ONE EDIT, THREE WORLDS: EVALUATING LLM REASONING BEYOND ANSWER ACCURACY

Anonymous authors
Paper under double-blind review

ABSTRACT

A reliable reasoner must be invariant to changes that do not matter and sensitive to the one change that does. Benchmarks measure the first property well and the second rarely, because every item is scored in isolation. Few prior studies, if any, systematically analyze whether the answer follows the premise that decides it. We propose an evaluation principle that tests both properties together. It builds sibling problems that differ in exactly one premise, and it counts an answer as correct only if the answer changes with that premise and stays the same when the problem is asked again together with its siblings. We call the resulting score verified accuracy. We instantiate the principle as One Edit, Three Worlds (OETW), a benchmark of 1,200 triads, each a base problem with three siblings whose one edited premise leaves the target with no valid answer, exactly one, or two, built on three constraint problems and certified by a solver rather than an LLM judge. The edit is minimal for the model: siblings are byte-identical outside the edited premise, surface-feature classifiers stay near chance, and where probabilities can be read the siblings receive nearly identical ones before reasoning begins. As problems grow, the raw accuracy of seven frontier and open-weight models falls to 33% and their verified accuracy to 2%, where the first is the level of random guessing, and the second means almost every isolated correct answer fails to survive its siblings. What remains is not random guessing but a low-entropy, model-dependent fallback, and neither abstention nor clarification repairs it.

1 INTRODUCTION

Expertise is largely the ability to notice which detail decides the case. A patient record that says "no history of" a condition and one that omits the "no" call for different drugs. A contract clause with "may" in place of "shall" imposes no obligation. A program with one bound written as $<$ instead of $\leq$ is correct, wrong, or never terminates. In each case almost everything on the page is unchanged and the right answer is entirely different.

Reliability has two halves. Invariance is measured well: relabel the options, reorder the premises, add irrelevant text, and check that the answer stays put. Sensitivity, whether the answer moves when the decisive detail does, is rarely measured, because each item is scored in isolation, so a model that answers from the gist can still score well. The question that matters in practice is harder: when two problems differ in one premise that cannot be picked out before reasoning, does the answer track it?

We propose an evaluation principle that answers it. Take any reasoning problem whose answer an oracle can decide, write siblings that share every byte except one premise, and count an answer as correct only if it tracks the edit and survives being re-asked alongside its siblings. We call the resulting score *verified* accuracy and measure every other analysis against it. It captures sensitivity and invariance in one number, and it is stricter than ordinary accuracy, which we call raw accuracy: it discards answers that were right by a constant policy, by label position or by being told that the siblings differ.

We instantiate the principle as *One Edit, Three Worlds* (OETW): a common base and target question, one slot in one premise, and three values of the slot under which the target has no feasible answer, exactly one, or two. The three siblings form a *triad*. The principle needs only two siblings whose

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

answers differ; we use three because these worlds are as far apart as answers can be, and telling them apart requires reasoning about the whole instance. We build OETW on three constraint problems (families) only because a solver decides their answer spaces exactly, with no LLM judge. Each can be restated in natural language or mapped to an agent task, such as configuring options under rules, scheduling, or assigning workers to tasks, and Section 6 shows how the principle carries to other benchmarks.

The OETW edit is minimal in three ways. The sibling prompts are byte-identical outside the edited value, classifiers trained on all public surface features predict a sibling's world no better than chance, and three open-weight models assign almost the same probabilities to the three worlds for all three siblings before reasoning begins. We take these properties as our operational definition of a minimal edit, one whose consequence cannot be detected from the text or from the model's probabilities before reasoning, whatever its edit distance. Perturbation benchmarks test whether a model recomputes once it notices a difference. OETW instead tests whether reasoning can find the consequence of a difference that cannot be detected in advance.

Our results show why verified accuracy is needed. GPT-OSS-20B gave the same one of the three answers to all 1,200 siblings of one family and scored 33.3%, the expected accuracy of a uniform guess. Accuracy cannot tell this run from a guesser, nor from a model that reasons and is right one time in three. Where raw accuracy has settled at 33% as problems grow, median verified accuracy is 2.0% and no triad survives whole. On the open-weight models, small problems separate the three worlds during reasoning and large problems never do. What remains is not a uniform guess but a low-entropy fallback whose direction depends on the model and on the family. On those problems, the answers a model keeps after abstaining are verified in a median of 1.5% of cases, and even when a model recognizes, however the problem is shown, that two answers remain, it usually cannot ask the question that would separate them.

Our contributions are (i) an evaluation principle for any reasoning problem with a decidable answer, with verified accuracy as its score and a measurable definition of a minimal edit; (ii) its instantiation One Edit, Three Worlds, an exact, reproducible release of 1,200 solver-certified triads with a paired protocol; and (iii) evidence across seven models and three families that failure on problems the models can no longer solve is a structured fallback that neither abstention nor clarification repairs and, for the open-weight models, reasoning trajectories that never separate the three worlds.

2 Related Work

Minimal pairs and perturbations. CheckList (Ribeiro et al., 2020) tests classifiers with invariance tests, where a label-preserving edit must not move the prediction, and directional tests, where an edit must move it; contrast sets (Gardner et al., 2020), BLiMP (Warstadt et al., 2020) and counterfactually augmented data (Kaushik et al., 2020) build minimal pairs whose gold label changes. For reasoning, GSM-Symbolic (Mirzadeh et al., 2025) and Putnam-AXIOM (Gulati et al., 2025) change numbers, names and distractors, BBQ (Parrish et al., 2022) pairs ambiguous with disambiguated contexts, and Li et al. (2023a) test whether a model follows a stated hypothetical; a change of wording, an irrelevant sentence or a negation is enough to break current models (Nezhurina et al., 2025; Shi et al., 2023; García-Ferrero et al., 2023). In all of these the edit is visible and its consequence can be derived locally, so what is tested is recomputation or resistance to distraction. OETW asks the same two questions of reasoning problems with an exact oracle, with edits the model cannot see before reasoning, with worlds in which no answer or more than one answer exists, and with both questions tied into one criterion.

Invariance and consistency. Answers move with option labels and positions (Zhao et al., 2021; Zheng et al., 2024; Pezeshkpour & Hruschka, 2024), prompt format and wording (Sclar et al., 2024; Mizrahi et al., 2024), paraphrase (Elazar et al., 2021; Raj et al., 2022; Jang & Lukasiewicz, 2023), premise order (Chen et al., 2024b) and the user's stated opinion (Sharma et al., 2024), and answers to related questions need not agree with one another (Chen et al., 2024a). These measure invariance as a robustness score of its own. OETW measures invariance and sensitivity on the same sibling and uses invariance as a filter on accuracy.

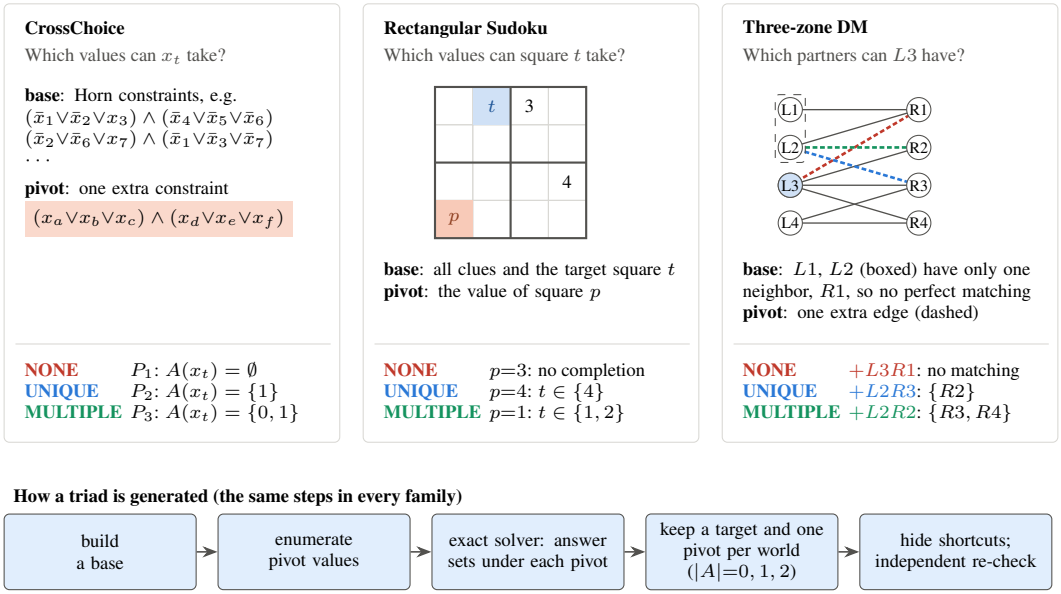


Figure 1: One edit, three worlds, in each family (top) and how every triad is generated (bottom). The siblings share the base and differ only in the pivot, and the solver certifies the target's answer set under each pivot. The detailed generation procedure of each family is in Appendix B.

Abstention and clarification. Unanswerable or underdetermined problems (Sun et al., 2024; Ouyang, 2025; Kirichenko et al., 2025) and studies of self-knowledge (Kadavath et al., 2022) score "unanswerable" as a category of its own, and AR-Bench (Zhou et al., 2025), CLAMBER (Zhang et al., 2024) and CLAM (Kuhn et al., 2022) ask whether a model can pose useful questions under incomplete information. OETW places impossible, determined and underdetermined in one sibling set, so a model must discriminate among the three rather than recognize one, and it accepts any clarifying question the solver certifies as resolving, scoring it both over all triads and over those where the model recognized, however the problem was shown, that two answers remain.

Constraint problems as test beds. ZebraLogic (Lin et al., 2025), PUZZLES (Estermann et al., 2024), SATBench (Wei et al., 2025), Sudoku-Bench (Seely et al., 2025) and Reasoning Gym (Stojanovski et al., 2025) measure solving ability on puzzles of tunable difficulty, and Hazra et al. (2024) characterize reasoning by 3-SAT hardness. Graph problems have been posed in natural language (Wang et al., 2023a; Tang et al., 2024), and Dziri et al. (2023) and Wu et al. (2024) use compositional and counterfactual variants of familiar tasks to separate reasoning from recall. OETW does not compete on difficulty: the constraint problems serve only to make the three answer spaces solver-certifiable, without an LLM judge (Zheng et al., 2023; Wang et al., 2024a).

3 One Edit, Three Worlds

General construction. A finite, decidable reasoning problem is given by premises B, a target question q and one pivot premise $P(\square)$ with a slot. For each legal value v of the slot, let $A(v)$ be the set of answers to q that hold in some solution of $B + P(v)$. We keep three values for which $|A(v)|$ is 0, 1 and 2; the three resulting problems are the siblings of a triad, and $|A(v)|$ is the *state* of the target: NONE (no answer is possible), UNIQUE (the answer is determined) and MULTIPLE (the answer is not yet determined); these three states are the three worlds of the title. NONE thus means that the premises cannot hold together, not that the target has a value we cannot find.

Three instantiations. *CrossChoice* is a system of constraints that are each two 3-clauses; the target is one Boolean variable and the pivot is one extra constraint that differs across the siblings. *Rectangular Sudoku* is a shared partial grid; the pivot is one square filled with different values and the target is another empty square. *Three-zone DM* (named after the Dulmage–Mendelsohn

Table 1: Every request a triad receives: thirteen core requests per triad (counts in parentheses) and two sets of controls on 18 triads per cell. The separate requests test sensitivity (the three answers must differ); the joint, pairwise, relabel and recode requests test invariance (the same sibling, shown with its siblings or under other names, must keep its answer). † marks the four views that define a verified answer: the separate must-answer, the label in the joint forced request, and the labels in the two pairwise requests that contain the sibling.

Request (per triad)	What the model sees and is told	Correct answer
Core requests, every triad		
Separate, optional ($\times 3$)	one sibling; it may answer NONE, UNIQUE or MULTIPLE or abstain, and gives the target value if UNIQUE	the sibling's certified state, and for UNIQUE the exact value
Separate, must-answer† ($\times 3$)	the same sibling; it must commit to one of the three states	as above
Joint, free ($\times 1$)	the base and all three pivots labeled A, B, C; the labels are arbitrary and the states need not occur once each; one state per label, abstaining allowed	each label's certified state
Joint, free must-answer ($\times 1$)	as above, but it must commit	as above
Joint, forced† ($\times 1$)	as above, and the three labels carry exactly one NONE, one UNIQUE and one MULTIPLE	as above
Pairwise† ($\times 3$, one per pair)	the base and two pivots labeled A, B; the two states differ, and which state is absent is not said	each label's certified state
Clarification ($\times 1$)	the MULTIPLE sibling; the target has exactly two values; it asks one yes/no question "is $x = v$?" about another variable	any question whose two answers are both feasible and each makes the target unique, with different values (solver-checked)
Controls, 18 triads per cell		
Relabel ($\times 5$)	the joint forced request under the five other A/B/C orders	each label's certified state
Recode ($\times 6$)	one sibling, with the three states renamed by a temporary codebook; it must commit	the sibling's certified state, in the codebook

decomposition used to build it) is a balanced bipartite graph; each sibling adds a different missing edge and the target is the set of partners one vertex can have across all perfect matchings (Figure 1, top).

How triads are generated. Figure 1 (bottom) shows the steps that every generator follows, with an exact solver in the loop. Appendix B gives the details, and Appendix K shows one real triad of each family.

Validity and release. Every triad passes the independent check, and classifiers trained on public surface features predict the state no better than chance (Appendix D). The release has 100 triads for each of four sizes N per family (the number of variables, the grid order, or the vertices per side), 1,200 in all, built from one master seed by one command in about an hour on 24 CPU cores, with no model output or human labeling; size is a knob within a family, not a difficulty scale across families.

4 MEASUREMENT

We write a *cell* for one model on one family at one size (84 in all, 100 triads each). Every triad receives the thirteen requests of Table 1, and 18 triads per cell receive two further sets of control requests, relabel and recode, reported as separate checks. A *view* is one way a sibling is shown: on its own (a separate request), with both other siblings (joint) or with one of them (pairwise). Separate requests are asked once allowing abstention (optional) and once not (must-answer); the joint request is asked with and without the hint that the three states occur once each (forced and free).

Why the siblings are needed. Suppose a model gives the same answer whether or not it is true. When that answer happens to be right, it tells us nothing about the world, because a constant policy would produce exactly the same answer. A scoring rule that counted one would have to count the other. Whether the model really solved the problem cannot be observed. The only way to see

whether an answer depends on a premise is to change that premise and ask again. This is the purpose of the edit. It matters most in the joint and pairwise views, because in these views one answer cannot be right for every sibling at once. Asking a sibling again on its own, in other words or under other names, cannot reveal this, because a constant answer is perfectly stable. Agreement across samples (Wang et al., 2023c) cannot detect it either. Each request is answered once. A correct answer that does not reappear is unreliable, whether the view changed or a second sample would differ (Atil et al., 2024; Ouyang et al., 2023), and verified accuracy discounts both cases.

Sensitivity. *Raw accuracy* is the share of separate must-answer siblings whose returned state is correct. A triad is *differentiated* when its three siblings receive three different answers and *all correct* when all three are right; a constant policy differentiates nothing, and uniform guessing differentiates $2/9$ of triads and gets $1/27$ all correct. For the three open-weight models we also read the model's own reasoning. Each must-answer trace is replayed to the model cut at six points from 0% to 100% of its length, and at each cut we read the probability the model places on each state, over three code tokens that name the states (Appendix C). When the three siblings are treated identically, the probability on the correct state is $1/3$ and the *sibling divergence*, the Jensen–Shannon divergence between the three siblings' distributions normalized to $[0, 1]$, is 0; at the 0% cut it measures how visible the edit is before any reasoning.

Invariance and verified accuracy. A sibling's answer is *verified* when it is correct in all four views: its separate must-answer, its label in the joint forced request, and its label in the two pairwise requests that contain it. *Verified accuracy* is the share of verified siblings, *survival* the share of raw-correct siblings that are verified, and *verified triads* the share of triads whose three siblings are all verified. A random mapping places a sibling correctly in one view with probability $1/3$ and in all three views that show its siblings with probability $1/27$, so if a separate answer's correctness were independent of its mapping, verified accuracy would be raw accuracy divided by 27. The joint and pairwise requests tell the model that the siblings' states differ, which is meant to help it, so verified accuracy, if anything, overstates how consistent a model would be without being told.

Regimes. A rule fixed before the analysis groups the 84 cells: *post-frontier* when the 95% confidence interval of raw accuracy contains $1/3$, *pre-frontier* when its lower end is at least 80%, and *transition* otherwise. The grouping is used only to report medians.

Secondary measures, all against verified outcomes. The *failure analysis* describes where unverified answers go: the entropy of the answer distribution, the share of triads answered with one state for all three siblings, and the matrix from true to returned state. *Abstention* pairs the optional request with the must-answer request on the same sibling: precision and recall take a wrong must-answer as the error, *selective accuracy* (Geifman & El-Yaniv, 2017; Kamath et al., 2020) is the accuracy of the non-abstained answers, and *verified selective accuracy* the share of them that are verified. *Clarification* reports the share of resolving questions (Table 1) over all triads and over the triads whose MULTIPLE sibling is verified. Confidence intervals are bootstrap over whole triads (Appendix A).

5 RESULTS

Seven models answered every request: four API models at medium reasoning effort (GPT-5.6 Terra, Luna and Sol, and Gemini 3.7 Flash) and three open-weight models run locally at temperature 0 (Qwen3.8-27B, Gemma4-26B-A4B and GPT-OSS-20B); prompts, budgets and scoring rules are in Appendix K. We also tested Claude Opus 5, Sonnet 5 and Haiku 4.5 but do not report them: on the larger sizes they keep reasoning until they hit their own maximum output length and are cut off before they answer. GPT-5.6 Sol is the running example, and Figure 3 and Table 2 cover all 84 cells.

5.1 ACCURACY FALLS, AND BEYOND THE FRONTIER THE WORLDS ARE NEVER SEPARATED

Accuracy falls with size in every family: at the smallest size Sol answers 88% to 99% of siblings correctly, and at the largest no model does better than guessing. By the rule of Section 4, 10 of the 84 cells are pre-frontier, 22 transition and 52 post-frontier (Table 6). The first show that the tasks are solvable and the scoring works; the last give a body of answers the models can no longer derive.

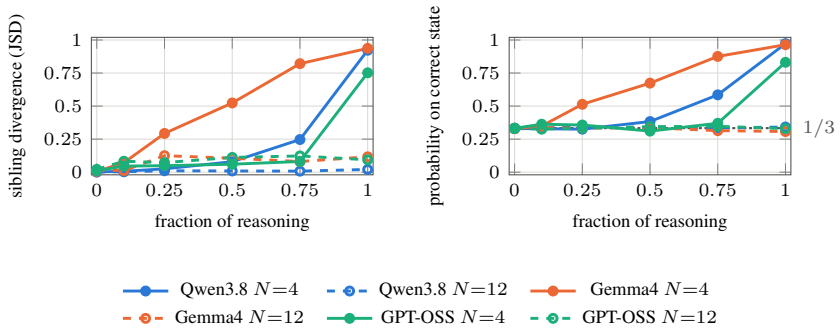


Figure 2: State probabilities along the model's own reasoning, for the three open-weight models on Sudoku (solid lines $N{=}4$, dashed lines $N{=}12$). Left: divergence between the three siblings' state distributions (0 means the model treats them identically). Right: probability on the correct state ($1/3$ when the siblings are treated identically). All curves start at these baseline values before reasoning. The $N{=}4$ curves then separate, but the $N{=}12$ curves never do. All 36 curves are in Appendix J.

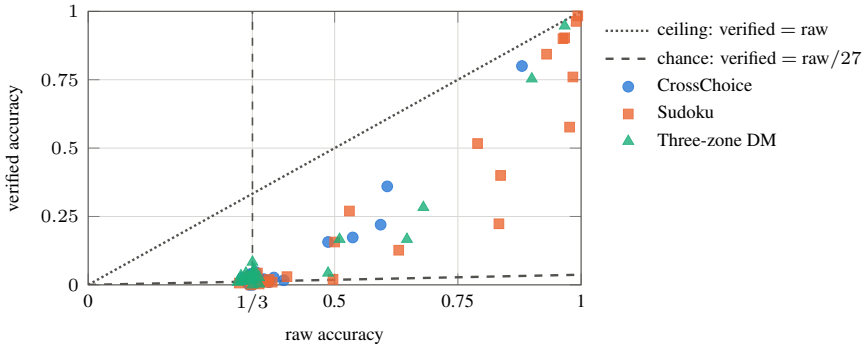


Figure 3: Raw against verified accuracy for all 84 cells, one marker per cell, colored by family. Dotted: the ceiling, where every raw-correct answer survives. Dashed: chance, where a random mapping keeps one raw-correct answer in 27. The vertical line marks $1/3$, the raw accuracy of guessing.

The edit is minimal for the model. Before any reasoning, the three siblings receive almost the same state probabilities in every open-weight cell (sibling divergence at most 0.08, and below 0.03 in 35 of the 36 cells). With the byte-identity and classifier checks of Section 3, this establishes the minimality of the edit. Reasoning then separates the worlds only where the model can still solve the problem (Figure 2). On the smallest Sudoku the probability on the correct state climbs above 0.8, while in post-frontier cells the divergence stays below 0.14 and that probability stays near $1/3$ for the whole trace. The final answers agree. The median share of triads whose siblings receive three different answers is 90% in pre-frontier cells and 0 in post-frontier cells.

5.2 How much raw accuracy overstates: verified accuracy

Where a model can reason, most of its correct answers survive; where it cannot, verified accuracy collapses (Figure 3, Table 2). For Sol, verified accuracy is 80% to 98% at the smallest size of each family and 0.7% to 5.7% at the largest (Table 5). Across models, pre-frontier cells keep a median of 92% of their correct answers and post-frontier cells only 6.0%, close to the $1/27$ of a random mapping; post-frontier verified accuracy is 2.0%, and no triad in any of the 52 cells is verified whole. This is not an artifact of requiring four views: the joint forced view alone keeps a post-frontier median of 35.2%, the $1/3$ of chance, and the survivors do not concentrate in an easy state (Appendix E).

Table 2: Median (range) over the cells of each regime (Section 4).

Metric (%)	**Pre-frontier** ($n = 10$)	**Transition** ($n = 22$)	**Post-frontier** ($n = 52$)
Raw accuracy	96.7 (88–99)	50.5 (30–84)	33.3 (31–38)
Differentiated triads	90.0 (68–98)	9.6 (0–58)	0.0 (0–15)
Verified accuracy	87.2 (58–98)	15.7 (0–52)	2.0 (0–8)
Survival of raw-correct answers	92.2 (59–99)	26.3 (1–65)	6.0 (0–25)
Verified triads	70.0 (27–96)	0.0 (0–25)	0.0 (0–0)
Selective accuracy (raw)	95.8 (84–99)	53.5 (31–89)	34.0 (22–42)
Selective accuracy (verified)	83.3 (57–98)	12.2 (0–44)	1.5 (0–8)

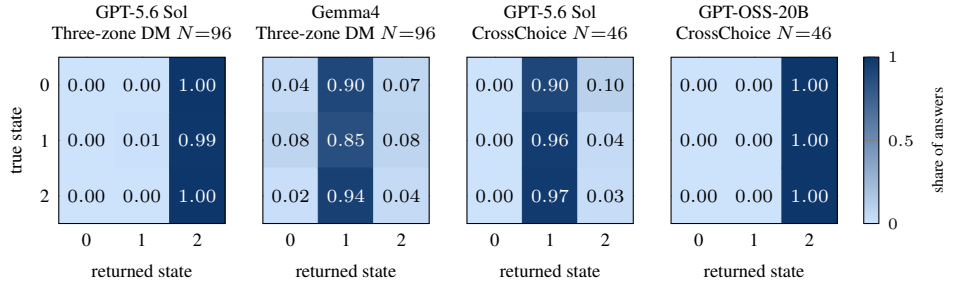


Figure 4: Distribution of unverified answers in four post-frontier cells. Rows are true states and columns are returned states, both written as the size of the answer space (0 = NONE, 1 = UNIQUE, 2 = MULTIPLE). Each row sums to one, and uniform guessing would put $1/3$ in every entry. Each model concentrates on one column, and the column depends on model and family.

In post-frontier cells the correct answers are what a constant policy earns: raw accuracy is 33%, and the returned state carries almost no information about the true state (Cramér's V, a 0-to-1 measure of association, is at most 0.12; Appendix G). This needs no joint or pairwise request. Raw accuracy can also exceed verified accuracy when a model solves a sibling alone but not beside the others: Gemma4 answers 97.7% of 4×4 Sudoku siblings correctly, but only 57.7% are verified. Such cases are few. In pre-frontier cells the joint view alone keeps a median of 99% of correct answers, and several problems in one prompt cost current models little in general (Cheng et al., 2023; Son et al., 2024; Wang et al., 2024b).

Controls. The other checks agree (Appendix F). Relabeling the siblings leaves the answers unchanged in a median of 86.1% of pre-frontier triads but in at most 5.6% of post-frontier ones. Renaming the states re-asks one sibling alone and cannot catch a constant answer: the 15 post-frontier cells that give one state to all three siblings in at least 90% of triads keep a median of 100% of their correct answers under it, against 37% in the joint view. The one-state-each hint is obeyed almost always, yet in post-frontier cells the mapping it asks for stays at chance (18.0% of triads, against 16.7%).

5.3 What remains: a structured fallback that abstention and clarification do not repair

Fallback. The unverified answers are not uniform (Figure 4, Appendix G). In 15 of the 52 post-frontier cells, the model gives one state to all three siblings in at least 90% of triads. The favored state depends on model and family. Terra, for example, falls back to MULTIPLE on Sudoku N=9 and to UNIQUE on N=12. Because the direction changes but the concentration does not, we call this a fallback rather than a fixed bias. The fallback still uses the instance. On CrossChoice at N=33 and 46, Sol, Terra and Luna give the correct unique value for 80% to 98% of the UNIQUE siblings (chance is 50%), yet they also call the systems with two answers or none unique. They compute the assignment, but their verdict on the answer space ignores the instance.

Table 3: The same principle on common benchmarks: one edit, an answer that must change, and an oracle that decides it. Each row lists the outcomes one edit can have; any two siblings whose answers differ already make a test. The last row needs answer spaces decided by people; the others have exact oracles.

Task	One edit	What must change	Who decides
Word problems: GSM8K, MATH (Cobbe et al., 2021; Hendrycks et al., 2021b)	one condition	the number changes, or the problem has no answer or several (Sun et al., 2024; Li et al., 2024)	symbolic solver
Code: HumanEval, MBPP (Chen et al., 2021; Austin et al., 2021; Liu et al., 2023)	one bound or one clause of the specification (Wang et al., 2023b)	the reference solution still passes, must change, or the specification cannot be met	test suite
Text-to-SQL: Spider, BIRD (Yu et al., 2018; Li et al., 2023b)	one word of the question ("at least" to "more than") (Gan et al., 2021; Chang et al., 2023)	the result set changes, becomes empty, or grows	execution on the database
Planning: PlanBench (Valmeekam et al., 2022; 2024)	one precondition or goal	the plan still works, a new plan is needed, or the goal is unreachable	planner
Formal mathematics: miniF2F, PutnamBench (Zheng et al., 2022; Tsoukalas et al., 2024)	one hypothesis	the statement holds, fails, or does not follow	proof assistant
Statutory reasoning: SARA, LegalBench (Holzenberger et al., 2020; Guha et al., 2023)	one fact of the case	the liability changes, disappears, or the rule no longer applies	executable statute
Reading comprehension, contracts: SQuAD 2.0, CUAD (Rajpurkar et al., 2018; Hendrycks et al., 2021a)	one fact of the passage, one word of a clause	the answer changes or becomes unanswerable; an obligation appears or vanishes	people

Abstention. What matters is whether the answers a model keeps can be trusted. Post-frontier abstention rates range from none to 94%, and abstentions do fall on siblings the model would get wrong (a median precision of 68% across the cells that abstain). Yet the answers kept in post-frontier cells have a median raw selective accuracy of 34.0% and a verified selective accuracy of 1.5%: abstention removes some wrong answers but does not leave a trustworthy subset (Appendix H).

Clarification. Told that the target has two values, a model must ask one yes/no question that would settle it. Such resolving questions come in a median of 95% of pre-frontier triads and 4% of post-frontier ones. Knowing that two answers remain does not help: restricted to triads whose MULTIPLE sibling is verified, Sol and Terra resolve 17.6% of 17 and 25% of 8 on Three-zone DM N=96. Unreadable replies count as not resolving (Appendix I).

6 Discussion

Carrying the principle to other problems. The principle needs only a common base, one premise to edit, and an oracle that decides the answer of each sibling, with any number of siblings from two up. Table 3 lists what one edit does to common benchmarks and who can decide the answer. Where an exact oracle exists (a test suite, a database, a planner, a proof assistant, an executable statute), siblings can be produced at scale; where only human judgment exists, the construction still applies at the cost of answer spaces decided by people, as contrast sets do. The visibility checks of Sections 3 and 5.1 (byte-identity outside the edit, a surface classifier at chance, identical sibling distributions before reasoning) apply unchanged, and we suggest that perturbation benchmarks report them as the initial visibility of their edit. The precision of the oracle bounds the precision of the benchmark; that is why we started with constraint problems. Appendix L gives a step-by-step recipe.

What verified accuracy is and is not. It is an external, checkable criterion: the same answer must hold in every view in which it is re-asked (four in OETW). It does not claim to read internal understanding, and it does not assume that the three states are equally hard: each sibling is compared only with its own answer under another view. The fallback direction differs across models and families, so what is measured is a general consequence of failed derivation, not a fixed prior. An answer can

fail it because it never depended on the instance, or because it did not hold once the siblings were shown; both are failures of reliability, and the separate requests tell them apart (Section 5.2).

For evaluation practice. Report, alongside accuracy, whether answers track the edit and survive re-asking, and how they are distributed; and judge abstention by the reliability of the answers a model keeps, not by how often it abstains.

7 LIMITATIONS

The evidence is on three synthetic constraint families, posed in formal notation although each can be restated in natural language; the applicability of the principle elsewhere is a design claim. Larger sizes also lengthen the prompt, so post-frontier failure may include a long-context component that we do not separate (Levy et al., 2024; Liu et al., 2024), and the generator may leave fingerprints that our classifiers do not detect. The open-weight models have much smaller reasoning budgets than the API models and are not compared with them at matched compute, and there is no human baseline. The trajectory readouts are behavioral evidence at fixed points, not a claim about faithfulness (Lanham et al., 2023; Chen et al., 2025): they are conditional probabilities over three code tokens rather than calibrated confidence, and the end of a trace may be the budget. They exist only for the open-weight models, so for the API models the minimality of the edit rests on byte-identity and the classifier audit. Take one correct answer. In principle no test can cleanly show whether the model derived it or would have given it anyway; the two look the same in every request. We therefore claim only that such answers carry no information about the world, not that no reasoning took place.

8 CONCLUSION

Reliable reasoning is not being right on isolated problems. It is keeping the answer when what changed does not matter, and changing it when one key premise does. The second requirement can be made an evaluation principle: pair problems that differ in one premise, let an oracle decide their answers, and count an answer only if it tracks the edit and survives re-asking. One Edit, Three Worlds is our instantiation: the edit decides whether the target has no answer, exactly one, or two, while neither the surface nor the model's initial state reveals it. Verified accuracy built on it is a stricter estimate than raw accuracy, and it gives abstention, failure analysis and clarification a sound reference. On problems they can no longer solve, the models still answer fluently and are sometimes right in isolation, but those answers almost never survive the four views, and where probabilities can be read the three worlds were never separated. A reliable reasoner must know what follows, what is impossible, what remains unresolved, and what information would resolve it. When one premise changes it must update all four together; when nothing that matters changes it must hold.

REPRODUCIBILITY STATEMENT

The release is built from one fixed master seed by a single command and contains the exact public request bytes, the label permutations, the request identifiers and the private scoring oracle; scoring joins model responses to that oracle after the fact, so every number in this paper except the trajectory readouts, which need the model weights, can be recomputed from the released responses. The construct-validity checks, the metric definitions and the bootstrap procedure are stated in Sections 3 and 4 and Appendices A and D, and all tables and data files behind the figures are generated by one script from the published analysis files. Prompts for every request type and one real triad of each family are in Appendix K. Code, the release, all model responses and the analysis files are in the supplementary material.

AI USE STATEMENT

Large language models are the systems under evaluation in this work. In addition, an LLM-based coding assistant was used, under the authors' direction and review, to draft and edit the manuscript text, to write the analysis and typesetting scripts that produce the tables and figures from the computed results, and to check citations. All experimental design, the construction and certification

of the benchmark, and the interpretation of results are the authors' own, and the authors take full responsibility for the content.

REFERENCES

Berk Atil, Sarp Aykent, Alexa Chittams, Lisheng Fu, Rebecca J. Passonneau, Evan Radcliffe, Guru Rajan Rajagopal, Adam Sloan, Tomasz Tudrej, Ferhan Ture, et al. Non-determinism of "deterministic" LLM settings. *arXiv preprint arXiv:2408.04667*, 2024.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

Shuaichen Chang, Jun Wang, Mingwen Dong, Lin Pan, Henghui Zhu, Alexander Hanbo Li, Wuwei Lan, Sheng Zhang, Jiarong Jiang, Joseph Lilien, et al. Dr.Spider: A diagnostic evaluation benchmark towards Text-to-SQL robustness. In *International Conference on Learning Representations (ICLR)*, 2023. arXiv:2301.08881.

Angelica Chen, Jason Phang, Alicia Parrish, Vishakh Padmakumar, Chen Zhao, Samuel R. Bowman, and Kyunghyun Cho. Two failures of self-consistency in the multi-step reasoning of LLMs. *Transactions on Machine Learning Research*, 2024a. arXiv:2305.14279.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Xinyun Chen, Ryan A. Chi, Xuezhi Wang, and Denny Zhou. Premise order matters in reasoning with large language models. In *International Conference on Machine Learning (ICML)*, 2024b. arXiv:2402.08939.

Yanda Chen, Joe Benton, Ansh Radhakrishnan, Jonathan Uesato, Carson Denison, John Schulman, Arushi Somani, Peter Hase, Misha Wagner, Fabien Roger, et al. Reasoning models don't always say what they think. *arXiv preprint arXiv:2505.05410*, 2025.

Zhoujun Cheng, Jungo Kasai, and Tao Yu. Batch prompting: Efficient inference with large language model APIs. In *Findings of the Association for Computational Linguistics: EMNLP*, 2023. arXiv:2301.08721.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Nouha Dziri, Ximing Lu, Melanie Sclar, Xiang Lorraine Li, Liwei Jiang, Bill Yuchen Lin, Peter West, Chandra Bhagavatula, Ronan Le Bras, Jena D. Hwang, et al. Faith and fate: Limits of transformers on compositionality. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. arXiv:2305.18654.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. Measuring and improving consistency in pretrained language models. *Transactions of the Association for Computational Linguistics*, 2021. arXiv:2102.01017.

Benjamin Estermann, Luca A. Lanzendörfer, Yannick Niedermayr, and Roger Wattenhofer. PUZZLES: A benchmark for neural algorithmic reasoning. *arXiv preprint arXiv:2407.00401*, 2024.

Yujian Gan, Xinyun Chen, Qiuping Huang, Matthew Purver, John R. Woodward, Jinxia Xie, and Pengsheng Huang. Towards robustness of Text-to-SQL models against synonym substitution. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2021. arXiv:2106.01065.

Iker García-Ferrero, Begoña Altuna, Javier Álvez, Itziar Gonzalez-Dios, and German Rigau. This is not a dataset: A large negation benchmark to challenge large language models. *arXiv preprint arXiv:2310.15941*, 2023.

Matt Gardner, Yoav Artzi, Victoria Basmova, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, et al. Evaluating models' local decision boundaries via contrast sets. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 2020. arXiv:2004.02709.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems 30*, 2017. arXiv:1705.08500.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2020. arXiv:2004.07780.

Neel Guha, Julian Nyarko, Daniel E. Ho, Christopher Ré, Adam Chilton, Aditya Narayana, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel N. Rockmore, et al. LegalBench: A collaboratively built benchmark for measuring legal reasoning in large language models. In *Advances in Neural Information Processing Systems 36, Datasets and Benchmarks Track*, 2023. arXiv:2308.11462.

Aryan Gulati, Brando Miranda, Eric Chen, Emily Xia, Kai Fronsdal, Bruno de Moraes Dumont, and Sanmi Koyejo. Putnam-AXIOM: A functional and static benchmark for measuring higher level mathematical reasoning in LLMs. *arXiv preprint arXiv:2508.08292*, 2025.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R. Bowman, and Noah A. Smith. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2018. arXiv:1803.02324.

Rishi Hazra, Gabriele Venturato, Pedro Zuidberg Dos Martires, and Luc De Raedt. Can large language models reason? a characterization via 3-SAT. *arXiv preprint arXiv:2408.07215*, 2024.

Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. CUAD: An expert-annotated NLP dataset for legal contract review. In *Advances in Neural Information Processing Systems 34, Datasets and Benchmarks Track*, 2021a. arXiv:2103.06268.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2021b. arXiv:2103.03874.

Nils Holzenberger, Andrew Blair-Stanek, and Benjamin Van Durme. A dataset for statutory reasoning in tax law entailment and question answering. *arXiv preprint arXiv:2005.05257*, 2020.

Myeongjun Erik Jang and Thomas Lukasiewicz. Consistency analysis of ChatGPT. *arXiv preprint arXiv:2303.06273*, 2023.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Amita Kamath, Robin Jia, and Percy Liang. Selective question answering under domain shift. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020. arXiv:2006.09462.

Divyansh Kaushik, Eduard Hovy, and Zachary C. Lipton. Learning the difference that makes a difference with counterfactually-augmented data. In *International Conference on Learning Representations (ICLR)*, 2020. arXiv:1909.12434.

Polina Kirichenko, Mark Ibrahim, Kamalika Chaudhuri, and Samuel J. Bell. AbstentionBench: Reasoning LLMs fail on unanswerable questions. *arXiv preprint arXiv:2506.09038*, 2025.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. CLAM: Selective clarification for ambiguous questions with generative language models. *arXiv preprint arXiv:2212.07769*, 2022.

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. *arXiv preprint arXiv:2307.13702*, 2023.

Mosh Levy, Alon Jacoby, and Yoav Goldberg. Same task, more tokens: the impact of input length on the reasoning performance of large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024. arXiv:2402.14848.

Jiaxuan Li, Lang Yu, and Allyson Ettinger. Counterfactual reasoning: Testing language models' understanding of hypothetical scenarios. In *Findings of the Association for Computational Linguistics: EMNLP*, 2023a. arXiv:2305.16572.

Jinyang Li, Binyuan Hui, Ge Qu, Jiaxi Yang, Binhua Li, Bowen Li, Bailin Wang, Bowen Qin, Ruiying Geng, Nan Huo, et al. Can LLM already serve as a database interface? A big bench for large-scale database grounded text-to-SQLs. *arXiv preprint arXiv:2305.03111*, 2023b.

Qintong Li, Leyang Cui, Xueliang Zhao, Lingpeng Kong, and Wei Bi. GSM-Plus: A comprehensive benchmark for evaluating the robustness of LLMs as mathematical problem solvers. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2024. arXiv:2402.19255.

Bill Yuchen Lin, Ronan Le Bras, Kyle Richardson, Ashish Sabharwal, Radha Poovendran, Peter Clark, and Yejin Choi. ZebraLogic: On the scaling limits of LLMs for logical reasoning. *arXiv preprint arXiv:2502.01100*, 2025.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by ChatGPT really correct? rigorous evaluation of large language models for code generation. In *Advances in Neural Information Processing Systems 36*, 2023. arXiv:2305.01210.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 2024. arXiv:2307.03172.

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 *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2410.05229.

Moran Mizrahi, Guy Kaplan, Dan Malkin, Rotem Dror, Dafna Shahaf, and Gabriel Stanovsky. State of what art? a call for multi-prompt LLM evaluation. *Transactions of the Association for Computational Linguistics*, 2024. arXiv:2401.00595.

Marianna Nezhurina, Lucia Cipolina-Kun, Mehdi Cherti, and Jenia Jitsev. Alice in wonderland: Simple tasks showing complete reasoning breakdown in state-of-the-art large language models. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2406.02061.

Jialin Ouyang. TreeCut: A synthetic unanswerable math word problem dataset for LLM hallucination evaluation. *arXiv preprint arXiv:2502.13442*, 2025.

Shuyin Ouyang, Jie M. Zhang, Mark Harman, and Meng Wang. An empirical study of the non-determinism of ChatGPT in code generation. *arXiv preprint arXiv:2308.02828*, 2023.

Alicia Parrish, Angelica Chen, Nikita Nangia, Vishakh Padmakumar, Jason Phang, Jana Thompson, Phu Mon Htut, and Samuel R. Bowman. BBQ: A hand-built bias benchmark for question answering. In *Findings of the Association for Computational Linguistics: ACL 2022*, 2022. arXiv:2110.08193.

Pouya Pezeshkpour and Estevam Hruschka. Large language models sensitivity to the order of options in multiple-choice questions. In *Findings of the Association for Computational Linguistics: NAACL*, 2024. arXiv:2308.11483.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics (*SEM)*, 2018. arXiv:1805.01042.

Harsh Raj, Domenic Rosati, and Subhabrata Majumdar. Measuring reliability of large language models through semantic consistency. *arXiv preprint arXiv:2211.05853*, 2022.

Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018. arXiv:1806.03822.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020. arXiv:2005.04118.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. Quantifying language models' sensitivity to spurious features in prompt design or: How I learned to start worrying about prompt formatting. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2310.11324.

Jeffrey Seely, Yuki Imajuku, Tianyu Zhao, Edoardo Cetin, and Llion Jones. Sudoku-Bench: Evaluating creative reasoning with Sudoku variants. *arXiv preprint arXiv:2505.16135*, 2025.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R. Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R. Johnston, et al. Towards understanding sycophancy in language models. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2310.13548.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H. Chi, Nathanael Schärli, and Denny Zhou. Large language models can be easily distracted by irrelevant context. In *International Conference on Machine Learning (ICML)*, 2023. arXiv:2302.00093.

Guijin Son, Sangwon Baek, Sangdae Nam, Ilgyun Jeong, and Seungone Kim. Multi-task inference: Can large language models follow multiple instructions at once? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024. arXiv:2402.11597.

Zafir Stojanovski, Oliver Stanley, Joe Sharratt, Richard Jones, Abdulhakeem Adefioye, Jean Kaddour, and Andreas Köpf. Reasoning Gym: Reasoning environments for reinforcement learning with verifiable rewards. *arXiv preprint arXiv:2505.24760*, 2025.

Yuhong Sun, Zhangyue Yin, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Hui Zhao. Benchmarking hallucination in large language models based on unanswerable math word problem. In *Proceedings of the Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING)*, 2024. arXiv:2403.03558.

Jianheng Tang, Qifan Zhang, Yuhan Li, Nuo Chen, and Jia Li. GraphArena: Evaluating and exploring large language models on graph computation. *arXiv preprint arXiv:2407.00379*, 2024.

George Tsoukalas, Jasper Lee, John Jennings, Jimmy Xin, Michelle Ding, Michael Jennings, Amitayush Thakur, and Swarat Chaudhuri. PutnamBench: Evaluating neural theorem-provers on the Putnam mathematical competition. In *Advances in Neural Information Processing Systems 37, Datasets and Benchmarks Track*, 2024. arXiv:2407.11214.

Karthik Valmeekam, Matthew Marquez, Alberto Olmo, Sarath Sreedharan, and Subbarao Kambhampati. PlanBench: An extensible benchmark for evaluating large language models on planning and reasoning about change. *arXiv preprint arXiv:2206.10498*, 2022.

Karthik Valmeekam, Kaya Stechly, and Subbarao Kambhampati. LLMs still can't plan; can LRMs? A preliminary evaluation of OpenAI's o1 on PlanBench. *arXiv preprint arXiv:2409.13373*, 2024.

Heng Wang, Shangbin Feng, Tianxing He, Zhaoxuan Tan, Xiaochuang Han, and Yulia Tsvetkov. Can language models solve graph problems in natural language? In *Advances in Neural Information Processing Systems 36*, 2023a. arXiv:2305.10037.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. Large language models are not fair evaluators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024a. arXiv:2305.17926.

Shiqi Wang, Zheng Li, Haifeng Qian, Chenghao Yang, Zijian Wang, Mingyue Shang, Varun Kumar, Samson Tan, Baishakhi Ray, Parminder Bhatia, et al. ReCode: Robustness evaluation of code generation models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023b. arXiv:2212.10264.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations (ICLR)*, 2023c. arXiv:2203.11171.

Zhengxiang Wang, Jordan Kodner, and Owen Rambow. Evaluating LLMs with multiple problems at once. *arXiv preprint arXiv:2406.10786*, 2024b.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8, 2020. arXiv:1912.00582.

Anjiang Wei, Yuheng Wu, Yingjia Wan, Tarun Suresh, Huanmi Tan, Zhanke Zhou, Sanmi Koyejo, Ke Wang, and Alex Aiken. SATBench: Benchmarking LLMs' logical reasoning via automated puzzle generation from SAT formulas. *arXiv preprint arXiv:2505.14615*, 2025.

Zhaofeng Wu, Linlu Qiu, Alexis Ross, Ekin Akyürek, 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 the 2024 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2024. arXiv:2307.02477.

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, et al. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-SQL task. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2018. arXiv:1809.08887.

Tong Zhang, Peixin Qin, Yang Deng, Chen Huang, Wenqiang Lei, Junhong Liu, Dingnan Jin, Hongru Liang, and Tat-Seng Chua. CLAMBER: A benchmark of identifying and clarifying ambiguous information needs in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024. arXiv:2405.12063.

Tony Z. Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning (ICML)*, 2021. arXiv:2102.09690.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. Large language models are not robust multiple choice selectors. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2309.03882.

Kunhao Zheng, Jesse Michael Han, and Stanislas Polu. miniF2F: A cross-system benchmark for formal Olympiad-level mathematics. In *International Conference on Learning Representations (ICLR)*, 2022. arXiv:2109.00110.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, et al. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems 36, Datasets and Benchmarks Track*, 2023. arXiv:2306.05685.

Zhanke Zhou, Xiao Feng, Zhaocheng Zhu, Jiangchao Yao, Sanmi Koyejo, and Bo Han. From passive to active reasoning: Can large language models ask the right questions under incomplete information? *arXiv preprint arXiv:2506.08295*, 2025.

A REGIMES AND FULL PER-MODEL TABLES

Table 6 gives the regime of every cell under the rule of Section 4; Table 2 the medians of each metric by regime; Table 5 the running example with confidence intervals; Table 4 raw and verified accuracy for every cell, and Tables 7 to 13 the headline metrics of every cell, one table per model, with confidence intervals for verified accuracy and survival. All confidence intervals resample whole triads, 10,000 bootstrap draws.

Table 4: Raw accuracy → verified accuracy (%) for every model, family and size. Sizes N_1 to N_4 are CrossChoice 7/20/33/46, Sudoku 4/6/9/12, Three-zone DM (DM) 24/48/72/96. Each cell has 100 triads (300 siblings). Full metrics with confidence intervals are in Appendix A.

Family	N_1	N_2	N_3	N_4	Family	N_1	N_2	N_3	N_4
GPT-5.6 Sol					**Qwen3.8-27B**				
CrossChoice	88 → 80	54 → 17	34 → 5	33 → 4	CrossChoice	36 → 2	33 → 1	34 → 3	36 → 2
Sudoku	99 → 98	96 → 90	40 → 3	33 → 1	Sudoku	98 → 76	63 → 13	37 → 1	33 → 3
DM	97 → 95	51 → 17	33 → 8	34 → 6	DM	49 → 4	33 → 4	33 → 3	32 → 3
GPT-5.6 Terra					**Gemma4-26B**				
CrossChoice	61 → 36	31 → 3	33 → 3	33 → 4	CrossChoice	40 → 2	32 → 1	33 → 0	33 → 0
Sudoku	93 → 84	79 → 52	36 → 1	35 → 0	Sudoku	98 → 58	50 → 16	33 → 1	31 → 1
DM	90 → 75	34 → 4	33 → 3	33 → 3	DM	33 → 3	31 → 1	30 → 1	31 → 1
GPT-5.6 Luna					**GPT-OSS-20B**				
CrossChoice	49 → 16	38 → 3	33 → 3	33 → 2	CrossChoice	33 → 0	33 → 0	33 → 2	33 → 1
Sudoku	97 → 90	84 → 40	37 → 2	34 → 1	Sudoku	83 → 22	50 → 2	36 → 1	33 → 0
DM	68 → 28	32 → 4	33 → 2	33 → 3	DM	31 → 2	33 → 2	31 → 3	33 → 1
Gemini 3.7 Flash									
CrossChoice	59 → 22	36 → 2	35 → 1	37 → 1					
Sudoku	99 → 96	53 → 27	34 → 4	33 → 2					
DM	65 → 17	35 → 3	35 → 0	33 → 0					

Table 5: Running example (GPT-5.6 Sol), in %. Raw acc.: siblings answered correctly on their own. Differentiated: triads whose three siblings received three different answers. Verified acc.: siblings correct in all four views. Survival: share of raw-correct siblings that are verified. Verified triads: triads with all three siblings verified.

Family	N	Raw acc.	Differentiated	Verified acc. [95% CI]	Survival [95% CI]	Verified triads
Sudoku	4	99.3	98	98.3 [96.3, 99.7]	99.0 [97.3, 100.0]	96
Sudoku	12	33.0	0	0.7 [0.0, 1.7]	2.0 [0.0, 5.1]	0
Three-zone DM	24	96.7	90	94.7 [92.0, 97.0]	97.9 [96.2, 99.3]	85
Three-zone DM	96	33.7	0	5.7 [3.3, 8.0]	16.8 [9.9, 24.8]	0
CrossChoice	7	88.0	68	80.0 [75.3, 84.3]	90.9 [87.1, 94.4]	51
CrossChoice	46	33.0	0	4.0 [2.0, 6.3]	12.1 [6.0, 19.0]	0

Table 6: Regime of every cell (pre = pre-frontier, trans = transition, post = post-frontier). Cr = CrossChoice, Su = Sudoku, Th = Three-zone DM.

Model	Cr7	Cr20	Cr33	Cr46	Su4	Su6	Su9	Su12	Th24	Th48	Th72	Th96
GPT-5.6 Sol	pre	trans	post	post	pre	pre	trans	post	pre	trans	post	post
GPT-5.6 Terra	trans	post	post	post	pre	trans	post	post	pre	post	post	post
GPT-5.6 Luna	trans	post	post	post	pre	trans	post	post	trans	trans	post	post
Gemini 3.7 Flash	trans	post	post	post	pre	trans	post	post	trans	post	trans	post
Qwen3.8-27B	post	post	post	post	pre	trans	post	post	trans	post	post	post
Gemma4-26B	trans	post	post	post	pre	trans	post	trans	post	post	trans	post
GPT-OSS-20B	post	post	post	post	trans	trans	post	post	trans	post	post	post

Table 7: GPT-5.6 Sol: headline metrics per cell (%; DM = Three-zone DM). All 3 correct: triads whose three siblings are all correct on their own. Verified acc. and survival use all four views. Resolving query: clarification requests whose question the solver accepts as resolving. Raw-accuracy confidence intervals and the regime of each cell are in Table 6.

Family	N	Raw acc.	All 3 correct	Differ-entiated	Verified acc. [CI]	Survival [CI]	Verified triads	Abstain	Resolving query
CrossChoice	7	88.0	67	68	80.0 [75.3, 84.3]	90.9 [87.1, 94.4]	51	0	97
CrossChoice	20	53.7	1	1	17.3 [13.0, 21.7]	32.3 [24.7, 40.5]	0	0	31
CrossChoice	33	34.0	0	0	4.7 [2.7, 7.0]	13.7 [7.5, 20.4]	0	0	8
CrossChoice	46	33.0	0	0	4.0 [2.0, 6.3]	12.1 [6.0, 19.0]	0	0	7
Sudoku	4	99.3	98	98	98.3 [96.3, 99.7]	99.0 [97.3, 100.0]	96	0	100
Sudoku	6	96.3	89	89	90.0 [85.3, 94.0]	93.4 [89.2, 96.9]	80	0	98
Sudoku	9	40.3	1	1	3.0 [1.0, 5.3]	7.4 [2.6, 12.9]	0	1	54
Sudoku	12	33.0	0	0	0.7 [0.0, 1.7]	2.0 [0.0, 5.1]	0	5	23
DM	24	96.7	90	90	94.7 [92.0, 97.0]	97.9 [96.2, 99.3]	85	0	75
DM	48	51.0	7	7	16.7 [12.3, 21.3]	32.7 [24.5, 41.5]	0	0	21
DM	72	33.3	0	0	8.3 [5.3, 11.3]	25.0 [16.8, 34.0]	0	0	14
DM	96	33.7	0	0	5.7 [3.3, 8.0]	16.8 [9.9, 24.8]	0	0	12

Table 8: GPT-5.6 Terra: headline metrics per cell (%; DM = Three-zone DM). All 3 correct: triads whose three siblings are all correct on their own. Verified acc. and survival use all four views. Resolving query: clarification requests whose question the solver accepts as resolving. Raw-accuracy confidence intervals and the regime of each cell are in Table 6.

Family	N	Raw acc.	All 3 correct	Differ-entiated	Verified acc. [CI]	Survival [CI]	Verified triads	Abstain	Resolving query
CrossChoice	7	60.7	17	17	36.0 [31.7, 40.3]	59.3 [52.1, 66.7]	1	0	83
CrossChoice	20	31.3	0	0	2.7 [1.0, 4.7]	8.5 [3.0, 15.2]	0	0	21
CrossChoice	33	32.7	0	0	2.7 [1.0, 4.7]	8.2 [3.1, 13.7]	0	0	6
CrossChoice	46	33.0	0	0	3.7 [1.7, 5.7]	11.1 [5.2, 17.7]	0	0	6
Sudoku	4	93.0	82	83	84.3 [79.7, 89.0]	90.7 [86.6, 94.5]	64	0	97
Sudoku	6	79.0	50	51	51.7 [44.7, 58.7]	65.4 [57.4, 73.0]	25	0	76
Sudoku	9	36.3	0	0	1.0 [0.0, 2.3]	2.8 [0.0, 6.1]	0	19	31
Sudoku	12	34.7	0	0	0.3 [0.0, 1.0]	1.0 [0.0, 3.1]	0	63	16
DM	24	90.0	74	74	75.3 [69.7, 80.7]	83.7 [78.6, 88.5]	48	0	58
DM	48	33.7	0	0	4.3 [2.3, 6.7]	12.9 [6.9, 19.6]	0	0	13
DM	72	33.3	0	0	2.7 [1.0, 4.7]	8.0 [3.0, 14.0]	0	0	11
DM	96	33.3	0	0	2.7 [1.0, 4.7]	8.0 [3.0, 14.0]	0	0	8

Table 9: GPT-5.6 Luna: headline metrics per cell (%; DM = Three-zone DM). All 3 correct: triads whose three siblings are all correct on their own. Verified acc. and survival use all four views. Resolving query: clarification requests whose question the solver accepts as resolving. Raw-accuracy confidence intervals and the regime of each cell are in Table 6.

Family	N	Raw acc.	All 3 correct	Differ-entiated	Verified acc. [CI]	Survival [CI]	Verified triads	Abstain	Resolving query
CrossChoice	7	48.7	6	8	15.7 [12.0, 19.7]	32.2 [24.5, 40.1]	0	0	55
CrossChoice	20	37.7	0	0	2.7 [1.0, 4.7]	7.1 [2.5, 12.7]	0	0	21
CrossChoice	33	33.0	0	0	3.3 [1.7, 5.3]	10.1 [4.9, 16.3]	0	0	7
CrossChoice	46	32.7	0	0	2.0 [0.7, 3.7]	6.1 [2.0, 11.2]	0	0	5
Sudoku	4	96.7	90	90	90.3 [86.7, 94.0]	93.4 [89.8, 96.6]	76	0	95
Sudoku	6	83.7	58	58	40.0 [32.3, 47.3]	47.8 [39.9, 55.8]	20	0	74
Sudoku	9	36.7	0	0	2.0 [0.7, 3.7]	5.5 [1.8, 10.0]	0	4	19
Sudoku	12	33.7	0	0	0.7 [0.0, 1.7]	2.0 [0.0, 5.1]	0	9	17
DM	24	68.0	26	28	28.3 [23.0, 33.7]	41.7 [34.3, 49.2]	3	0	19
DM	48	32.0	0	0	4.3 [2.3, 6.7]	13.5 [7.2, 20.8]	0	0	6
DM	72	33.3	0	0	2.0 [0.7, 3.7]	6.0 [2.0, 11.0]	0	0	5
DM	96	33.3	0	0	2.7 [1.0, 4.7]	8.0 [3.0, 14.0]	0	0	0

Table 10: Gemini 3.7 Flash: headline metrics per cell (%; DM = Three-zone DM). All 3 correct: triads whose three siblings are all correct on their own. Verified acc. and survival use all four views. Resolving query: clarification requests whose question the solver accepts as resolving. Raw-accuracy confidence intervals and the regime of each cell are in Table 6.

Family	N	Raw acc.	All 3 correct	Differ-entiated	Verified acc. [CI]	Survival [CI]	Verified triads	Abstain	Resolving query
CrossChoice	7	59.3	12	13	22.0 [16.7, 27.7]	37.1 [28.6, 45.4]	2	1	35
CrossChoice	20	35.7	1	15	2.0 [0.7, 3.7]	5.6 [1.8, 10.2]	0	22	6
CrossChoice	33	34.7	3	10	1.3 [0.3, 2.7]	3.8 [0.9, 7.8]	0	29	4
CrossChoice	46	36.7	4	12	1.0 [0.0, 2.3]	2.7 [0.0, 6.1]	0	37	4
Sudoku	4	99.0	97	97	96.3 [94.0, 98.3]	97.3 [95.0, 99.0]	90	0	87
Sudoku	6	53.0	10	12	27.0 [22.0, 32.0]	50.9 [42.4, 59.6]	1	1	51
Sudoku	9	34.3	0	0	4.3 [2.3, 6.7]	12.6 [6.7, 19.2]	0	1	20
Sudoku	12	33.3	0	0	2.0 [0.7, 3.7]	6.0 [2.0, 11.0]	0	6	18
DM	24	64.7	29	34	16.7 [12.7, 20.7]	25.8 [19.8, 31.9]	0	6	0
DM	48	34.7	0	0	2.7 [1.0, 4.7]	7.7 [2.9, 13.0]	0	12	8
DM	72	34.7	0	0	0.3 [0.0, 1.0]	1.0 [0.0, 2.9]	0	17	7
DM	96	33.3	0	0	0.0 [0.0, 0.0]	0.0 [0.0, 0.0]	0	25	0

Table 11: Qwen3.8-27B: headline metrics per cell (%; DM = Three-zone DM). All 3 correct: triads whose three siblings are all correct on their own. Verified acc. and survival use all four views. Resolving query: clarification requests whose question the solver accepts as resolving. Raw-accuracy confidence intervals and the regime of each cell are in Table 6.

Family	N	Raw acc.	All 3 correct	Differ-entiated	Verified acc. [CI]	Survival [CI]	Verified triads	Abstain	Resolving query
CrossChoice	7	35.7	0	0	2.0 [0.7, 3.7]	5.6 [1.8, 10.2]	0	14	1
CrossChoice	20	32.7	0	0	0.7 [0.0, 1.7]	2.0 [0.0, 5.2]	0	52	3
CrossChoice	33	34.0	1	5	3.0 [1.0, 5.3]	8.8 [3.2, 15.5]	0	57	0
CrossChoice	46	35.7	0	2	1.7 [0.3, 3.3]	4.7 [0.9, 9.1]	0	33	1
Sudoku	4	98.3	95	95	76.0 [70.0, 81.7]	77.3 [71.3, 83.1]	55	1	95
Sudoku	6	63.0	20	22	12.7 [8.7, 17.0]	20.1 [14.1, 26.5]	1	14	26
Sudoku	9	37.3	1	1	1.0 [0.0, 2.3]	2.7 [0.0, 6.0]	0	40	5
Sudoku	12	32.7	0	0	3.0 [1.3, 5.0]	9.2 [4.0, 15.2]	0	49	1
DM	24	48.7	12	18	4.3 [2.0, 7.0]	8.9 [4.4, 14.1]	0	41	2
DM	48	33.3	0	1	4.3 [2.3, 6.7]	13.0 [6.9, 19.6]	0	36	2
DM	72	33.3	0	0	3.3 [1.3, 5.3]	10.0 [4.3, 16.0]	0	54	0
DM	96	32.0	0	0	2.7 [1.0, 4.7]	8.3 [3.2, 14.3]	0	49	0

Table 12: Gemma4-26B: headline metrics per cell (%; DM = Three-zone DM). All 3 correct: triads whose three siblings are all correct on their own. Verified acc. and survival use all four views. Resolving query: clarification requests whose question the solver accepts as resolving. Raw-accuracy confidence intervals and the regime of each cell are in Table 6.

Family	N	Raw acc.	All 3 correct	Differ-entiated	Verified acc. [CI]	Survival [CI]	Verified triads	Abstain	Resolving query
CrossChoice	7	39.7	1	4	1.7 [0.3, 3.3]	4.2 [0.9, 8.0]	0	0	0
CrossChoice	20	32.3	0	0	0.7 [0.0, 1.7]	2.1 [0.0, 5.3]	0	1	0
CrossChoice	33	32.7	0	1	0.0 [0.0, 0.0]	0.0 [0.0, 0.0]	0	9	0
CrossChoice	46	32.7	0	0	0.3 [0.0, 1.0]	1.0 [0.0, 3.2]	0	5	0
Sudoku	4	97.7	93	94	57.7 [51.0, 64.3]	59.0 [52.2, 65.5]	27	0	31
Sudoku	6	50.0	9	10	15.7 [11.3, 20.0]	31.3 [23.5, 39.2]	0	1	12
Sudoku	9	33.3	0	0	1.0 [0.0, 2.3]	3.0 [0.0, 6.9]	0	2	4
Sudoku	12	30.7	0	0	0.7 [0.0, 1.7]	2.2 [0.0, 5.6]	0	4	4
DM	24	33.0	0	0	3.3 [1.3, 5.3]	10.1 [4.8, 16.3]	0	0	0
DM	48	31.0	0	0	1.3 [0.3, 2.7]	4.3 [1.0, 8.8]	0	1	0
DM	72	30.3	0	0	1.0 [0.0, 2.3]	3.3 [0.0, 7.4]	0	1	0
DM	96	31.0	1	1	1.3 [0.3, 2.7]	4.3 [1.0, 8.9]	0	1	0

Table 13: GPT-OSS-20B: headline metrics per cell (%; DM = Three-zone DM). All 3 correct: triads whose three siblings are all correct on their own. Verified acc. and survival use all four views. Resolving query: clarification requests whose question the solver accepts as resolving. Raw-accuracy confidence intervals and the regime of each cell are in Table 6.

Family	N	Raw acc.	All 3 correct	Differ-entiated	Verified acc. [CI]	Survival [CI]	Verified triads	Abstain	Resolving query
CrossChoice	7	33.3	0	0	0.3 [0.0, 1.0]	1.0 [0.0, 3.0]	0	1	23
CrossChoice	20	33.3	0	0	0.0 [0.0, 0.0]	0.0 [0.0, 0.0]	0	5	1
CrossChoice	33	33.3	0	0	2.0 [0.7, 3.7]	6.0 [2.0, 11.0]	0	4	4
CrossChoice	46	33.3	0	0	1.3 [0.3, 2.7]	4.0 [1.0, 8.0]	0	2	1
Sudoku	4	83.3	55	56	22.3 [18.0, 27.0]	26.8 [21.6, 32.0]	0	1	18
Sudoku	6	49.7	8	9	2.0 [0.3, 4.0]	4.0 [0.7, 8.1]	0	19	6
Sudoku	9	35.7	0	0	1.0 [0.0, 2.3]	2.8 [0.0, 6.3]	0	63	1
Sudoku	12	33.0	0	0	0.0 [0.0, 0.0]	0.0 [0.0, 0.0]	0	94	2
DM	24	31.3	0	0	1.7 [0.3, 3.3]	5.3 [1.1, 10.3]	0	10	0
DM	48	33.0	0	0	2.0 [0.7, 3.7]	6.1 [2.0, 11.2]	0	20	0
DM	72	31.0	0	0	3.3 [1.3, 5.3]	10.8 [4.7, 17.4]	0	10	0
DM	96	32.7	0	0	1.3 [0.3, 2.7]	4.1 [1.0, 8.3]	0	20	0

B GENERATION AND CERTIFICATION, PER FAMILY

All three generators are seeded from one master seed with a fixed offset per family, so a release is reproducible byte for byte; no model output is consulted at any point. Public prompts contain no state, answer or certificate; all scoring is done by the solver. Every accepted triad is then recomputed by a checker that shares no solving code with the generator.

CrossChoice. A constraint $\mathrm{CrossChoice}((l_1 \vee l_2 \vee l_3), (r_1 \vee r_2 \vee r_3))$ over six distinct variables requires both disjunctions to hold; one constraint is one atomic edit. The base is a satisfiable Horn core of strict 3-clauses: variables are drawn with weights decreasing in their current degree, each side of a new constraint is a Horn clause (all-negative with probability 0.22, otherwise one positive head), duplicates are rejected, and the constraint count is about $0.18\,n^2$ for n variables. The core, the target and the three pivots are then transformed by one random variable permutation and random literal sign flips, so the public formula is not Horn as written; it is Horn after deleting one constraint and renaming, which is what keeps the public class polynomial. For a target t, the generator precomputes for every pair of literals (a, b) the set of values of t that remain feasible when both are true (one Horn closure per value). A candidate constraint (A, B) has answer set equal to the union over its nine pairs. Because each half of a candidate on its own must leave both target values open, no single clause gives the state away. A NONE candidate needs all nine pairs infeasible while each half alone leaves both values; a UNIQUE candidate needs the union to be one value with each half alone leaving both; a MULTIPLE candidate needs both values and a designed resolver variable q in its left half such that q true and q false each pin t to a different value, and the final sibling must have one or two resolving variables in total. Candidates are ranked by the depth and size of the Horn closure needed to exclude the wrong value, the ten best per role are kept, candidates whose full formula is still Horn-renamable are discarded, and the triple with the smallest difference in pivot variable degrees is chosen. The independent checker recomputes the three answer sets and the resolving query with a separate recognizer and closure.

Rectangular Sudoku. Order N is composite with $h \times w$ boxes (2×2, 2×3, 3×3 and 3×4 for the four sizes). A full grid is sampled by a randomized minimum-remaining-values search. A pivot square P is chosen; the UNIQUE value is the grid's value at P and the NONE value another symbol. Clue deletion starts from P and every peer of P whose value equals the NONE value, so that the NONE sibling never shows a duplicate in a row, column or box. Clues are then deleted in random order, and a deletion is kept only if the NONE sibling still has no completion. After each kept deletion, every other symbol is tried as the MULTIPLE value: if the pivot filled with it admits 2 to 256 completions, a square that takes exactly two values across those completions and exactly one value in the UNIQUE sibling can serve as the target T, together with a clarification square whose two possible values each determine T. The target is therefore chosen after the pivot values, not before. Further deletion continues while all three conditions and the clarification hold. Among the exact candidates found within the budget, a human-technique solver trace ranks difficulty and the hardest is kept. The generator's search is a bitmask backtracking solver in C++ that never treats an exhausted budget as a proof; the independent checker re-derives all three target sets and the clarification with a Z3 encoding.

Three-zone DM. The graph $G = (L \cup R, E)$ has $|L| = |R| = N$. Fix the near-perfect matching $M_0 = \{(L_i, R_i) : i = 1, \ldots, N-1\}$, leaving L_0 and R_0 unmatched, and contract each matched pair to one vertex of a digraph with source $s = L_0$ and sink $z = R_0$; G has a perfect matching exactly when s reaches z. The paired vertices are split into three zones of equal size: zone A, reachable from s; zone B, which reaches z; and zone X, a structural clone of B whose terminals lead to a neutral anchor instead of z. Arcs inside and between zones are near-regular (out-degree 9, return degree 3), and the unmatched source and sink are given the same degree floor as ordinary vertices. The base has a single Hall deficit, $|N(S)| = |S| - 1$ for $S = \{L_0\} \cup \{\text{left vertices of zone A}\}$, and therefore no perfect matching. A missing edge from A to B repairs the deficit and the sibling has perfect matchings; a missing edge from A to X does not. An exact scan over such edges computes, for every vertex, the set of partners across all perfect matchings (0 for the A→X pivots, 1 or 2 for the A→B pivots), and the target and the three pivots are then chosen so that the endpoints of the pivots have matching degree profiles. The public prompt lists base and pivot edges in one shuffled

available-edge list, so the pivot is not marked. The independent checker recomputes the partner sets by matching and alternating-path reachability, and checks the clarification query.

Label schedule. Within each family and size the physical roles of the 100 triads are mapped to the labels A, B, C by a deterministic schedule that uses all six permutations 17, 17, 17, 17, 16 and 16 times; a triad keeps the same mapping across all its requests. The two label orders of each pairwise request are balanced 50/50 within each family and size. The first 18 triads of each family and size additionally receive the five other label orders under the joint forced request and six temporary codebooks on one scheduled sibling, chosen without reference to any model outcome.

C TRAJECTORY READOUT

For each separate must-answer trace of the three open-weight models, the model's own reasoning is replayed to the same model cut at 0, 10, 25, 50, 75 and 100% of its token length, followed by a request to name the state as one of three code tokens. The three tokens' raw log-probabilities are read in a single forward pass and normalized over the three; the result is a conditional distribution over the codes, not a calibrated confidence over all text. The probability on the correct state is $1/3$ when the three siblings receive the same distribution; the sibling divergence is the Jensen–Shannon divergence between the three siblings' distributions normalized to $[0, 1]$. Traces that ended at the budget are included and reported as a separate stratum in Table 23. Every readout is bound to the original response by hash.

D CONSTRUCT VALIDITY

Every triad in the release was recomputed by an independent checker: the three answer sets $A(v)$, the value of the UNIQUE sibling, the two values of the MULTIPLE sibling and the clarification oracle. All 1,200 triads pass. Table 14 reports the byte-identity check and the surface-feature classifier audit for each family and size. Below a one-line header that names the sibling, the three public prompts are byte-identical outside one contiguous pivot span for every family and size: one symbol in Sudoku, one edge in Three-zone DM, one constraint in CrossChoice. This is the check that hypothesis-only baselines make for shortcuts (Gururangan et al., 2018; Poliak et al., 2018; Geirhos et al., 2020). The classifier is a multinomial logistic regression on hashed features (2,048 dimensions for character n-grams of the pivot span, with span length, offset and digit counts; 4,096 for word n-grams of the full prompt; the A/B/C label; and all of these combined), trained and tested under 10-fold cross-validation grouped by triad, so that the three siblings of a triad always fall in the same fold. An exact-span lookup baseline predicts the state of a test sibling from training siblings with the same pivot span. Confidence intervals are bootstrap over triads. The label-only classifier is below chance for every family and size because the label schedule is balanced by construction. No feature set comes close to a usable shortcut: the largest accuracy of any single feature set for any family and size is 40.0% (Sudoku $N{=}4$, pivot-span n-grams and exact-span lookup), three of the 48 single-feature-set tests sit a few points above chance at the 95% level, and the combined classifier is within the interval of chance for all twelve combinations of family and size.

Table 14: Surface-feature shortcut audit. Identical: the three siblings are byte-identical outside one contiguous pivot span. Classifier accuracy (%) with 95% triad-bootstrap CI under triad-grouped 10-fold cross-validation; chance is 33.3.

Family	N	Identical	Span chars	Label only	Span n-grams	Full-prompt n-grams	All features	Span lookup
CrossChoice	7	yes	54–88	22.7 [18.0, 27.7]	31.0 [27.7, 34.0]	33.3 [32.3, 34.3]	32.0 [29.3, 34.7]	33.7 [28.7, 38.7]
CrossChoice	20	yes	62–92	21.3 [16.3, 27.0]	32.3 [27.7, 37.3]	33.3 [32.3, 34.3]	35.0 [30.7, 39.7]	38.7 [32.3, 44.7]
CrossChoice	33	yes	66–88	17.7 [13.7, 21.7]	36.7 [32.0, 41.3]	33.7 [33.3, 34.3]	34.7 [30.7, 38.7]	29.3 [24.7, 34.0]
CrossChoice	46	yes	66–92	21.7 [16.7, 26.7]	37.3 [32.0, 42.3]	33.3 [33.3, 33.3]	35.7 [30.7, 40.7]	34.0 [29.3, 38.7]
Sudoku	4	yes	1–1	24.3 [19.3, 30.0]	40.0 [34.3, 46.3]	35.3 [31.3, 39.3]	35.7 [31.0, 40.7]	40.0 [34.7, 46.3]
Sudoku	6	yes	1–1	16.0 [12.0, 20.3]	35.7 [29.7, 41.7]	33.0 [30.7, 35.3]	34.3 [31.3, 37.3]	37.3 [31.3, 43.3]
Sudoku	9	yes	1–1	17.7 [13.3, 22.3]	28.7 [23.7, 34.0]	34.7 [32.0, 37.3]	33.3 [31.0, 35.7]	31.7 [26.0, 37.0]
Sudoku	12	yes	0–2	21.7 [17.0, 26.7]	25.3 [20.3, 30.3]	35.0 [32.7, 37.7]	35.7 [33.3, 38.0]	26.0 [21.0, 31.0]
Three-zone DM	24	yes	6–8	22.0 [17.3, 27.0]	31.0 [26.7, 35.7]	34.0 [32.3, 35.7]	32.0 [29.3, 34.7]	35.0 [29.7, 40.3]
Three-zone DM	48	yes	6–8	19.3 [14.7, 24.0]	33.3 [28.7, 38.0]	33.3 [32.3, 34.3]	36.3 [32.0, 41.0]	33.7 [28.3, 39.3]
Three-zone DM	72	yes	6–8	20.3 [15.0, 26.0]	39.3 [34.7, 44.7]	33.0 [32.3, 33.3]	36.0 [32.0, 40.0]	35.3 [30.0, 41.0]
Three-zone DM	96	yes	7–8	16.3 [12.3, 20.7]	27.3 [22.7, 32.0]	32.7 [31.3, 34.0]	31.3 [26.7, 35.7]	39.0 [33.3, 44.7]

E PER-STATE RECALL AND SURVIVAL

The survivors of the four views do not concentrate in an easy state. For example, Sol on CrossChoice N=46 answers UNIQUE almost always, is right on 96% of the UNIQUE siblings, and 12.5% of those answers survive.

Table 15: Per-state recall of the separate must-answer and four-view survival of the raw-correct siblings, in %; states are written as the size of the answer space (0 = NONE, 1 = UNIQUE, 2 = MULTIPLE). Survival is undefined when no sibling of that state was correct.

Model	Family	N	Recall 0	Recall 1	Recall 2	Survival 0	Survival 1	Survival 2
GPT-5.6 Sol	CrossChoice	7	77	91	96	79	92	99
GPT-5.6 Sol	CrossChoice	20	1	72	88	0	31	34
GPT-5.6 Sol	CrossChoice	33	0	80	22	–	18	0
GPT-5.6 Sol	CrossChoice	46	0	96	3	–	12	0
GPT-5.6 Sol	Sudoku	4	99	99	100	99	99	99
GPT-5.6 Sol	Sudoku	6	96	96	97	96	93	92
GPT-5.6 Sol	Sudoku	9	3	20	98	0	20	5
GPT-5.6 Sol	Sudoku	12	0	3	96	–	0	2
GPT-5.6 Sol	Three-zone DM	24	97	94	99	96	99	99
GPT-5.6 Sol	Three-zone DM	48	20	34	99	30	32	33
GPT-5.6 Sol	Three-zone DM	72	0	1	99	–	100	24
GPT-5.6 Sol	Three-zone DM	96	0	1	100	–	0	17
GPT-5.6 Terra	CrossChoice	7	26	59	97	23	44	78
GPT-5.6 Terra	CrossChoice	20	0	40	54	–	10	7
GPT-5.6 Terra	CrossChoice	33	0	94	4	–	9	0
GPT-5.6 Terra	CrossChoice	46	1	98	0	0	11	–
GPT-5.6 Terra	Sudoku	4	91	94	94	95	93	85
GPT-5.6 Terra	Sudoku	6	68	79	90	59	71	66
GPT-5.6 Terra	Sudoku	9	0	18	91	–	11	1
GPT-5.6 Terra	Sudoku	12	0	56	48	–	2	0
GPT-5.6 Terra	Three-zone DM	24	92	83	95	84	86	82
GPT-5.6 Terra	Three-zone DM	48	0	4	97	–	0	13
GPT-5.6 Terra	Three-zone DM	72	0	0	100	–	–	8
GPT-5.6 Terra	Three-zone DM	96	0	0	100	–	–	8
GPT-5.6 Luna	CrossChoice	7	17	43	86	12	19	43
GPT-5.6 Luna	CrossChoice	20	0	40	73	–	12	4
GPT-5.6 Luna	CrossChoice	33	0	94	5	–	11	0
GPT-5.6 Luna	CrossChoice	46	0	98	0	–	6	–
GPT-5.6 Luna	Sudoku	4	95	95	100	96	94	91
GPT-5.6 Luna	Sudoku	6	75	85	91	45	52	46
GPT-5.6 Luna	Sudoku	9	0	31	79	–	6	5
GPT-5.6 Luna	Sudoku	12	0	47	54	–	2	2
GPT-5.6 Luna	Three-zone DM	24	68	42	94	40	36	46
GPT-5.6 Luna	Three-zone DM	48	0	0	96	–	–	14
GPT-5.6 Luna	Three-zone DM	72	0	0	100	–	–	6
GPT-5.6 Luna	Three-zone DM	96	0	0	100	–	–	8
Gemini 3.7 Flash	CrossChoice	7	20	70	88	40	40	34
Gemini 3.7 Flash	CrossChoice	20	51	24	32	4	12	3
Gemini 3.7 Flash	CrossChoice	33	50	41	13	4	5	0
Gemini 3.7 Flash	CrossChoice	46	38	60	12	0	5	0
Gemini 3.7 Flash	Sudoku	4	97	100	100	100	96	96
Gemini 3.7 Flash	Sudoku	6	17	53	89	41	51	53
Gemini 3.7 Flash	Sudoku	9	1	98	4	0	13	0
Gemini 3.7 Flash	Sudoku	12	0	100	0	–	6	–
Gemini 3.7 Flash	Three-zone DM	24	56	68	70	48	18	16
Gemini 3.7 Flash	Three-zone DM	48	4	11	89	25	9	7
Gemini 3.7 Flash	Three-zone DM	72	1	3	100	0	0	1
Gemini 3.7 Flash	Three-zone DM	96	0	0	100	–	–	0
Qwen3.8-27B	CrossChoice	7	0	54	53	–	4	8
Qwen3.8-27B	CrossChoice	20	0	68	30	–	0	7
Qwen3.8-27B	CrossChoice	33	3	51	48	0	12	6
Qwen3.8-27B	CrossChoice	46	7	11	89	0	0	6
Qwen3.8-27B	Sudoku	4	100	99	96	79	77	76
Qwen3.8-27B	Sudoku	6	56	87	46	14	29	11
Qwen3.8-27B	Sudoku	9	1	46	65	0	4	2
Qwen3.8-27B	Sudoku	12	0	19	79	–	5	10
Qwen3.8-27B	Three-zone DM	24	44	36	66	9	0	14
Qwen3.8-27B	Three-zone DM	48	2	7	91	0	0	14
Qwen3.8-27B	Three-zone DM	72	0	17	83	–	12	10
Qwen3.8-27B	Three-zone DM	96	0	25	71	–	8	8
Gemma4-26B	CrossChoice	7	7	81	31	14	5	0
Gemma4-26B	CrossChoice	20	0	75	22	–	1	5
Gemma4-26B	CrossChoice	33	4	51	43	0	0	0
Gemma4-26B	CrossChoice	46	1	68	29	0	1	0
Gemma4-26B	Sudoku	4	98	100	95	62	66	48
Gemma4-26B	Sudoku	6	34	96	20	41	31	15
Gemma4-26B	Sudoku	9	0	95	5	–	3	0

Model	Family	N	Recall 0	Recall 1	Recall 2	Survival 0	Survival 1	Survival 2
Gemma4-26B	Sudoku	12	0	90	2	–	2	0
Gemma4-26B	Three-zone DM	24	2	95	2	0	11	0
Gemma4-26B	Three-zone DM	48	4	86	3	0	5	0
Gemma4-26B	Three-zone DM	72	3	86	2	33	2	0
Gemma4-26B	Three-zone DM	96	4	85	4	0	5	0
GPT-OSS-20B	CrossChoice	7	0	0	100	–	–	1
GPT-OSS-20B	CrossChoice	20	0	0	100	–	–	0
GPT-OSS-20B	CrossChoice	33	0	0	100	–	–	6
GPT-OSS-20B	CrossChoice	46	0	0	100	–	–	4
GPT-OSS-20B	Sudoku	4	92	98	60	48	5	30
GPT-OSS-20B	Sudoku	6	16	63	70	0	8	1
GPT-OSS-20B	Sudoku	9	1	25	81	0	4	2
GPT-OSS-20B	Sudoku	12	0	20	79	–	0	0
GPT-OSS-20B	Three-zone DM	24	1	1	92	0	0	5
GPT-OSS-20B	Three-zone DM	48	2	11	86	0	0	7
GPT-OSS-20B	Three-zone DM	72	0	7	86	–	0	12
GPT-OSS-20B	Three-zone DM	96	0	8	90	–	0	4

F INVARIANCE CONTROLS

Under a temporary codebook, post-frontier accuracy stays at chance (23.1% to 40.7%, median 33.3%). On Three-zone DM N=96, Terra, Luna and Sol keep all of their correct answers under the new names: the constant MULTIPLE is anchored to the meaning of the state, not to its token.

Table 16: Joint and pairwise requests, in % (DM = Three-zone DM). Legal: the assignment obeys what the model was told (one state each, or two different states). Exact: every pivot mapped to its true state. Free-must: joint, must-answer, without the one-state-each hint; forced: joint with the hint; pair: pairwise. Chance: exact $1/6$ for forced and pair, pair sibling accuracy $1/3$.

Model	Family	N	Free-must legal	Free-must exact	Forced legal	Forced exact	Pair legal	Pair exact	Pair sibling acc.
GPT-5.6 Sol	CrossChoice	7	73	73	100	100	100	91	95
GPT-5.6 Sol	CrossChoice	20	20	17	100	47	100	43	61
GPT-5.6 Sol	CrossChoice	33	17	6	100	26	100	25	43
GPT-5.6 Sol	CrossChoice	46	11	3	100	22	100	22	38
GPT-5.6 Sol	Sudoku	4	97	97	100	100	100	99	100
GPT-5.6 Sol	Sudoku	6	78	76	100	99	100	95	96
GPT-5.6 Sol	Sudoku	9	44	4	100	16	100	23	38
GPT-5.6 Sol	Sudoku	12	31	5	100	16	100	16	30
GPT-5.6 Sol	DM	24	92	92	100	100	100	98	99
GPT-5.6 Sol	DM	48	37	32	100	56	100	45	59
GPT-5.6 Sol	DM	72	22	10	100	43	100	24	42
GPT-5.6 Sol	DM	96	20	5	100	39	100	21	39
GPT-5.6 Terra	CrossChoice	7	19	18	98	81	98	58	72
GPT-5.6 Terra	CrossChoice	20	1	1	100	25	98	18	35
GPT-5.6 Terra	CrossChoice	33	0	0	100	14	100	19	38
GPT-5.6 Terra	CrossChoice	46	0	0	100	20	100	17	34
GPT-5.6 Terra	Sudoku	4	92	92	100	96	99	92	96
GPT-5.6 Terra	Sudoku	6	47	44	100	81	99	74	82
GPT-5.6 Terra	Sudoku	9	14	4	100	10	100	13	27
GPT-5.6 Terra	Sudoku	12	29	4	100	19	100	18	34
GPT-5.6 Terra	DM	24	67	67	100	95	98	88	92
GPT-5.6 Terra	DM	48	0	0	100	34	99	28	44
GPT-5.6 Terra	DM	72	0	0	100	25	100	15	32
GPT-5.6 Terra	DM	96	0	0	100	25	100	20	37
GPT-5.6 Luna	CrossChoice	7	21	20	96	55	97	41	59
GPT-5.6 Luna	CrossChoice	20	0	0	99	17	99	17	36
GPT-5.6 Luna	CrossChoice	33	0	0	100	18	100	17	35
GPT-5.6 Luna	CrossChoice	46	0	0	100	25	100	17	33
GPT-5.6 Luna	Sudoku	4	88	87	100	97	98	95	97
GPT-5.6 Luna	Sudoku	6	44	39	100	65	100	64	75
GPT-5.6 Luna	Sudoku	9	6	1	100	13	100	20	38
GPT-5.6 Luna	Sudoku	12	4	1	100	15	100	17	33
GPT-5.6 Luna	DM	24	11	11	98	61	98	55	70
GPT-5.6 Luna	DM	48	0	0	99	12	100	19	36
GPT-5.6 Luna	DM	72	0	0	100	25	100	16	31
GPT-5.6 Luna	DM	96	0	0	100	17	100	14	32
Gemini 3.7 Flash	CrossChoice	7	34	34	100	56	100	60	72
Gemini 3.7 Flash	CrossChoice	20	6	0	100	17	100	16	33
Gemini 3.7 Flash	CrossChoice	33	2	0	100	12	100	15	34
Gemini 3.7 Flash	CrossChoice	46	2	1	100	16	100	16	35
Gemini 3.7 Flash	Sudoku	4	96	96	100	99	100	98	99

Model	Family	N	Free-must legal	Free-must exact	Forced legal	Forced exact	Pair legal	Pair exact	Pair sibling acc.
Gemini 3.7 Flash	Sudoku	6	4	4	100	66	99	53	70
Gemini 3.7 Flash	Sudoku	9	13	4	100	20	100	16	34
Gemini 3.7 Flash	Sudoku	12	4	2	100	17	100	17	34
Gemini 3.7 Flash	DM	24	13	9	100	49	100	39	60
Gemini 3.7 Flash	DM	48	4	2	100	29	100	14	32
Gemini 3.7 Flash	DM	72	7	1	100	24	100	16	32
Gemini 3.7 Flash	DM	96	4	0	100	21	100	17	33
Qwen3.8-27B	CrossChoice	7	0	0	83	23	87	24	45
Qwen3.8-27B	CrossChoice	20	0	0	95	15	71	11	31
Qwen3.8-27B	CrossChoice	33	0	0	100	14	97	17	36
Qwen3.8-27B	CrossChoice	46	0	0	98	17	99	20	36
Qwen3.8-27B	Sudoku	4	55	51	99	83	99	88	93
Qwen3.8-27B	Sudoku	6	15	10	99	42	97	37	54
Qwen3.8-27B	Sudoku	9	4	1	99	18	98	14	32
Qwen3.8-27B	Sudoku	12	0	0	85	13	91	15	32
Qwen3.8-27B	DM	24	0	0	69	14	66	13	38
Qwen3.8-27B	DM	48	0	0	89	15	80	13	35
Qwen3.8-27B	DM	72	0	0	84	14	77	12	32
Qwen3.8-27B	DM	96	0	0	94	18	81	15	34
Gemma4-26B	CrossChoice	7	0	0	98	17	100	22	38
Gemma4-26B	CrossChoice	20	0	0	100	17	100	19	35
Gemma4-26B	CrossChoice	33	0	0	98	15	100	17	33
Gemma4-26B	CrossChoice	46	0	0	98	22	100	18	34
Gemma4-26B	Sudoku	4	33	25	95	71	100	72	85
Gemma4-26B	Sudoku	6	8	6	96	38	100	31	52
Gemma4-26B	Sudoku	9	0	0	97	18	100	17	34
Gemma4-26B	Sudoku	12	0	0	97	14	100	15	32
Gemma4-26B	DM	24	0	0	100	19	100	19	38
Gemma4-26B	DM	48	0	0	97	15	100	17	34
Gemma4-26B	DM	72	0	0	97	13	100	17	34
Gemma4-26B	DM	96	0	0	99	21	100	18	34
GPT-OSS-20B	CrossChoice	7	1	0	100	21	100	18	33
GPT-OSS-20B	CrossChoice	20	4	1	100	18	100	14	32
GPT-OSS-20B	CrossChoice	33	1	0	100	19	94	17	31
GPT-OSS-20B	CrossChoice	46	6	1	100	15	99	16	32
GPT-OSS-20B	Sudoku	4	22	18	89	46	93	43	62
GPT-OSS-20B	Sudoku	6	3	0	98	21	95	21	39
GPT-OSS-20B	Sudoku	9	4	3	96	16	97	19	36
GPT-OSS-20B	Sudoku	12	3	0	100	18	100	17	32
GPT-OSS-20B	DM	24	0	0	100	13	98	17	34
GPT-OSS-20B	DM	48	0	0	96	13	90	14	33
GPT-OSS-20B	DM	72	0	0	92	24	89	13	34
GPT-OSS-20B	DM	96	0	0	87	15	93	16	34

Table 17: Relabel and recode controls (first 18 triads per cell), in % (DM = Three-zone DM). Relabel exact: all three pivots mapped correctly in each of the six A/B/C orders; identical: the decoded mapping is the same in all six orders; agreement: a sibling receives the same state in any two orders. Recode acc.: accuracy when the states are renamed by a temporary codebook; retention: siblings correct under the ordinary names that stay correct under the codebook (a re-ask of one sibling alone). Survival: share of raw-correct siblings that stay correct in the joint forced view alone, in the two pairwise views alone, and in all four views (all 100 triads).

Model	Family	N	Relabel exact	Relabel identical	Relabel agreement	Recode acc.	Recode retention	Survival joint	Survival pair	Survival 4 views
GPT-5.6 Sol	CrossChoice	7	99	94	99	90	92	100	95	91
GPT-5.6 Sol	CrossChoice	20	52	0	49	55	73	57	70	32
GPT-5.6 Sol	CrossChoice	33	26	6	46	39	73	41	53	14
GPT-5.6 Sol	CrossChoice	46	15	0	41	34	81	35	54	12
GPT-5.6 Sol	Sudoku	4	100	100	100	98	98	100	99	99
GPT-5.6 Sol	Sudoku	6	98	89	98	99	99	99	96	93
GPT-5.6 Sol	Sudoku	9	16	0	40	32	71	37	45	7
GPT-5.6 Sol	Sudoku	12	10	0	44	35	69	28	15	2
GPT-5.6 Sol	DM	24	100	100	100	97	98	100	99	98
GPT-5.6 Sol	DM	48	69	11	62	56	68	65	73	33
GPT-5.6 Sol	DM	72	35	0	48	35	100	47	63	25
GPT-5.6 Sol	DM	96	39	6	50	33	100	45	63	17
GPT-5.6 Terra	CrossChoice	7	81	28	74	59	96	91	80	59
GPT-5.6 Terra	CrossChoice	20	32	0	37	26	33	44	52	9
GPT-5.6 Terra	CrossChoice	33	16	0	32	33	94	26	53	8
GPT-5.6 Terra	CrossChoice	46	18	0	37	37	97	33	50	11
GPT-5.6 Terra	Sudoku	4	99	94	99	97	98	97	96	91
GPT-5.6 Terra	Sudoku	6	77	28	73	92	90	86	86	65
GPT-5.6 Terra	Sudoku	9	12	0	34	31	67	31	15	3
GPT-5.6 Terra	Sudoku	12	13	0	42	31	56	31	27	1

Model	Family	N	Relabel exact	Relabel identical	Relabel agreement	Recode acc.	Recode retention	Survival joint	Survival pair	Survival 4 views
GPT-5.6 Terra	DM	24	97	83	96	82	83	96	93	84
GPT-5.6 Terra	DM	48	37	0	46	32	97	45	58	13
GPT-5.6 Terra	DM	72	25	0	40	34	100	37	46	8
GPT-5.6 Terra	DM	96	23	0	50	33	100	41	52	8
GPT-5.6 Luna	CrossChoice	7	56	0	53	53	70	73	68	32
GPT-5.6 Luna	CrossChoice	20	18	0	34	31	46	38	31	7
GPT-5.6 Luna	CrossChoice	33	18	0	34	33	94	33	47	10
GPT-5.6 Luna	CrossChoice	46	19	0	33	35	100	42	45	6
GPT-5.6 Luna	Sudoku	4	95	78	94	95	95	98	98	93
GPT-5.6 Luna	Sudoku	6	71	17	66	93	95	77	77	48
GPT-5.6 Luna	Sudoku	9	19	0	33	28	52	36	37	5
GPT-5.6 Luna	Sudoku	12	21	0	35	23	33	31	29	2
GPT-5.6 Luna	DM	24	57	6	63	69	82	75	73	42
GPT-5.6 Luna	DM	48	17	0	30	34	100	29	56	14
GPT-5.6 Luna	DM	72	23	0	35	33	100	39	44	6
GPT-5.6 Luna	DM	96	20	0	35	33	100	36	50	8
Gemini 3.7 Flash	CrossChoice	7	63	11	57	47	63	65	75	37
Gemini 3.7 Flash	CrossChoice	20	27	0	40	34	44	37	30	6
Gemini 3.7 Flash	CrossChoice	33	10	0	33	32	46	25	38	4
Gemini 3.7 Flash	CrossChoice	46	17	0	32	37	46	28	42	3
Gemini 3.7 Flash	Sudoku	4	95	78	94	96	96	99	99	97
Gemini 3.7 Flash	Sudoku	6	71	17	68	50	74	74	81	51
Gemini 3.7 Flash	Sudoku	9	20	0	37	33	100	45	51	13
Gemini 3.7 Flash	Sudoku	12	11	0	43	33	100	34	52	6
Gemini 3.7 Flash	DM	24	49	0	51	37	44	61	61	26
Gemini 3.7 Flash	DM	48	38	0	45	33	67	41	35	8
Gemini 3.7 Flash	DM	72	28	0	42	32	83	33	27	1
Gemini 3.7 Flash	DM	96	31	0	48	34	100	31	21	0
Qwen3.8-27B	CrossChoice	7	19	0	34	37	43	35	38	6
Qwen3.8-27B	CrossChoice	20	16	0	26	33	29	35	36	2
Qwen3.8-27B	CrossChoice	33	15	0	24	41	67	32	43	9
Qwen3.8-27B	CrossChoice	46	19	0	23	30	89	33	39	5
Qwen3.8-27B	Sudoku	4	78	17	74	100	100	89	93	77
Qwen3.8-27B	Sudoku	6	46	0	40	66	75	59	62	20
Qwen3.8-27B	Sudoku	9	14	0	24	36	62	34	43	3
Qwen3.8-27B	Sudoku	12	10	0	27	33	60	38	51	9
Qwen3.8-27B	DM	24	18	6	38	44	52	42	39	9
Qwen3.8-27B	DM	48	16	0	24	33	97	39	60	13
Qwen3.8-27B	DM	72	12	0	26	34	74	39	54	10
Qwen3.8-27B	DM	96	12	0	27	31	67	32	50	8
Gemma4-26B	CrossChoice	7	22	0	27	51	36	30	37	4
Gemma4-26B	CrossChoice	20	19	0	32	33	40	34	46	2
Gemma4-26B	CrossChoice	33	16	0	33	32	47	26	39	0
Gemma4-26B	CrossChoice	46	14	0	31	31	42	37	42	1
Gemma4-26B	Sudoku	4	67	17	62	99	99	80	85	59
Gemma4-26B	Sudoku	6	31	0	37	55	76	60	67	31
Gemma4-26B	Sudoku	9	13	0	28	31	47	37	48	3
Gemma4-26B	Sudoku	12	16	0	29	35	58	30	47	2
Gemma4-26B	DM	24	15	0	31	37	86	34	57	10
Gemma4-26B	DM	48	18	0	26	28	67	34	48	4
Gemma4-26B	DM	72	21	0	28	38	92	25	50	3
Gemma4-26B	DM	96	19	0	27	34	79	35	51	4
GPT-OSS-20B	CrossChoice	7	19	0	25	33	100	38	47	1
GPT-OSS-20B	CrossChoice	20	14	0	27	33	100	35	46	0
GPT-OSS-20B	CrossChoice	33	14	0	29	35	97	37	47	6
GPT-OSS-20B	CrossChoice	46	17	0	26	39	86	29	43	4
GPT-OSS-20B	Sudoku	4	43	0	47	85	87	63	63	27
GPT-OSS-20B	Sudoku	6	21	0	33	48	67	35	45	4
GPT-OSS-20B	Sudoku	9	16	0	24	40	47	41	41	3
GPT-OSS-20B	Sudoku	12	17	0	23	35	75	28	44	0
GPT-OSS-20B	DM	24	23	0	32	33	100	29	47	5
GPT-OSS-20B	DM	48	17	0	29	33	97	34	45	6
GPT-OSS-20B	DM	72	16	0	31	31	75	40	51	11
GPT-OSS-20B	DM	96	15	0	34	33	100	30	45	4

G FAILURE ANALYSIS

The favored state changes with model and family. On Three-zone DM six models fall to MULTIPLE and Gemma4 to UNIQUE; on CrossChoice Sol, Terra, Gemma4 and, at the two largest sizes, Luna fall to UNIQUE, while GPT-OSS falls to MULTIPLE in every triad and Qwen in three of its four post-frontier cells. Terra, Luna and Sol give one state to all three siblings in 94% to 100% of triads on Three-zone DM N=72 and 96. Gemini on CrossChoice and Qwen on CrossChoice N=33 are the exceptions closest to uniform.

Table 18: Where wrong answers go, over must-answer responses that returned a state. Entropy of the answer distribution, normalized to [0, 1]; all same: share of triads answered with one state for all three siblings; differentiated: triads with three different answers; rigidity is $2 \max_d M[s, d] - 1$ averaged over true states with errors, where $M[s, d]$ is the share of the errors on true state s that go to state d; the attractor is the wrong state that absorbs the largest share of errors, with its strength in %. V: Cramér's V between the true state and the returned state over the 300 separate must-answer siblings (0 = the returned state carries no information about the true state, 1 = determined by it). DM = Three-zone DM.

Model	Family	N	Entropy	All same	Differ-entiated	Rigidity	Attractor	Strength	V
GPT-5.6 Sol	CrossChoice	7	0.99	1	84	0.88	UNIQUE	91	0.83
GPT-5.6 Sol	CrossChoice	20	0.65	11	45	0.79	UNIQUE	84	0.39
GPT-5.6 Sol	CrossChoice	33	0.50	51	24	0.81	UNIQUE	86	0.06
GPT-5.6 Sol	CrossChoice	46	0.20	83	8	0.93	UNIQUE	95	0.09
GPT-5.6 Sol	Sudoku	4	1.00	0	99	1.00	UNIQUE	100	0.99
GPT-5.6 Sol	Sudoku	6	1.00	0	94	1.00	MULTIPLE	100	0.95
GPT-5.6 Sol	Sudoku	9	0.39	64	18	0.90	MULTIPLE	92	0.19
GPT-5.6 Sol	Sudoku	12	0.17	86	7	0.95	MULTIPLE	97	0.06
GPT-5.6 Sol	DM	24	1.00	0	95	1.00	NONE	50	0.95
GPT-5.6 Sol	DM	48	0.59	50	28	0.93	MULTIPLE	95	0.38
GPT-5.6 Sol	DM	72	0.09	94	3	0.97	MULTIPLE	98	0.07
GPT-5.6 Sol	DM	96	0.02	99	0	1.00	MULTIPLE	100	0.06
GPT-5.6 Terra	CrossChoice	7	0.82	16	50	0.69	UNIQUE	78	0.48
GPT-5.6 Terra	CrossChoice	20	0.63	34	33	0.71	UNIQUE	78	0.09
GPT-5.6 Terra	CrossChoice	33	0.19	86	7	0.86	UNIQUE	98	0.06
GPT-5.6 Terra	CrossChoice	46	0.07	96	2	0.66	UNIQUE	100	0.06
GPT-5.6 Terra	Sudoku	4	1.00	0	92	0.63	UNIQUE	81	0.90
GPT-5.6 Terra	Sudoku	6	0.97	5	73	0.83	MULTIPLE	88	0.71
GPT-5.6 Terra	Sudoku	9	0.36	65	18	0.91	MULTIPLE	94	0.08
GPT-5.6 Terra	Sudoku	12	0.63	35	32	0.68	UNIQUE	76	0.03
GPT-5.6 Terra	DM	24	1.00	2	86	0.54	NONE	52	0.85
GPT-5.6 Terra	DM	48	0.17	89	6	0.75	MULTIPLE	98	0.10
GPT-5.6 Terra	DM	72	-0.00	100	0	1.00	MULTIPLE	100	0.00
GPT-5.6 Terra	DM	96	-0.00	100	0	1.00	MULTIPLE	100	0.00
GPT-5.6 Luna	CrossChoice	7	0.77	25	42	0.65	MULTIPLE	74	0.31
GPT-5.6 Luna	CrossChoice	20	0.60	40	30	0.72	MULTIPLE	79	0.10
GPT-5.6 Luna	CrossChoice	33	0.20	85	8	0.96	UNIQUE	97	0.01
GPT-5.6 Luna	CrossChoice	46	0.08	95	2	0.98	UNIQUE	98	0.07
GPT-5.6 Luna	Sudoku	4	1.00	0	95	1.00	UNIQUE	100	0.95
GPT-5.6 Luna	Sudoku	6	0.98	3	78	0.84	MULTIPLE	88	0.77
GPT-5.6 Luna	Sudoku	9	0.55	47	26	0.77	MULTIPLE	85	0.10
GPT-5.6 Luna	Sudoku	12	0.63	28	36	0.71	MULTIPLE	78	0.02
GPT-5.6 Luna	DM	24	0.91	11	58	0.31	MULTIPLE	73	0.59
GPT-5.6 Luna	DM	48	0.11	92	4	0.97	MULTIPLE	98	0.08
GPT-5.6 Luna	DM	72	0.02	99	0	0.99	MULTIPLE	100	0.06
GPT-5.6 Luna	DM	96	-0.00	100	0	1.00	MULTIPLE	100	0.00
Gemini 3.7 Flash	CrossChoice	7	0.83	18	48	0.49	MULTIPLE	78	0.43
Gemini 3.7 Flash	CrossChoice	20	0.96	22	46	0.21	NONE	63	0.08
Gemini 3.7 Flash	CrossChoice	33	0.85	15	48	0.50	NONE	70	0.08
Gemini 3.7 Flash	CrossChoice	46	0.79	34	39	0.65	UNIQUE	83	0.12
Gemini 3.7 Flash	Sudoku	4	1.00	0	98	1.00	MULTIPLE	100	0.99
Gemini 3.7 Flash	Sudoku	6	0.73	43	34	0.81	MULTIPLE	86	0.36
Gemini 3.7 Flash	Sudoku	9	0.11	93	4	1.00	UNIQUE	100	0.10
Gemini 3.7 Flash	Sudoku	12	-0.00	100	0	1.00	UNIQUE	100	0.00
Gemini 3.7 Flash	DM	24	0.95	14	60	0.83	UNIQUE	88	0.55
Gemini 3.7 Flash	DM	48	0.39	68	16	0.91	MULTIPLE	93	0.12
Gemini 3.7 Flash	DM	72	0.10	94	3	0.98	MULTIPLE	99	0.09
Gemini 3.7 Flash	DM	96	0.04	98	1	0.98	MULTIPLE	99	0.08
Qwen3.8-27B	CrossChoice	7	0.65	34	33	0.67	MULTIPLE	76	0.08
Qwen3.8-27B	CrossChoice	20	0.57	39	30	0.78	UNIQUE	84	0.02
Qwen3.8-27B	CrossChoice	33	0.78	30	38	0.57	MULTIPLE	75	0.10
Qwen3.8-27B	CrossChoice	46	0.53	52	25	0.64	MULTIPLE	87	0.10
Qwen3.8-27B	Sudoku	4	1.00	0	98	1.00	UNIQUE	100	0.98
Qwen3.8-27B	Sudoku	6	0.88	15	54	0.80	UNIQUE	89	0.53
Qwen3.8-27B	Sudoku	9	0.65	25	38	0.69	MULTIPLE	77	0.08
Qwen3.8-27B	Sudoku	12	0.44	58	21	0.89	MULTIPLE	92	0.04
Qwen3.8-27B	DM	24	0.93	16	51	0.44	MULTIPLE	70	0.32
Qwen3.8-27B	DM	48	0.30	79	11	0.95	MULTIPLE	96	0.06

Model	Family	N	Entropy	All same	Differ-entiated	Rigidity	Attractor	Strength	V
Qwen3.8-27B	DM	72	0.40	73	14	0.91	MULTIPLE	94	0.04
Qwen3.8-27B	DM	96	0.54	64	18	0.80	MULTIPLE	85	0.04
Gemma4-26B	CrossChoice	7	0.56	51	26	0.78	UNIQUE	97	0.23
Gemma4-26B	CrossChoice	20	0.56	54	23	0.76	UNIQUE	84	0.07
Gemma4-26B	CrossChoice	33	0.68	34	33	0.70	UNIQUE	78	0.12
Gemma4-26B	CrossChoice	46	0.58	39	30	0.79	UNIQUE	84	0.06
Gemma4-26B	Sudoku	4	1.00	0	97	1.00	MULTIPLE	100	0.97
Gemma4-26B	Sudoku	6	0.57	54	28	1.00	UNIQUE	100	0.41
Gemma4-26B	Sudoku	9	0.16	87	7	0.97	UNIQUE	98	0.02
Gemma4-26B	Sudoku	12	0.23	83	9	0.94	UNIQUE	95	0.10
Gemma4-26B	DM	24	0.17	89	6	0.72	UNIQUE	99	0.07
Gemma4-26B	DM	48	0.29	81	9	0.85	UNIQUE	98	0.12
Gemma4-26B	DM	72	0.31	80	10	0.71	UNIQUE	95	0.09
Gemma4-26B	DM	96	0.34	78	12	0.61	UNIQUE	95	0.09
GPT-OSS-20B	CrossChoice	7	-0.00	100	0	1.00	MULTIPLE	100	0.00
GPT-OSS-20B	CrossChoice	20	-0.00	100	0	1.00	MULTIPLE	100	0.00
GPT-OSS-20B	CrossChoice	33	-0.00	100	0	1.00	MULTIPLE	100	0.00
GPT-OSS-20B	CrossChoice	46	-0.00	100	0	1.00	MULTIPLE	100	0.00
GPT-OSS-20B	Sudoku	4	0.95	3	76	0.75	UNIQUE	81	0.79
GPT-OSS-20B	Sudoku	6	0.83	26	42	0.61	MULTIPLE	77	0.28
GPT-OSS-20B	Sudoku	9	0.48	61	20	0.88	MULTIPLE	91	0.08
GPT-OSS-20B	Sudoku	12	0.43	61	20	0.91	MULTIPLE	94	0.07
GPT-OSS-20B	DM	24	0.17	89	6	0.98	MULTIPLE	98	0.12
GPT-OSS-20B	DM	48	0.38	81	10	0.92	MULTIPLE	94	0.07
GPT-OSS-20B	DM	72	0.34	76	12	0.87	MULTIPLE	93	0.07
GPT-OSS-20B	DM	96	0.30	81	10	0.87	MULTIPLE	95	0.04

H ABSTENTION

Abstention precision ranges from 33% to 100% across the 37 post-frontier cells that abstain, with a median of 68%. Models that never abstain are no different: Terra, Luna and Sol answer every sibling of Three-zone DM N=96 and CrossChoice N=46; 65.7% to 67.3% of those answers are wrong and 2.7% to 5.7% are verified.

Table 19: Abstention metrics per cell, in %. Precision and recall use the paired must-answer as the error criterion; verified recall counts a sibling as an error when it is not four-view verified.

Model	Family	N	Abstain	Precision	Recall	Selective acc.	Verified sel. acc.	Verified recall
GPT-5.6 Sol	CrossChoice	7	0.0	–	0.0	92.3	75.0	0.0
GPT-5.6 Sol	CrossChoice	20	0.0	–	0.0	53.3	14.0	0.0
GPT-5.6 Sol	CrossChoice	33	0.0	–	0.0	36.0	3.3	0.0
GPT-5.6 Sol	CrossChoice	46	0.0	–	0.0	34.0	3.3	0.0
GPT-5.6 Sol	Sudoku	4	0.0	–	0.0	99.3	97.7	0.0
GPT-5.6 Sol	Sudoku	6	0.0	–	0.0	95.7	88.0	0.0
GPT-5.6 Sol	Sudoku	9	0.7	100.0	1.1	39.3	2.7	0.7
GPT-5.6 Sol	Sudoku	12	4.7	78.6	5.5	35.3	0.7	4.7
GPT-5.6 Sol	Three-zone DM	24	0.0	–	0.0	96.0	90.7	0.0
GPT-5.6 Sol	Three-zone DM	48	0.0	–	0.0	55.7	13.0	0.0
GPT-5.6 Sol	Three-zone DM	72	0.0	–	0.0	35.0	8.0	0.0
GPT-5.6 Sol	Three-zone DM	96	0.0	–	0.0	34.0	5.7	0.0
GPT-5.6 Terra	CrossChoice	7	0.0	–	0.0	60.3	31.0	0.0
GPT-5.6 Terra	CrossChoice	20	0.3	100.0	0.5	36.8	1.7	0.3
GPT-5.6 Terra	CrossChoice	33	0.0	–	0.0	32.0	2.0	0.0
GPT-5.6 Terra	CrossChoice	46	0.0	–	0.0	34.3	3.7	0.0
GPT-5.6 Terra	Sudoku	4	0.0	–	0.0	94.0	80.7	0.0
GPT-5.6 Terra	Sudoku	6	0.0	–	0.0	82.7	44.0	0.0
GPT-5.6 Terra	Sudoku	9	19.3	62.1	18.8	36.0	0.8	19.5
GPT-5.6 Terra	Sudoku	12	63.3	62.6	60.7	41.8	0.0	63.2
GPT-5.6 Terra	Three-zone DM	24	0.0	–	0.0	84.3	64.7	0.0
GPT-5.6 Terra	Three-zone DM	48	0.0	–	0.0	35.7	4.3	0.0
GPT-5.6 Terra	Three-zone DM	72	0.0	–	0.0	33.3	2.7	0.0
GPT-5.6 Terra	Three-zone DM	96	0.0	–	0.0	33.3	2.7	0.0
GPT-5.6 Luna	CrossChoice	7	0.0	–	0.0	53.7	12.0	0.0
GPT-5.6 Luna	CrossChoice	20	0.0	–	0.0	35.0	1.0	0.0
GPT-5.6 Luna	CrossChoice	33	0.0	–	0.0	32.3	2.7	0.0
GPT-5.6 Luna	CrossChoice	46	0.0	–	0.0	34.3	2.0	0.0
GPT-5.6 Luna	Sudoku	4	0.0	–	0.0	95.3	86.0	0.0
GPT-5.6 Luna	Sudoku	6	0.0	–	0.0	81.7	35.0	0.0
GPT-5.6 Luna	Sudoku	9	4.0	83.3	5.3	37.5	1.4	4.1
GPT-5.6 Luna	Sudoku	12	8.7	61.5	8.0	31.4	0.7	8.7
GPT-5.6 Luna	Three-zone DM	24	0.0	–	0.0	70.3	23.3	0.0

Model	Family	N	Abstain	Precision	Recall	Selective acc.	Verified sel. acc.	Verified recall
GPT-5.6 Luna	Three-zone DM	48	0.0	–	0.0	33.7	4.3	0.0
GPT-5.6 Luna	Three-zone DM	72	0.0	–	0.0	33.3	2.0	0.0
GPT-5.6 Luna	Three-zone DM	96	0.0	–	0.0	32.7	2.7	0.0
Gemini 3.7 Flash	CrossChoice	7	0.7	50.0	0.8	73.5	19.5	0.9
Gemini 3.7 Flash	CrossChoice	20	22.0	65.2	22.3	35.5	1.3	22.4
Gemini 3.7 Flash	CrossChoice	33	28.7	67.4	29.6	35.0	0.0	28.0
Gemini 3.7 Flash	CrossChoice	46	36.7	66.4	38.4	39.5	1.1	36.7
Gemini 3.7 Flash	Sudoku	4	0.0	–	0.0	99.0	95.3	0.0
Gemini 3.7 Flash	Sudoku	6	1.0	0.0	0.0	55.6	24.6	0.9
Gemini 3.7 Flash	Sudoku	9	0.7	100.0	1.0	35.2	4.4	0.7
Gemini 3.7 Flash	Sudoku	12	6.0	72.2	6.5	33.7	2.1	6.1
Gemini 3.7 Flash	Three-zone DM	24	6.0	38.9	6.6	70.2	12.4	7.2
Gemini 3.7 Flash	Three-zone DM	48	12.3	56.8	10.7	37.3	2.3	12.7
Gemini 3.7 Flash	Three-zone DM	72	17.3	75.0	19.9	35.5	0.4	17.4
Gemini 3.7 Flash	Three-zone DM	96	25.3	61.8	23.5	33.0	0.0	25.3
Qwen3.8-27B	CrossChoice	7	14.3	60.5	13.5	40.9	1.2	14.6
Qwen3.8-27B	CrossChoice	20	52.0	67.9	52.5	30.6	0.7	52.3
Qwen3.8-27B	CrossChoice	33	57.0	66.1	57.1	36.4	1.6	57.0
Qwen3.8-27B	CrossChoice	46	33.3	66.0	34.2	34.0	1.0	32.9
Qwen3.8-27B	Sudoku	4	0.7	0.0	0.0	98.7	75.2	0.0
Qwen3.8-27B	Sudoku	6	13.7	36.6	13.5	65.3	11.2	14.5
Qwen3.8-27B	Sudoku	9	39.7	70.6	44.7	38.9	0.6	40.1
Qwen3.8-27B	Sudoku	12	49.0	73.3	53.2	32.0	2.0	48.8
Qwen3.8-27B	Three-zone DM	24	41.0	52.0	41.6	49.2	1.7	40.4
Qwen3.8-27B	Three-zone DM	48	36.3	72.5	39.5	33.5	3.7	36.9
Qwen3.8-27B	Three-zone DM	72	54.3	59.5	48.5	31.6	0.7	53.8
Qwen3.8-27B	Three-zone DM	96	49.3	70.3	51.2	37.6	2.7	49.7
Gemma4-26B	CrossChoice	7	0.3	0.0	0.0	38.3	1.3	0.3
Gemma4-26B	CrossChoice	20	0.7	100.0	1.0	32.6	0.3	0.7
Gemma4-26B	CrossChoice	33	9.3	71.4	10.0	34.2	0.0	9.3
Gemma4-26B	CrossChoice	46	5.0	53.3	4.0	35.1	0.4	5.0
Gemma4-26B	Sudoku	4	0.0	–	0.0	99.0	57.0	0.0
Gemma4-26B	Sudoku	6	1.3	25.0	0.7	45.6	13.2	1.2
Gemma4-26B	Sudoku	9	1.7	80.0	2.0	33.9	1.0	1.7
Gemma4-26B	Sudoku	12	4.0	75.0	4.4	33.7	0.7	4.0
Gemma4-26B	Three-zone DM	24	0.0	–	0.0	34.6	3.4	0.0
Gemma4-26B	Three-zone DM	48	0.7	100.0	1.0	33.9	1.3	0.7
Gemma4-26B	Three-zone DM	72	0.7	100.0	0.5	34.2	0.7	0.7
Gemma4-26B	Three-zone DM	96	1.3	75.0	1.6	32.9	1.4	1.4
GPT-OSS-20B	CrossChoice	7	1.0	33.3	0.5	33.0	0.3	1.0
GPT-OSS-20B	CrossChoice	20	5.3	62.5	5.0	33.1	0.0	5.3
GPT-OSS-20B	CrossChoice	33	4.0	83.3	5.0	34.0	2.1	4.1
GPT-OSS-20B	CrossChoice	46	2.3	42.9	1.5	32.8	1.4	2.4
GPT-OSS-20B	Sudoku	4	0.7	50.0	2.0	88.9	21.5	0.9
GPT-OSS-20B	Sudoku	6	19.0	42.1	15.9	50.2	2.1	19.0
GPT-OSS-20B	Sudoku	9	62.7	62.8	61.1	39.3	0.0	62.3
GPT-OSS-20B	Sudoku	12	94.0	66.0	92.5	22.2	0.0	94.0
GPT-OSS-20B	Three-zone DM	24	10.0	60.0	8.7	31.1	1.9	10.2
GPT-OSS-20B	Three-zone DM	48	19.7	69.5	20.4	33.6	2.1	19.7
GPT-OSS-20B	Three-zone DM	72	10.3	77.4	11.6	34.2	3.7	10.7
GPT-OSS-20B	Three-zone DM	96	19.7	67.8	19.8	32.0	1.7	19.9

I CLARIFICATION

Table 20: Clarification per cell, in %. Parsable: the formal query could be read; consistent: the natural-language question matches the formal query; resolving: the query splits the two target values under the solver. Verified MULTIPLE: triads whose MULTIPLE sibling passed all four views (count, then the resolving rate among them).

Model	Family	N	Parsable	Consistent	Resolving	Verified MULTIPLE	Resolving if verified
GPT-5.6 Sol	CrossChoice	7	100	100	97	95	97
GPT-5.6 Sol	CrossChoice	20	100	100	31	30	33
GPT-5.6 Sol	CrossChoice	33	98	98	8	0	–
GPT-5.6 Sol	CrossChoice	46	99	99	7	0	–
GPT-5.6 Sol	Sudoku	4	100	100	100	99	100
GPT-5.6 Sol	Sudoku	6	100	100	98	89	99
GPT-5.6 Sol	Sudoku	9	100	100	54	5	60
GPT-5.6 Sol	Sudoku	12	100	100	23	2	50
GPT-5.6 Sol	Three-zone DM	24	99	81	75	98	74
GPT-5.6 Sol	Three-zone DM	48	100	75	21	33	15
GPT-5.6 Sol	Three-zone DM	72	100	85	14	24	25
GPT-5.6 Sol	Three-zone DM	96	100	79	12	17	18

Model	Family	N	Parsable	Consistent	Resolving	Verified MULTIPLE	Resolving if verified
GPT-5.6 Terra	CrossChoice	7	100	100	83	76	83
GPT-5.6 Terra	CrossChoice	20	100	100	21	4	25
GPT-5.6 Terra	CrossChoice	33	100	100	6	0	–
GPT-5.6 Terra	CrossChoice	46	100	100	6	0	–
GPT-5.6 Terra	Sudoku	4	100	100	97	80	98
GPT-5.6 Terra	Sudoku	6	100	100	76	59	85
GPT-5.6 Terra	Sudoku	9	100	100	31	1	100
GPT-5.6 Terra	Sudoku	12	100	100	16	0	–
GPT-5.6 Terra	Three-zone DM	24	100	99	58	78	60
GPT-5.6 Terra	Three-zone DM	48	100	100	13	13	31
GPT-5.6 Terra	Three-zone DM	72	100	100	11	8	0
GPT-5.6 Terra	Three-zone DM	96	100	99	8	8	25
GPT-5.6 Luna	CrossChoice	7	100	100	55	37	46
GPT-5.6 Luna	CrossChoice	20	100	100	21	3	33
GPT-5.6 Luna	CrossChoice	33	100	100	7	0	–
GPT-5.6 Luna	CrossChoice	46	100	100	5	0	–
GPT-5.6 Luna	Sudoku	4	100	100	95	91	96
GPT-5.6 Luna	Sudoku	6	100	100	74	42	79
GPT-5.6 Luna	Sudoku	9	100	100	19	4	25
GPT-5.6 Luna	Sudoku	12	100	100	17	1	100
GPT-5.6 Luna	Three-zone DM	24	100	98	19	43	16
GPT-5.6 Luna	Three-zone DM	48	100	100	6	13	15
GPT-5.6 Luna	Three-zone DM	72	99	97	5	6	33
GPT-5.6 Luna	Three-zone DM	96	100	99	0	8	0
Gemini 3.7 Flash	CrossChoice	7	100	100	35	30	23
Gemini 3.7 Flash	CrossChoice	20	100	100	6	1	0
Gemini 3.7 Flash	CrossChoice	33	100	100	4	0	–
Gemini 3.7 Flash	CrossChoice	46	100	100	4	0	–
Gemini 3.7 Flash	Sudoku	4	100	100	87	96	86
Gemini 3.7 Flash	Sudoku	6	100	100	51	47	57
Gemini 3.7 Flash	Sudoku	9	100	100	20	0	–
Gemini 3.7 Flash	Sudoku	12	100	100	18	0	–
Gemini 3.7 Flash	Three-zone DM	24	100	79	0	11	0
Gemini 3.7 Flash	Three-zone DM	48	100	56	8	6	0
Gemini 3.7 Flash	Three-zone DM	72	100	60	7	1	0
Gemini 3.7 Flash	Three-zone DM	96	100	60	0	0	–
Qwen3.8-27B	CrossChoice	7	3	3	1	4	0
Qwen3.8-27B	CrossChoice	20	28	28	3	2	0
Qwen3.8-27B	CrossChoice	33	48	48	0	3	0
Qwen3.8-27B	CrossChoice	46	37	37	1	5	0
Qwen3.8-27B	Sudoku	4	97	97	95	73	93
Qwen3.8-27B	Sudoku	6	26	26	26	5	60
Qwen3.8-27B	Sudoku	9	8	8	5	1	0
Qwen3.8-27B	Sudoku	12	3	3	1	8	0
Qwen3.8-27B	Three-zone DM	24	75	43	2	9	0
Qwen3.8-27B	Three-zone DM	48	64	33	2	13	0
Qwen3.8-27B	Three-zone DM	72	82	54	0	8	0
Qwen3.8-27B	Three-zone DM	96	77	47	0	6	0
Gemma4-26B	CrossChoice	7	0	0	0	0	–
Gemma4-26B	CrossChoice	20	1	1	0	1	0
Gemma4-26B	CrossChoice	33	0	0	0	0	–
Gemma4-26B	CrossChoice	46	0	0	0	0	–
Gemma4-26B	Sudoku	4	47	47	31	46	26
Gemma4-26B	Sudoku	6	16	16	12	3	67
Gemma4-26B	Sudoku	9	29	28	4	0	–
Gemma4-26B	Sudoku	12	31	31	4	0	–
Gemma4-26B	Three-zone DM	24	13	3	0	0	–
Gemma4-26B	Three-zone DM	48	3	2	0	0	–
Gemma4-26B	Three-zone DM	72	0	0	0	0	–
Gemma4-26B	Three-zone DM	96	1	0	0	0	–
GPT-OSS-20B	CrossChoice	7	95	95	23	1	0
GPT-OSS-20B	CrossChoice	20	28	28	1	0	–
GPT-OSS-20B	CrossChoice	33	71	71	4	6	0
GPT-OSS-20B	CrossChoice	46	91	91	1	4	0
GPT-OSS-20B	Sudoku	4	98	98	18	18	17
GPT-OSS-20B	Sudoku	6	92	92	6	1	0
GPT-OSS-20B	Sudoku	9	99	99	1	2	0
GPT-OSS-20B	Sudoku	12	90	90	2	0	–
GPT-OSS-20B	Three-zone DM	24	45	10	0	5	0
GPT-OSS-20B	Three-zone DM	48	47	18	0	6	0
GPT-OSS-20B	Three-zone DM	72	48	25	0	10	0
GPT-OSS-20B	Three-zone DM	96	43	14	0	4	0

J REASONING TRAJECTORIES

Each separate must-answer trace is replayed to the model cut at 0, 10, 25, 50, 75 and 100% of its length, and at each cut the model is asked for the state. On Sudoku $N{=}4$ most of the rise in the probability on the correct state comes in the last quarter of the trace for Qwen and GPT-OSS. GPT-OSS on CrossChoice ends at divergence 0.000 to 0.001 at every size, the internal counterpart of its 1,200 identical MULTIPLE answers.

Table 21: State-probability readouts along the model's own reasoning, replayed and cut at the given fraction of its length. Sibling divergence is 0 when the three siblings receive identical distributions; the probability on the correct state is $1/3$ under identical distributions.

Cell	Model	Sibling divergence (JSD) at 0 → 0.5 → 0.75 → 1	Prob. on correct state at 0 → 1
Sudoku 4	Qwen3.8-27B	0.002 → 0.079 → 0.247 → 0.923	0.330 → 0.972
Sudoku 4	Gemma4-26B	0.005 → 0.524 → 0.821 → 0.937	0.329 → 0.964
Sudoku 4	GPT-OSS-20B	0.010 → 0.060 → 0.081 → 0.752	0.332 → 0.831
Sudoku 12	Qwen3.8-27B	0.001 → 0.009 → 0.008 → 0.020	0.333 → 0.342
Sudoku 12	Gemma4-26B	0.015 → 0.105 → 0.081 → 0.116	0.330 → 0.308
Sudoku 12	GPT-OSS-20B	0.022 → 0.112 → 0.123 → 0.094	0.330 → 0.330
Three-zone DM 96	Qwen3.8-27B	0.000 → 0.025 → 0.027 → 0.034	0.334 → 0.348
Three-zone DM 96	Gemma4-26B	0.008 → 0.046 → 0.050 → 0.060	0.336 → 0.332
Three-zone DM 96	GPT-OSS-20B	0.006 → 0.041 → 0.047 → 0.068	0.334 → 0.326
CrossChoice 46	Qwen3.8-27B	0.000 → 0.016 → 0.033 → 0.090	0.334 → 0.340
CrossChoice 46	Gemma4-26B	0.006 → 0.073 → 0.093 → 0.093	0.334 → 0.337
CrossChoice 46	GPT-OSS-20B	0.083 → 0.011 → 0.013 → 0.001	0.328 → 0.333

Table 22: Sibling divergence (JSD) and probability on the correct state at the six replay cuts (0, 10, 25, 50, 75, 100% of the model's own reasoning) for all 36 open-weight cells (stratum: all 100 triads).

Family	N	Model	JSD at 0	0.1	0.25	0.5	0.75	1	P(correct) at 0	at 1
CrossChoice	7	Qwen3.8-27B	0.000	0.006	0.004	0.011	0.025	0.068	0.333	0.379
CrossChoice	7	Gemma4-26B	0.007	0.066	0.081	0.125	0.158	0.229	0.329	0.400
CrossChoice	7	GPT-OSS-20B	0.005	0.003	0.005	0.007	0.037	0.001	0.334	0.333
CrossChoice	20	Qwen3.8-27B	0.000	0.002	0.004	0.004	0.004	0.005	0.333	0.334
CrossChoice	20	Gemma4-26B	0.011	0.083	0.067	0.072	0.075	0.103	0.336	0.350
CrossChoice	20	GPT-OSS-20B	0.012	0.005	0.005	0.006	0.026	0.001	0.328	0.333
CrossChoice	33	Qwen3.8-27B	0.000	0.002	0.003	0.015	0.033	0.068	0.333	0.325
CrossChoice	33	Gemma4-26B	0.008	0.078	0.084	0.083	0.094	0.092	0.331	0.349
CrossChoice	33	GPT-OSS-20B	0.003	0.002	0.002	0.003	0.011	0.000	0.333	0.333
CrossChoice	46	Qwen3.8-27B	0.000	0.001	0.001	0.016	0.033	0.090	0.334	0.340
CrossChoice	46	Gemma4-26B	0.006	0.074	0.086	0.073	0.093	0.093	0.334	0.337
CrossChoice	46	GPT-OSS-20B	0.083	0.044	0.020	0.011	0.013	0.001	0.328	0.333
Sudoku	4	Qwen3.8-27B	0.002	0.007	0.025	0.079	0.247	0.923	0.330	0.972
Sudoku	4	Gemma4-26B	0.005	0.072	0.293	0.524	0.821	0.937	0.329	0.964
Sudoku	4	GPT-OSS-20B	0.010	0.046	0.049	0.060	0.081	0.752	0.332	0.831
Sudoku	6	Qwen3.8-27B	0.001	0.005	0.042	0.046	0.094	0.381	0.334	0.619
Sudoku	6	Gemma4-26B	0.013	0.019	0.101	0.130	0.200	0.255	0.336	0.490
Sudoku	6	GPT-OSS-20B	0.007	0.025	0.026	0.036	0.037	0.413	0.336	0.489
Sudoku	9	Qwen3.8-27B	0.002	0.041	0.029	0.024	0.026	0.054	0.333	0.361
Sudoku	9	Gemma4-26B	0.009	0.013	0.124	0.070	0.052	0.067	0.339	0.334
Sudoku	9	GPT-OSS-20B	0.015	0.075	0.076	0.131	0.117	0.102	0.334	0.355
Sudoku	12	Qwen3.8-27B	0.001	0.003	0.011	0.009	0.008	0.020	0.333	0.342
Sudoku	12	Gemma4-26B	0.015	0.012	0.125	0.105	0.081	0.116	0.330	0.308
Sudoku	12	GPT-OSS-20B	0.022	0.083	0.072	0.112	0.123	0.094	0.330	0.330
Three-zone DM	24	Qwen3.8-27B	0.001	0.007	0.008	0.038	0.072	0.256	0.334	0.469
Three-zone DM	24	Gemma4-26B	0.007	0.009	0.044	0.038	0.048	0.073	0.329	0.346
Three-zone DM	24	GPT-OSS-20B	0.011	0.026	0.032	0.031	0.040	0.034	0.332	0.322
Three-zone DM	48	Qwen3.8-27B	0.001	0.012	0.013	0.016	0.026	0.071	0.333	0.333
Three-zone DM	48	Gemma4-26B	0.011	0.027	0.066	0.048	0.051	0.052	0.332	0.323
Three-zone DM	48	GPT-OSS-20B	0.006	0.042	0.047	0.048	0.054	0.082	0.327	0.330
Three-zone DM	72	Qwen3.8-27B	0.001	0.015	0.017	0.017	0.020	0.029	0.333	0.325
Three-zone DM	72	Gemma4-26B	0.008	0.033	0.061	0.036	0.048	0.047	0.331	0.330
Three-zone DM	72	GPT-OSS-20B	0.009	0.030	0.028	0.029	0.032	0.060	0.335	0.317
Three-zone DM	96	Qwen3.8-27B	0.000	0.018	0.020	0.025	0.027	0.034	0.334	0.348
Three-zone DM	96	Gemma4-26B	0.008	0.047	0.065	0.046	0.050	0.060	0.336	0.332
Three-zone DM	96	GPT-OSS-20B	0.006	0.031	0.031	0.041	0.047	0.068	0.334	0.326

Table 23: End-of-reasoning readouts by outcome stratum: triad count, then sibling divergence (JSD) / probability on the correct state.

Family	N	Model	All three correct	Completed, not all correct	Budget/output capped
CrossChoice	7	Qwen3.8-27B	–	–	100: 0.07 / 0.38
CrossChoice	7	Gemma4-26B	–	–	100: 0.23 / 0.40
CrossChoice	7	GPT-OSS-20B	–	100: 0.00 / 0.33	–
CrossChoice	20	Qwen3.8-27B	–	–	100: 0.00 / 0.33
CrossChoice	20	Gemma4-26B	–	–	100: 0.10 / 0.35
CrossChoice	20	GPT-OSS-20B	–	100: 0.00 / 0.33	–
CrossChoice	33	Qwen3.8-27B	–	–	100: 0.07 / 0.32
CrossChoice	33	Gemma4-26B	–	–	100: 0.09 / 0.35
CrossChoice	33	GPT-OSS-20B	–	100: 0.00 / 0.33	–
CrossChoice	46	Qwen3.8-27B	–	–	100: 0.09 / 0.34
CrossChoice	46	Gemma4-26B	–	–	100: 0.09 / 0.34
CrossChoice	46	GPT-OSS-20B	–	100: 0.00 / 0.33	–
Sudoku	4	Qwen3.8-27B	89: 0.95 / 0.99	4: 0.55 / 0.66	7: 0.82 / 0.93
Sudoku	4	Gemma4-26B	–	–	100: 0.94 / 0.96
Sudoku	4	GPT-OSS-20B	55: 0.94 / 0.99	45: 0.53 / 0.64	–
Sudoku	6	Qwen3.8-27B	–	–	100: 0.38 / 0.62
Sudoku	6	Gemma4-26B	–	–	100: 0.26 / 0.49
Sudoku	6	GPT-OSS-20B	8: 0.83 / 0.94	92: 0.38 / 0.45	–
Sudoku	9	Qwen3.8-27B	–	–	100: 0.05 / 0.36
Sudoku	9	Gemma4-26B	–	–	100: 0.07 / 0.33
Sudoku	9	GPT-OSS-20B	–	96: 0.10 / 0.35	4: 0.23 / 0.44
Sudoku	12	Qwen3.8-27B	–	–	100: 0.02 / 0.34
Sudoku	12	Gemma4-26B	–	–	100: 0.12 / 0.31
Sudoku	12	GPT-OSS-20B	–	99: 0.09 / 0.33	1: 0.04 / 0.35
Three-zone DM	24	Qwen3.8-27B	–	1: 0.00 / 0.33	99: 0.26 / 0.47
Three-zone DM	24	Gemma4-26B	–	–	100: 0.07 / 0.35
Three-zone DM	24	GPT-OSS-20B	–	100: 0.03 / 0.32	–
Three-zone DM	48	Qwen3.8-27B	–	–	100: 0.07 / 0.33
Three-zone DM	48	Gemma4-26B	–	–	100: 0.05 / 0.32
Three-zone DM	48	GPT-OSS-20B	–	99: 0.08 / 0.33	1: 0.19 / 0.17
Three-zone DM	72	Qwen3.8-27B	–	1: 0.00 / 0.33	99: 0.03 / 0.33
Three-zone DM	72	Gemma4-26B	–	–	100: 0.05 / 0.33
Three-zone DM	72	GPT-OSS-20B	–	99: 0.06 / 0.32	1: 0.07 / 0.39
Three-zone DM	96	Qwen3.8-27B	–	–	100: 0.03 / 0.35
Three-zone DM	96	Gemma4-26B	–	–	100: 0.06 / 0.33
Three-zone DM	96	GPT-OSS-20B	–	100: 0.07 / 0.33	–

K PROMPTS AND ONE TRIAD OF EACH FAMILY

The release compiles to 17,976 requests. The API models ran at medium reasoning effort; the open-weight models ran locally at temperature 0 with a reasoning budget of 8,192 tokens and an output cap of 12,288 tokens. Reasoning trajectories were read for the three open-weight models only: 3,600 must-answer traces per model, six cuts each, 64,800 probability readings. Every prompt states the definitions of NONE, UNIQUE and MULTIPLE; the joint and pairwise prompts state that labels are arbitrary and must not be used to infer an answer. Every response must be a strict JSON object; malformed, refused and truncated responses count as wrong answers.

separate_must_answer

```
You are given one finite-domain constraint instance.
Classify the complete feasible value set of the stated target:
- NONE: no feasible target value exists.
- UNIQUE: exactly one feasible target value exists.
- MULTIPLE: more than one feasible target value exists.
You must choose NONE, UNIQUE, or MULTIPLE even if you are not fully certain. UNRESOLVED is not
↪  allowed.
Populate both response fields exactly:
- NONE: state is NONE and answer is an empty string.
- UNIQUE: state is UNIQUE and answer is the exact unique target value in the notation
↪  specified by the PUBLIC ANSWER INTERFACE. The answer field is a JSON string even when the
↪  value is numeric.
- MULTIPLE: state is MULTIPLE and answer is an empty string.
Return exactly one JSON object matching the supplied strict schema and no other text.
```

separate_optional

```
You are given one finite-domain constraint instance.
```

```
Classify the complete feasible value set of the stated target:
- NONE: no feasible target value exists.
- UNIQUE: exactly one feasible target value exists.
- MULTIPLE: more than one feasible target value exists.
- UNRESOLVED: you cannot determine the state reliably.
Populate both response fields exactly:
- NONE: state is NONE and answer is an empty string.
- UNIQUE: state is UNIQUE and answer is the exact unique target value in the notation
↪  specified by the PUBLIC ANSWER INTERFACE. The answer field is a JSON string even when the
↪  value is numeric.
- MULTIPLE: state is MULTIPLE and answer is an empty string.
- UNRESOLVED: state is UNRESOLVED and answer is an empty string.
Use UNRESOLVED instead of guessing only when you cannot establish the state reliably.
Return exactly one JSON object matching the supplied strict schema and no other text.
```

joint_free

```
You are given one common finite-domain constraint base and three alternative pivots labelled
↪  A, B, and C. For each label, independently substitute that pivot into the single <PIVOT
↪  SLOT>.
Classify the complete feasible value set of the stated target:
- NONE: no feasible target value exists.
- UNIQUE: exactly one feasible target value exists.
- MULTIPLE: more than one feasible target value exists.
- UNRESOLVED: you cannot determine a state reliably.
The labels are arbitrary identifiers. Do not infer an answer from label name, order, or
↪  position, and do not assume that the three states occur once each.
Return exactly one JSON object matching the supplied strict schema and no other text.
```

joint_free_must

```
You are given one common finite-domain constraint base and three alternative pivots labelled
↪  A, B, and C. For each label, independently substitute that pivot into the single <PIVOT
↪  SLOT>.
Classify the complete feasible value set of the stated target:
- NONE: no feasible target value exists.
- UNIQUE: exactly one feasible target value exists.
- MULTIPLE: more than one feasible target value exists.
You must choose NONE, UNIQUE, or MULTIPLE for every pivot even if you are not fully certain.
↪  UNRESOLVED is not allowed.
The labels are arbitrary identifiers. Do not infer an answer from label name, order, or
↪  position, and do not assume that the three states occur once each.
Return exactly one JSON object matching the supplied strict schema and no other text.
```

joint_forced

```
You are given one common finite-domain constraint base and three alternative pivots labelled
↪  A, B, and C. For each label, independently substitute that pivot into the single <PIVOT
↪  SLOT>.
Across the three pivots, the ground truth contains exactly one NONE, exactly one UNIQUE, and
↪  exactly one MULTIPLE. Assign a complete bijection and use each state exactly once.
Classify the complete feasible value set of the stated target:
- NONE: no feasible target value exists.
- UNIQUE: exactly one feasible target value exists.
- MULTIPLE: more than one feasible target value exists.
UNRESOLVED is not allowed. The labels are arbitrary identifiers; do not infer an answer from
↪  label name, order, or position.
Return exactly one JSON object matching the supplied strict schema and no other text.
```

pairwise_distinct

```
You are given one common finite-domain constraint base and two alternative pivots labelled A
↪  and B. For each label, independently substitute that pivot into the single <PIVOT SLOT>.
The two pivots have different ground-truth states. Each state is one of NONE, UNIQUE, and
↪  MULTIPLE. Exactly one of the three states is absent, but you are not told which one.
Determine the state of each pivot. Assign different states to A and B.
Classify the complete feasible value set of the stated target:
- NONE: no feasible target value exists.
- UNIQUE: exactly one feasible target value exists.
- MULTIPLE: more than one feasible target value exists.
UNRESOLVED is not allowed. The labels are arbitrary identifiers; do not infer an answer from
↪  label name, order, or position.
Return exactly one JSON object matching the supplied strict schema and no other text.
```

separate_must_answer_opaque

```
You are given one finite-domain constraint instance. Classify the complete feasible value set
↪  of the stated target using only this temporary codebook:
- KAV: no feasible target value exists.
- LUM: exactly one feasible target value exists.
- ZOR: more than one feasible target value exists.
You must choose exactly one code even if you are not fully certain. UNRESOLVED and the words
↪  NONE, UNIQUE, and MULTIPLE are not allowed in the response.
Return exactly one JSON object matching the supplied strict schema and no other text.
```

clarification_given_u

```
You are given one finite-domain constraint instance.
An independent exact verifier has established that the complete feasible value set of the
↪  stated target contains exactly two distinct values (MULTIPLE).
Do not classify the state and do not return either target value.
Invent one atomic positive equality question using the PUBLIC CLARIFICATION INTERFACE. The
↪  query must not directly ask for the target. Both Yes and No must remain feasible, and
↪  either reply must uniquely determine a different target value.
Return exactly one JSON object matching the supplied strict schema and no other text.
```

Sudoku, $N{=}4$: the three siblings and the joint forced input

Sibling with truth NONE

```
INSTANCE B:
Solve this order-4 rectangular Sudoku. Symbols are 1, 2, 3, 4.
Each box has 2 rows and 2 columns. Every row, column, and box contains every symbol exactly
↪  once.
Fixed clues are printed as plain values.

Grid:
      C01 C02 C03 C04
R01: . . | 3 .
R02: . . | . .

R03: . . | . 4
R04: 3 . | . .

Target question: What value is in R01C02?

PUBLIC ANSWER INTERFACE:
{"target_domain_values":["1","2","3","4"],"target_variable":"R01C02","unique_answer_contract"⌋
↪  :{"field":"answer","format":"one exact target_domain_values string"}}
```

Sibling with truth UNIQUE

```
INSTANCE C:
Solve this order-4 rectangular Sudoku. Symbols are 1, 2, 3, 4.
Each box has 2 rows and 2 columns. Every row, column, and box contains every symbol exactly
↪  once.
Fixed clues are printed as plain values.

Grid:
      C01 C02 C03 C04
R01: . . | 3 .
R02: . . | . .

R03: . . | . 4
R04: 4 . | . .

Target question: What value is in R01C02?

PUBLIC ANSWER INTERFACE:
{"target_domain_values":["1","2","3","4"],"target_variable":"R01C02","unique_answer_contract"⌋
↪  :{"field":"answer","format":"one exact target_domain_values string"}}
```

Sibling with truth MULTIPLE

```
INSTANCE A:
Solve this order-4 rectangular Sudoku. Symbols are 1, 2, 3, 4.
Each box has 2 rows and 2 columns. Every row, column, and box contains every symbol exactly
↪  once.
Fixed clues are printed as plain values.

Grid:
      C01 C02 C03 C04
R01: . . | 3 .
```

```
R02: . . | . .

R03: . . | . 4
R04: 1 . | . .

Target question: What value is in R01C02?

PUBLIC ANSWER INTERFACE:
{"target_domain_values":["1","2","3","4"],"target_variable":"R01C02","unique_answer_contract"⌋
↪  :{"field":"answer","format":"one exact target_domain_values string"}}
```

Joint-forced input for the same triad

```
COMMON BASE (use unchanged for all alternatives):
Solve this order-4 rectangular Sudoku. Symbols are 1, 2, 3, 4.
Each box has 2 rows and 2 columns. Every row, column, and box contains every symbol exactly
↪  once.
Fixed clues are printed as plain values.

Grid:
      C01 C02 C03 C04
R01: . . | 3 .
R02: . . | . .

R03: . . | . 4
R04: <PIVOT SLOT> . | . .

Target question: What value is in R01C02?

PIVOT ALTERNATIVES (substitute one at a time):
PIVOT A: 1
PIVOT B: 3
PIVOT C: 4
```

CrossChoice, $N{=}7$: the UNIQUE sibling in full

```
INSTANCE B:
Boolean constraint problem. Variable values are 0 or 1.
Target: V-VFALTV
CrossChoice constraints; each line requires both parenthesized OR groups:
C-0001: CrossChoice((NOT V-VFALTV OR V-N406WH OR NOT V-H3Z76M), (V-1WYEEH OR NOT V-4OH964 OR
↪  NOT V-M9B2TJ))
C-0002: CrossChoice((NOT V-VFALTV OR V-1WYEEH OR NOT V-M9B2TJ), (V-4OH964 OR V-N406WH OR
↪  V-AXP5L4))
C-0003: CrossChoice((NOT V-1WYEEH OR V-H3Z76M OR V-N406WH), (NOT V-AXP5L4 OR V-M9B2TJ OR NOT
↪  V-4OH964))
C-0004: CrossChoice((V-VFALTV OR NOT V-H3Z76M OR V-1WYEEH), (NOT V-M9B2TJ OR NOT V-4OH964 OR
↪  V-AXP5L4))
C-0005: CrossChoice((NOT V-H3Z76M OR NOT V-N406WH OR V-4OH964), (NOT V-VFALTV OR V-AXP5L4 OR
↪  V-M9B2TJ))
C-0006: CrossChoice((NOT V-H3Z76M OR V-M9B2TJ OR NOT V-VFALTV), (V-AXP5L4 OR NOT V-N406WH OR
↪  NOT V-1WYEEH))
C-0007: CrossChoice((V-4OH964 OR V-M9B2TJ OR NOT V-N406WH), (V-1WYEEH OR V-VFALTV OR
↪  V-AXP5L4))
C-0008: CrossChoice((NOT V-1WYEEH OR NOT V-N406WH OR NOT V-H3Z76M), (V-4OH964 OR V-VFALTV OR
↪  NOT V-M9B2TJ))
C-0009: CrossChoice((NOT V-1WYEEH OR NOT V-N406WH OR NOT V-M9B2TJ), (NOT V-4OH964 OR NOT
↪  V-H3Z76M OR V-AXP5L4))
C-0010: CrossChoice((V-VFALTV OR V-AXP5L4 OR NOT V-H3Z76M), (V-1WYEEH OR V-4OH964 OR
↪  V-N406WH))
C-0011: CrossChoice((V-4OH964 OR V-1WYEEH OR V-M9B2TJ), (V-N406WH OR V-AXP5L4 OR NOT
↪  V-VFALTV))
C-0012: CrossChoice((NOT V-N406WH OR V-1WYEEH OR NOT V-H3Z76M), (NOT V-VFALTV OR V-AXP5L4 OR
↪  NOT V-M9B2TJ))
C-0013: CrossChoice((NOT V-VFALTV OR V-AXP5L4 OR V-1WYEEH), (V-N406WH OR V-4OH964 OR NOT
↪  V-H3Z76M))
C-0014: CrossChoice((NOT V-VFALTV OR NOT V-H3Z76M OR NOT V-AXP5L4), (NOT V-4OH964 OR NOT
↪  V-N406WH OR NOT V-M9B2TJ))
C-0015: CrossChoice((NOT V-N406WH OR V-4OH964 OR V-1WYEEH), (NOT V-M9B2TJ OR NOT V-H3Z76M OR
↪  V-AXP5L4))

PUBLIC ANSWER INTERFACE:
{"target_value_domain":["0","1"],"unique_answer_contract":{"field":"answer","format":"the
↪  exact string 0 or 1"}}
```

The other two siblings The NONE sibling is byte-identical below the label line except that `C-0003: CrossChoice((NOT V-1WYEEH OR V-H3Z76M OR V-N406WH), (NOT V-AXP5L4 OR V-M9B2TJ OR NOT V-4OH964))` becomes `C-0003:`

`CrossChoice((NOT V-1WYEEH OR NOT V-AXP5L4 OR V-N406WH), (V-M9B2TJ OR V-H3Z76M OR NOT V-4OH964))`. The MULTIPLE sibling is byte-identical below the label line except that `C-0003: CrossChoice((NOT V-1WYEEH OR V-H3Z76M OR V-N406WH), (NOT V-AXP5L4 OR V-M9B2TJ OR NOT V-4OH964))` becomes `C-0003: CrossChoice((NOT V-1WYEEH OR V-N406WH OR V-M9B2TJ), (NOT V-4OH964 OR NOT V-AXP5L4 OR V-H3Z76M))`. Certified states: NONE, UNIQUE, MULTIPLE.

Three-zone DM, $N{=}24$: the UNIQUE sibling in full

```
INSTANCE B:
A valid world is a perfect matching between the left and right vertices: every vertex has
↪  exactly one partner and every selected pair must be an available edge.

Vertices: L000, L001, L002, L003, L004, L005, L006, L007, L008, L009, L010, L011, L012, L013,
↪  L014, L015, L016, L017, L018, L019, L020, L021, L022, L023, R000, R001, R002, R003, R004,
↪  R005, R006, R007, R008, R009, R010, R011, R012, R013, R014, R015, R016, R017, R018, R019,
↪  R020, R021, R022, R023.

Available edges: L017--R007, L021--R012, L008--R001, L022--R015, L009--R020, L003--R007,
↪  L012--R000, L017--R018, L004--R012, L012--R018, L012--R019, L014--R018, L023--R007,
↪  L020--R004, L002--R001, L016--R015, L007--R003, L010--R009, L017--R003, L023--R017,
↪  L000--R003, L022--R018, L010--R012, L009--R012, L006--R000, L012--R011, L009--R021,
↪  L008--R006, L014--R000, L004--R016, L014--R012, L010--R023, L011--R006, L002--R016,
↪  L011--R015, L003--R017, L022--R000, L009--R011, L004--R000, L004--R001, L010--R014,
↪  L014--R003, L006--R001, L000--R000, L008--R004, L016--R019, L012--R016, L009--R015,
↪  L010--R008, L010--R016, L021--R016, L009--R016, L002--R023, L020--R015, L012--R001,
↪  L002--R014, L011--R013, L018--R018, L005--R019, L018--R019, L002--R012, L005--R003,
↪  L002--R004, L015--R000, L003--R006, L011--R005, L019--R013, L019--R007, L008--R022,
↪  L012--R010, L020--R011, L013--R019, L014--R021, L019--R003, L023--R005, L015--R001,
↪  L023--R013, L004--R021, L019--R019, L020--R010, L000--R018, L022--R003, L000--R019,
↪  L012--R003, L008--R017, L001--R019, L005--R001, L015--R017, L011--R017, L008--R005,
↪  L021--R021, L013--R018, L008--R013, L006--R015, L007--R018, L021--R010, L002--R020,
↪  L021--R015, L000--R001, L018--R012, L011--R003, L008--R007, L022--R019, L003--R022,
↪  L002--R018, L001--R018, L019--R006, L011--R011, L005--R018, L020--R000, L016--R001,
↪  L016--R018, L018--R021, L021--R020, L012--R015, L003--R013, L003--R000, L010--R000,
↪  L008--R018, L018--R003, L017--R013, L015--R007, L019--R018, L015--R013, L001--R003,
↪  L003--R010, L006--R019, L022--R004, L013--R003, L008--R002, L016--R000, L023--R011,
↪  L022--R001, L011--R007, L007--R019, L004--R020, L017--R000, L009--R003, L023--R006,
↪  L015--R004, L003--R019, L023--R022, L002--R009, L021--R008, L010--R010, L021--R011,
↪  L003--R005, L015--R006, L023--R010, L006--R004, L003--R002, L018--R020, L021--R009,
↪  L006--R003, L009--R008, L010--R019, L010--R020, L016--R003, L002--R008, L006--R011,
↪  L004--R004, L006--R018, L023--R015.

Question: In a valid perfect matching, which vertex is L006 paired with?

PUBLIC ANSWER INTERFACE:
{"target_variable":"L006_partner","unique_answer_contract":{"field":"answer","format":"one
↪  exact opposite-side vertex name such as L004 or R004"}}
```

The other two siblings The NONE sibling is byte-identical below the label line except that `L012--R016` becomes `L012--R017`. The MULTIPLE sibling is byte-identical below the label line except that `L012--R016` becomes `L022--R016`. Certified states: NONE, UNIQUE, MULTIPLE.

L BUILDING ONE EDIT, THREE WORLDS ON ANOTHER PROBLEM TYPE

The construction of Section 3 needs a common base B, a target question q, one pivot premise $P(\square)$ with a slot, and an oracle that returns the answer space $A(v)$ for each legal slot value v. The steps are: (1) sample a base; (2) enumerate the legal values of the slot and query the oracle for each candidate target; (3) keep a target and one value per world, here $|A(v)| \in \{0, 1, 2\}$, and discard the base if no target has a value for every world; (4) render the three siblings from one template so that they are byte-identical outside the slot; (5) verify with an independent implementation of the oracle; (6) audit the release with surface-feature classifiers under triad-grouped cross-validation, and, where probabilities are available, read the siblings' state distributions before reasoning. The worlds need not be answer-space sizes: any partition of the oracle's output into outcomes that a correct reasoner must distinguish will do, provided the oracle is exact or its error rate is far below the effects being measured.