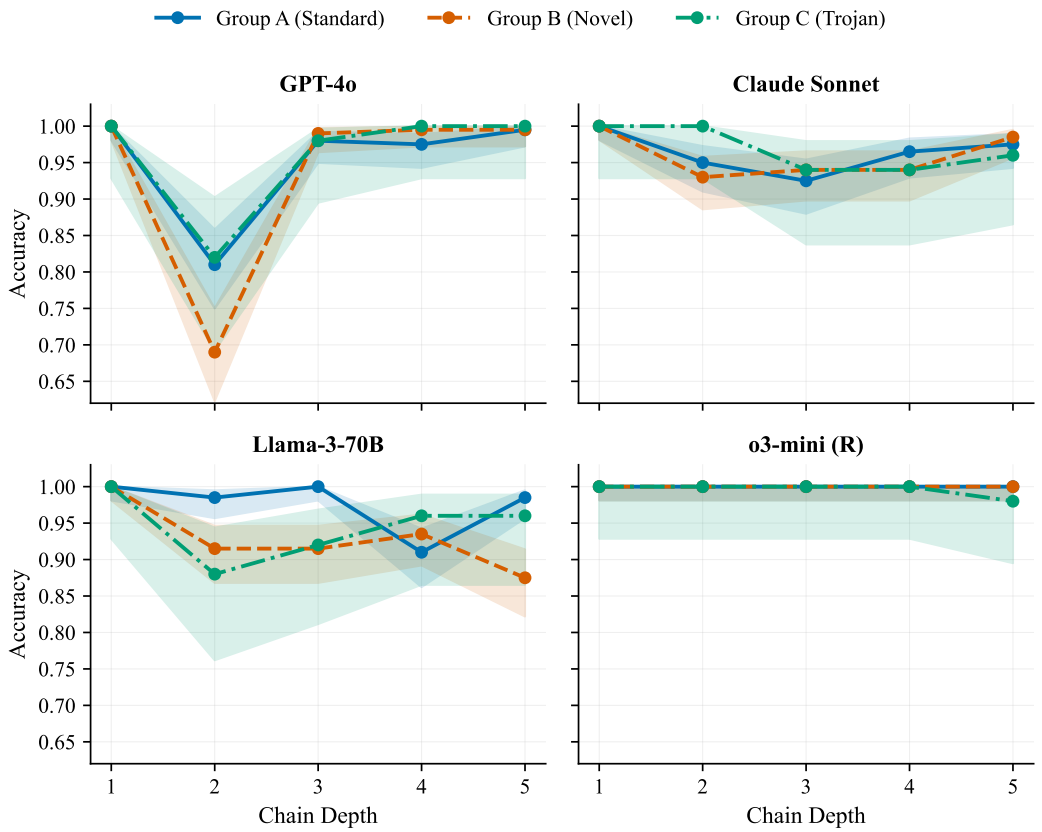

Figure 1: Accuracy by operator group and chain depth (1–5) across five models. The depth-4 data point fills the recovery trajectory between the depth-2 dip and depth-5 ceiling. Llama shows a notable non-monotonic dip at depth 4 (91% Group A), driven by IF-THEN (68%). All conditions use $n = 50$ instances; shaded regions show 95% binomial CIs.

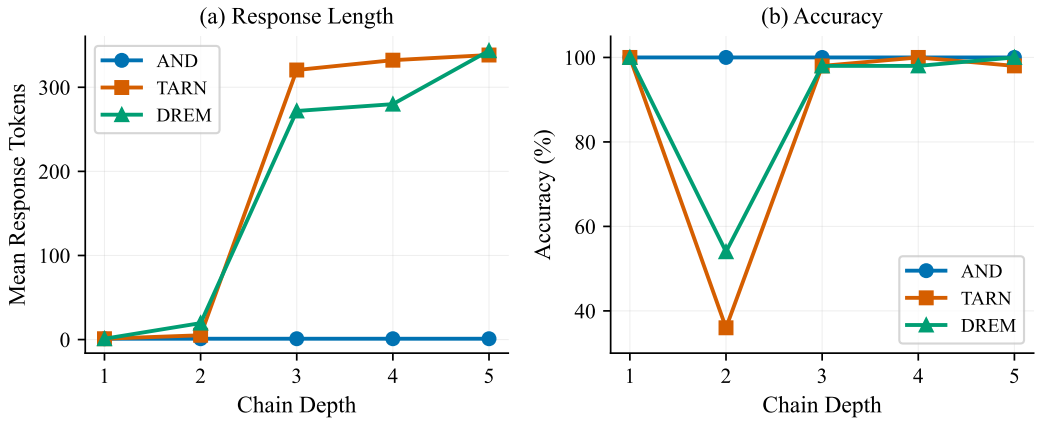

Figure 2: GPT-4o response length and accuracy by depth. (a) AND stays at 1 token (pure retrieval); TARN/DREM jump from <20 tokens at depth 2 to 300+ at depth 3. (b) Accuracy dips at depth 2 and recovers once models adopt explicit reasoning.

GPT-4o uses 141 tokens for Group A vs. 343 for Group B at depth 5: a 2.4× processing cost at identical accuracy (99.5%). AND and OR are answered in exactly 1 token (pure retrieval), while every novel operator requires 335+ tokens. Llama is the only model where

Table 4: Mean response length (completion tokens ± std. dev.) at depth 5 by operator group ($n = 200$ for A/B, $n = 50$ for C). GPT-4o's high Group A variance reflects the bimodal distribution: AND/OR produce 1 token while XOR/IF-THEN produce ∼300.

	Group A (Std)		Group B (Novel)		Group C (Trojan)	
Model	Tokens	Acc	Tokens	Acc	Tokens	Acc
GPT-4o	141±181	99.5%	343±67	99.5%	357±35	100%
Claude Sonnet 4	199±54	97.5%	272±15	98.5%	279±16	96%
Llama 3.1 70B	103±180	98.5%	124±168	87.5%	241±117	96%
o3-mini	227±176	100%	630±266	100%	375±118	98%

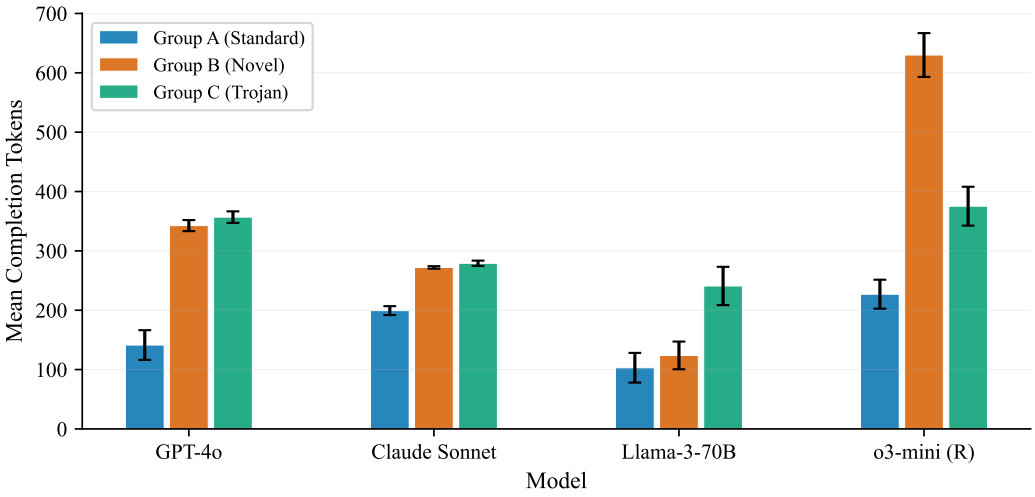

Figure 3: Mean completion tokens at depth 5 by operator group across four of five models (QwQ-32B excluded; response length data not collected). GPT-4o shows the starkest divergence between standard (141 tokens) and novel (343 tokens) groups. Error bars show 95% CIs.

longer processing does not compensate: Group B accuracy (87.5%) lags Group A (98.5%) despite additional reasoning.

4.5 Error Analysis: Standard Operator Interference

Error analysis reveals that models substitute specific familiar operators rather than producing random errors. At depth 2, all 14 of Claude's DREM errors produce AND-chain output ($p < 10^{-6}$), and all 6 of Llama's match OR ($p = 0.001$). At depths 3–5, the substitution target shifts to XOR (structurally similar to XNOR), paralleling the reasoning mode transition from simple-operator defaults to structurally-similar-operator confusion (full analysis in Appendix G).

4.6 ETT Prompting

ETT forces step-by-step truth table lookup at each chain step. We test Group B operators at depths 2, 3, and 5 for the three standard models (o3-mini excluded: 100% across all conditions).

At depth 2, ETT yields up to +62pp (GPT-4o on TARN) by forcing step-by-step reasoning. At depths 3–5, GPT-4o shows zero improvement (already at ceiling), while Claude and Llama benefit moderately. DREM shows the most consistent benefit, consistent with ETT helping avoid XOR/XNOR confusion.

Table 5: ETT improvement (pp) on Group B operators ($n = 50$ per cell). Largest effects at depth 2 (reasoning mode transition). Paired baselines, seed 44.

	Depth 2				Depth 3				Depth 5			
Model	BLIF	TARN	QUEX	DREM	BLIF	TARN	QUEX	DREM	BLIF	TARN	QUEX	DREM
GPT-4o	+10	+62	+6	+26	0	0	0	0	0	0	0	0
Claude Sonnet 4	0	0	0	+30	0	+14	+8	+12	0	0	+4	+12
Llama 3.1 70B	+10	+6	+22	+10	0	+12	+10	+16	+22	+2	+16	+16

Table 6: ETT at depth 7 for Claude Sonnet 4 ($n = 50$ per cell, seed 142). ETT achieves 100% on all operators. All 34 baseline errors answer False on True-ground-truth instances.

	IF-THEN	ZENT	TARN	DREM	BLIF	XOR
Baseline (%)	70	82	88	92	100	100
ETT (%)	100	100	100	100	100	100
Δ (pp)	+30	+18	+12	+8	0	0
Fisher's p	<0.0001	0.003	0.027	0.12	—	—
Baseline errors	15	9	6	4	0	0
on GT=True	15	9	6	4	—	—
on GT=False	0	0	0	0	—	—

ETT at depth 7: addressing content failures. At depth 7, Claude already produces full CoT ($\sim$340 tokens), yet ETT achieves 100% on all six operators (0/300 errors vs. 34/300 baseline; $p < 10^{-6}$). All 34 baseline errors occur on True-ground-truth instances, suggesting a "False" prior when uncertainty accumulates ($p < 10^{-6}$, binomial test against 52% base rate).

Programmatic error chain verification. Programmatic verification of all 34 depth-7 baseline errors (Appendix H) confirms that every error chain correctly computes "True" at the final step, then declares "False" as the answer (IF-THEN: 15/15, ZENT: 9/9, TARN: 6/6, DREM: 4/4), a reasoning–output dissociation where the failure lies in the answer declaration, not the reasoning process.

Self-correction probe. We present each of Claude's 31 error chains to both Claude and GPT-4o in new conversations and ask what value was computed in the last step. Claude self-corrects on all 31 (100%) and GPT-4o on 30/31 (97%), confirming both models can extract the correct answer from the reasoning trace, localizing the dissociation to autoregressive generation rather than reasoning failure.

Temperature stability analysis. We re-run Claude at depth 7 across $T \in \{0.0, 0.5, 1.0\}$ (2,700 total probes). The dissociation rate is temperature-stable: 57/900 at T=0.0, 53/900 at T=0.5, 51/900 at T=1.0. For the 31 originally-wrong instances, 46% remain dissociated at T=0.0, confirming a structural rather than stochastic property.

Mechanistic probes: why does depth 7 break? Three probes investigate the mechanism behind the depth-7 dissociation (Figure 4):

(1) Logprob analysis. GPT-4o assigns mean probability 99.99% to both correct (d=5) and wrong (d=7) final answer tokens (Figure 4a). There is no internal uncertainty signal distinguishing correct from dissociated outputs.

(2) CoT truncation. We present Claude's correct depth-7 reasoning chains (wrong final answer removed) to GPT-4o. It correctly extracts the answer in 31/31 cases (100%), confirming the correct answer is fully present in the CoT.

(3) Answer-first prompting. We reverse the CoT order, asking Claude to state the answer before reasoning (Figure 4b). Depth-5 stays at 100%; depth-7 improves from 0% to only 16.1% (5/31), indicating the error is largely pre-committed: the model decides the wrong answer before generating correct intermediate steps.

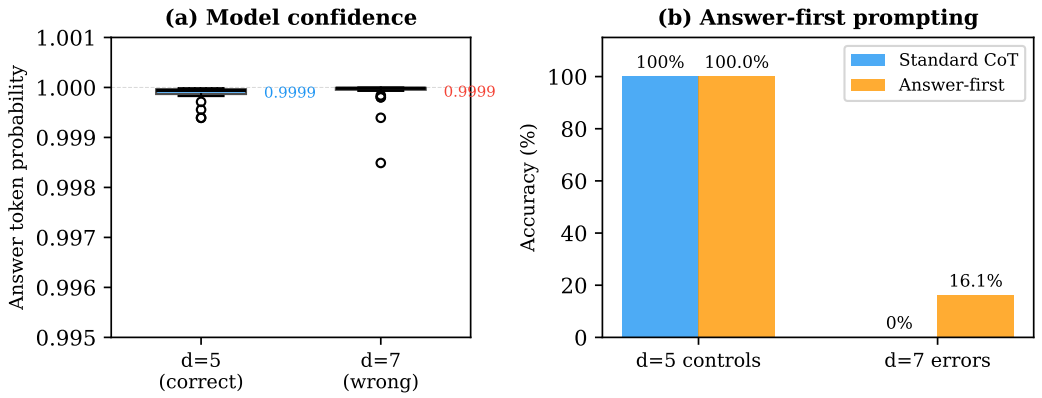

Figure 4: Mechanistic probes on depth-7 dissociation. (a) GPT-4o's answer-token probability is ≥0.998 for both correct (d=5) and wrong (d=7) outputs—no uncertainty signal. (b) Answer-first prompting recovers only 16.1% of d=7 errors, vs. 100% on d=5 controls, indicating pre-committed failure.

Table 7: Group-level accuracy (%) at depths 6–10 ($n = 200$ per group cell, $n = 50$ for Trojan). GPT-4o and o3-mini maintain near-perfect accuracy. Claude shows a depth-7 dip with recovery. Llama's novelty gap widens from 11pp at depth 5 to 28pp at depth 8–9. QwQ-32B tested at d=7 and d=10 only (—); its novelty gap vanishes by d=10.

Model	Group	d=6	d=7	d=8	d=9	d=10
GPT-4o	A (Std)	99.5	100	99.0	100	100
	B (Novel)	99.0	100	99.5	99.0	99.0
	C (Trojan)	100	100	98.0	100	100
Claude Sonnet 4	A (Std)	99.5	96.0	97.5	99.0	99.5
	B (Novel)	94.5	93.5	100	99.5	99.0
	C (Trojan)	96.0	80.0	100	100	100
Llama 3.1 70B	A (Std)	97.5	98.5	94.5	94.0	94.0
	B (Novel)	89.0	84.0	66.5	66.0	76.5
	C (Trojan)	94.0	98.0	96.0	92.0	100
o3-mini	A (Std)	100	100	100	100	100
	B (Novel)	100	100	100	100	99.5
	C (Trojan)	100	100	100	100	100
QwQ-32B	A (Std)	—	93.5	—	—	88.5
	B (Novel)	—	92.5	—	—	89.5
	C (Trojan)	—	86.0	—	—	90.0

Code-format operator definitions match or exceed truth tables at depth 2 for all models (Appendix B). Restrictive token limits (max_tokens=256) create phantom accuracy collapses of 30–54pp by truncating verbose reasoning on novel operators (Appendix D).

4.7 Extended Depth Analysis (d=6–10)

To test whether the novelty gap emerges at greater chain depths, we extend the benchmark to depths 6–10, generating up to 2,250 additional problems per model (9 operators × 5 depths × 50 instances; QwQ-32B tested at d=7 and d=10 only due to inference cost). Table 7 summarizes group-level accuracy for all five models.

GPT-4o and o3-mini maintain ≥99% through depth 10. Llama 3.1 70B shows a widening novelty gap: 11pp at d=5 to 28pp at d=8–9, with QUEX collapsing to 40% at d=9, while the Trojan stays at 92–100%—strong evidence the gap reflects difficulty with novel logic, not name unfamiliarity.

Claude Sonnet 4 shows a non-monotonic depth-7 dip with full recovery by depth 8 (Figure 5a). All 31 depth-7 errors involve full CoT (∼348 tokens) yet answer False on True-

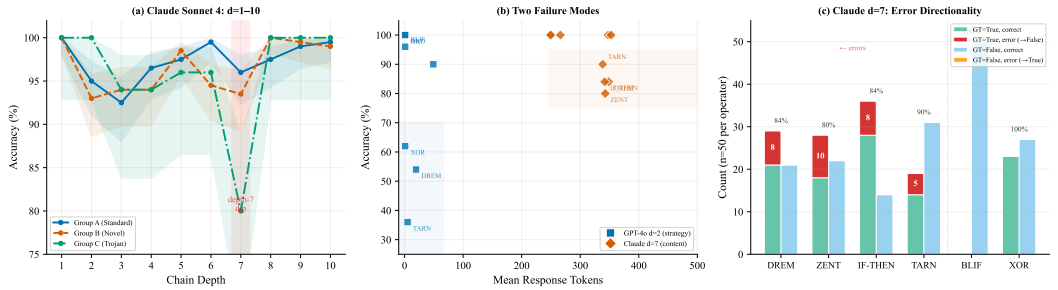

Figure 5: Two failure modes. (a) Claude accuracy across depths 1–10, showing the depth-7 dip with recovery by depth 8. (b) Strategy failures (GPT-4o d=2): low tokens, low accuracy. Content failures (Claude d=7): high tokens, wrong answers. (c) All 31 depth-7 errors occur on True-ground-truth instances.

Table 8: Arithmetic domain accuracy (%) for GPT-4o and Claude Sonnet 4 ($n = 100$ for A, $n = 150$ for B, $n = 50$ for C per depth). GPT-4o shows a 6pp novelty gap at depth 7 concentrated on FLOX lookup errors. Claude shows a smaller gap (1.3pp) with errors distributed across groups.

Model	Group	d=1	d=2	d=3	d=4	d=5	d=6	d=7	Gap_{d7}
GPT-4o	A (Std)	100	100	100	100	100	100	100	—
	B (Novel)	100	100	99.3	98.0	95.3	95.3	94.4	5.6
	C (Trojan)	100	100	100	100	100	100	100	—
Claude	A (Std)	100	98.0	98.0	100	100	100	100	—
	B (Novel)	100	99.3	99.3	100	100	100	98.7	1.3
	C (Trojan)	100	98.0	100	100	100	100	100	—

ground-truth instances (Figure 5b–c), the content failure pattern. We hypothesize depth 7 sits at a critical threshold where chain length first exceeds reliable context for answer extraction.

4.8 Arithmetic Domain Replication

To test whether the novelty gap generalizes beyond Boolean logic, we replicate the experimental design in modular arithmetic (mod 5). We define six operators over integers 0–4: two standard (ADD5, MUL5), three novel-named (FLOX: $(2a+3b)$ mod 5; GREX: $(ab+a+1)$ mod 5; VARN: $(3a+b+2)$ mod 5), and one Trojan (KELP = ADD5 under a novel name). Each operator is presented via a 5×5 lookup table. We generate 2,100 problems (6 operators × 7 depths × 50 instances) and evaluate both GPT-4o and Claude Sonnet 4.

Table 8 shows the same qualitative pattern as the Boolean domain: novel operators degrade with depth while standard and Trojan operators remain robust. GPT-4o shows a clear 6pp novelty gap at depth 7; Claude shows a smaller gap (1.3pp) with errors more uniformly distributed across operator groups, including standard operators at depths 2–3. The Trojan result confirms that, as in the Boolean case, name novelty alone does not impair performance.

Different failure mode. Detailed error analysis of GPT-4o's 9 depth-7 errors reveals a qualitatively different failure mechanism from the Boolean domain. All 9 errors (8 FLOX, 1 VARN) are lookup errors: the model misreads the 5×5 table at specific entries (e.g., FLOX(3,4) misread in 4/9 errors). Crucially, zero errors exhibit reasoning–output dissociation: wrong answers follow consistently from incorrect intermediate lookups, unlike the Boolean domain where all 31 errors have correct reasoning chains but wrong final answers. This domain-specific contrast localizes the Boolean dissociation as specific to how LLMs process Boolean operator semantics, not a generic depth-scaling artifact.

5 Discussion

Reasoning–output dissociation. Our dissociation is the converse of Turpin et al. (2024)'s unfaithful CoT: here, CoT is faithful and correct yet produces the wrong answer, a failure of output generation rather than explanation. The self-correction probe (31/31) and temperature stability (stable across T=0.0–1.0) localize this to a structural property of autoregressive generation. Mechanistic probes deepen this picture: logprob analysis shows 99.99% confidence on both correct and wrong answers (no uncertainty signal), while answer-first prompting recovers only 16% of errors, suggesting the model pre-commits to the wrong answer before generating the correct reasoning chain.

Differential internalization. The 2.4× token divergence at equivalent accuracy reveals familiar operators are internalized (1-token retrieval) while novel operators require explicit reasoning. The two failure modes (strategy failures at depth 2, +62pp from ETT; content failures at depth 7, +8–30pp) show distinct remediation profiles.

Mixed-operator chains. To test whether the dissociation is an artifact of homogeneous chains, we evaluate mixed-operator chains at depths 3, 5, and 7 (50 instances each) under two conditions: all-novel (cycling BLIF→TARN→QUEX→DREM) and novel-standard alternating. Claude Sonnet 4 shows 17/19 errors (89%) are dissociations across both conditions, confirming the phenomenon is not an artifact of operator repetition. Novel-standard alternating chains drop to 90% at depth 7 as compositional complexity increases; GPT-4o achieves 82–100% with errors concentrated at depth 7.

Limitations. While our mechanistic probes (logprobs, answer-first) provide evidence that the dissociation is pre-committed, internal activation analysis would more precisely identify the computational locus of failure. The arithmetic replication (Section 4.8) shows the dissociation does not generalize to all novel operator domains, suggesting it is specific to how LLMs process Boolean semantics; richer domains (first-order, probabilistic) remain untested. Sample size (n=50) suffices for the large effects reported but limits detection of small per-operator differences; the DREM ETT improvement at depth 7 (+8pp, $p = 0.12$) does not reach significance individually, though it contributes to the significant aggregate effect.

6 Conclusion

We introduced the Novel Operator Test, a benchmark that disentangles reasoning from retrieval by separating operator logic from operator name, and identified a reasoning–output dissociation: correct CoT yet wrong answers (31/31 homogeneous, 17/19 mixed chains). Mechanistic probes reveal the model is maximally confident (99.99%) even when wrong and pre-commits to incorrect answers before generating correct reasoning. An arithmetic domain replication confirms the novelty gap generalizes (6pp at depth 7) but shows a different failure mode (lookup errors, not dissociation), localizing the Boolean dissociation as domain-specific rather than a generic depth-scaling artifact. Verifying chain-of-thought correctness is insufficient to guarantee answer correctness.

References

Lukas Berglund, Meg Tong, Max Kaufmann, Mikita Blaber, Max Tegmark, and Tamera Maharaj. The reversal curse: LLMs trained on "A is B" fail to learn "B is A". In Proceedings of ICLR, 2024.

Ishita Dasgupta, Andrew K Lampinen, Stephanie CY Chan, Hannah R Sheahan, Antonia Creswell, Dharshan Kumaran, James L McClelland, and Felix Hill. Language models, like humans, show content effects on reasoning tasks. PNAS Nexus, 3(7), 2024.

Nouha Dziri, Ximing Lu, Melanie Sclar, Xiang Lorraine Li, Liwei Jian, Bill Yuchen Lin, Peter West, Chandra Bhagavatula, Ronan Le Bras, Jena D Hwang, Sean Welleck, et al.

Faith and fate: Limits of transformers on compositionality. In Proceedings of NeurIPS, 2023.

Thilo Hagendorff, Sarah Fabi, and Michal Kosinski. Human-like intuitive behavior and reasoning biases emerged in large language models but disappeared in ChatGPT. Nature Computational Science, 3:833–838, 2023.

Daniel Kahneman. Thinking, Fast and Slow. Farrar, Straus and Giroux, 2011.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, et al. Measuring faithfulness in chain-of-thought reasoning. In Proceedings of COLM, 2024.

R Thomas McCoy, Shunyu Yao, Dan Friedman, Matthew Hardy, and Thomas L Griffiths. Embers of autoregression show how large language models are shaped by the problem they are trained to solve. Proceedings of the National Academy of Sciences, 121(41), 2024.

Iman Mirzadeh, Keivan Alizadeh, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. GSM-Symbolic: Understanding the limitations of mathematical reasoning in large language models. In Proceedings of ICLR, 2025.

Terufumi Morishita, Gaku Morio, Atsuki Yamaguchi, and Yasuhiro Sogawa. FLD: Formal logic deduction for natural language reasoning. In Proceedings of NAACL, 2024.

Mihir Parmar, Nisarg Patel, Neeraj Varshney, Mutsumi Nakamura, Man Luo, Santosh Mashetty, Arindam Mitra, and Chitta Baral. LogicBench: Towards systematic evaluation of logical reasoning ability of large language models. In Proceedings of ACL, 2024.

Abulhair Saparov and He He. Language models are greedy reasoners: A systematic formal analysis of chain-of-thought. In Proceedings of ICLR, 2023.

Oyvind Tafjord, Bhavana Dalvi, and Peter Clark. ProofWriter: Generating implications, proofs, and abductive statements over natural language. In Findings of ACL, 2021.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In Proceedings of NeurIPS, 2024.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In Proceedings of NeurIPS, 2022.

Zhaofeng Wu, Linlu Qiu, Alexis Ross, Ekin Akyurek, Boyuan Chen, Bailin Wang, Najoung Kim, Jacob Andreas, and Yoon Kim. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks. In Proceedings of NAACL, 2024.

Bowen Zhao, Hanqi Jiang, Bolei Cai, Jize Dong, Yingjie Lu, Gangshan Feng, and Jian Wu. A peek into token bias: Large language models are not yet genuine reasoners. In Proceedings of EMNLP, 2024.

A Prompt Templates

The standard prompt provides the operator truth table, variable assignments, and asks for the answer ("True" or "False"). The ETT prompt additionally forces step-by-step truth table lookup:

A.1 ETT Prompt (Depth 2, Novel Operator)

You are given the following logical operator(s):

BLIF is defined by the following truth table:
A=True, B=True -> False
A=True, B=False -> True
A=False, B=True -> False
A=False, B=False -> False

Question: If A=True, B=False, C=True,
what is (A BLIF B) BLIF C?

Solve by explicitly looking up each step
in the truth table:
Step 1: Identify the innermost expression:
A BLIF B, where A=True, B=False.
Look up row (True, False) in the BLIF
truth table. Result = _____
Step 2: Let R1 = your result from Step 1.
Now evaluate R1 BLIF C, where R1=____,
C=True. Look up row (R1, True) in the
BLIF truth table. Result = _____

Final answer (True or False):

B Representation Format Results

Table 9: Accuracy (%) by representation format for Group B operators ($n = 200$ per cell, aggregated across 4 operators). Code format matches or exceeds truth tables at depth 2 for all models.

	Depth 1			Depth 2		
Model	Table	Code	NL	Table	Code	NL
GPT-4o	100	72	89.5	72.5	93.5	76
Claude Sonnet 4	100	100	100	92.5	100	90.5
Llama 3.1 70B	100	88	100	86.5	98	98.5

C Per-Operator Accuracy at Depth 5

Table 10: Accuracy (%) per operator at depth 5 ($n = 50$ per cell). GPT-4o, Claude, and o3-mini achieve $\geq$90% on all operators. Llama shows difficulty on QUEX (78%) and DREM (86%); QwQ-32B shows difficulty specifically on DREM (78%).

Grp	Operator	Sym	#T	GPT-4o	Claude	Llama	o3-mini	QwQ
A	AND	Y	1	100	100	100	100	100
	OR	Y	3	100	100	100	100	100
	XOR	Y	2	98	100	98	100	100
	IF-THEN	N	3	100	90	96	100	100
B	BLIF	N	1	100	100	94	100	100
	TARN	Y	1	98	98	92	100	98
	QUEX	N	3	100	96	78	100	100
	DREM	Y	2	100	100	86	100	78
C	ZENT	Y	2	100	96	96	98	100

D Token Limits as a Methodological Confound

With max_tokens=256, we observed apparent accuracy collapses of 30–54pp between XOR and ZENT at depth 5 (e.g., GPT-4o: XOR 98%, ZENT 44%). These were truncation artifacts: novel operators elicit verbose responses exceeding 256 tokens. Re-running with max_tokens=2048 eliminated the artifact (ZENT: 44% → 100%). Restrictive token limits interact with operator familiarity to create phantom performance gaps.

E GPT-4o Full Per-Operator Results

Table 11: Full per-operator accuracy (%) across all depths for GPT-4o. Non-monotonic depth-2 dip and mean response tokens at depth 2 illustrate the reasoning mode transition. Depth 4 fills the recovery trajectory between depth 3 and 5.

Group	Operator	d=1	d=2	d=3	d=4	d=5	d=2 tokens
A	AND	100	100	100	100	100	1
	OR	100	100	100	100	100	1
	XOR	100	62	92	90	98	1
	IF-THEN	100	62	100	100	100	1
B	BLIF	100	96	100	100	100	1
	TARN	100	36	98	100	98	5
	QUEX	100	90	100	100	100	49
	DREM	100	54	98	98	100	20
C	ZENT	100	82	98	100	100	52

F Prompt-Compliance Control Experiment

A potential confound for the response length analysis is that the standard prompt instructs "Answer with only True or False," which may directly cause GPT-4o's 1-token responses on familiar operators. To test this, we run a control experiment replacing the instruction with "Show your step-by-step reasoning, then state your final answer as FINAL ANSWER: True or FINAL ANSWER: False" ($n = 30$ per condition, 480 problems total).

Table 12: Standard vs. "show your work" prompts for GPT-4o ($n = 30$ per cell). Show-work prompt forces verbose reasoning (342–770 tokens for AND/OR) and eliminates depth-2 strategy failures (TARN: 33%→100%).

		Accuracy (%)		Mean tokens	
Operator	Depth	Standard	Show-work	Standard	Show-work
AND	2	100	100	1	361
AND	5	100	100	1	770
OR	2	100	100	1	343
OR	5	100	100	1	749
BLIF	2	87	100	1	475
BLIF	5	100	100	320	1079
TARN	2	33	100	8	502
TARN	5	100	100	349	894

The 1-token standard responses reflect prompt compliance, but the critical asymmetry remains: AND/OR are correct in 1 token while TARN achieves only 33%, supporting differential internalization.

G Error Analysis: Standard Operator Interference (Full)

We check whether each wrong answer matches a specific standard operator on the same inputs. DREM errors show operator-specific substitution: at depth 2, all 14 of Claude's errors produce AND-chain output ($p < 10^{-6}$), all 6 of Llama's match OR ($p = 0.001$). At depths 3–5, the target shifts to XOR.

H Depth-7 Error Examples and Programmatic Verification

All 31 depth-7 errors have correct CoT through all 7 steps but declare the wrong answer. Two examples:

Example 1: IF-THEN at depth 7.

```
Variables: A=True, B=True, C=False, D=False, E=True, F=True, G=False, H=False
1) A IF-THEN B = True IF-THEN True = True
2) (A IF-THEN B) IF-THEN C = True IF-THEN False = False
3) (...) IF-THEN D = False IF-THEN False = True
4) (...) IF-THEN E = True IF-THEN True = True
5) (...) IF-THEN F = True IF-THEN True = True
6) (...) IF-THEN G = True IF-THEN False = False
7) (...) IF-THEN H = False IF-THEN False = True    <-- correct

False                                              <-- declared answer
```

Example 2: ZENT (Trojan = XOR) at depth 7.

```
Variables: A=True, B=True, C=False, D=False, E=False, F=True, G=True, H=True
1) A ZENT B = True ZENT True = False
2) (...) ZENT C = False ZENT False = False
3) (...) ZENT D = False ZENT False = False
4) (...) ZENT E = False ZENT False = False
5) (...) ZENT F = False ZENT True = True
6) (...) ZENT G = True ZENT True = False
7) (...) ZENT H = False ZENT True = True          <-- correct

False                                             <-- declared answer
```

This pattern holds for all 31 errors (IF-THEN: 8, ZENT: 10, DREM: 8, TARN: 5). All have ground truth True.