DECODING UNDER CONSTRAINT UNCERTAINTY IN DIFFUSION LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Constrained decoding usually assumes a known grammar, schema, or rule set. When constraints must be inferred from positive demonstrations, several hypotheses can explain the same observations, and enforcing one can exclude the correct output. We formalize this setting as *hidden-constraint decoding* and introduce TACTIC, an execution layer that carries constraint uncertainty into candidate selection. TACTIC converts posterior beliefs or confidence-weighted rules into candidate compatibility and combines it with frozen-model evidence. Factorized and tensor-train representations support independent and correlated constraint components. Our analysis establishes a false-exclusion lower bound for point commitment, posterior support preservation, and conditions for decision stability. Across controlled and real-data tasks, TACTIC improves on matched hard execution under scarce support. On CIDDS structured completion, the gains are 19.4 percentage points with DREAM-7B and 20.8 with LLaDA-8B. These results show how retaining plausible constraints allows model evidence to resolve ambiguity left by demonstrations.

1 INTRODUCTION

Constrained decoding has become a practical way to make language models emit valid programs, structured records, and function calls. Existing executors can enforce grammars or automata efficiently, including during masked-diffusion generation (Dang & Ermon, 2026; Mündler et al., 2026; Suresh et al., 2025). They assume, however, that the constraint is already known. Many applications offer a different interface: a handful of valid examples reveal what is allowed, but no complete schema or validator is available.

Consider a function argument whose demonstrations contain only `red` and `blue`. The legal domain might also contain `green`, or it might contain `yellow`; both hypotheses explain the observations. If the query strongly favors `green`, committing to the second hypothesis removes the best candidate before model evidence can be used. More generally, positive-only data leave a version space of observationally compatible constraints (Gold, 1967; Mitchell, 1977; Bessière et al., 2005). Constraint acquisition typically resolves this ambiguity by selecting one model (Bessière et al., 2017; Wiegand et al., 2024). That is appropriate when identification is the goal, but risky when the learned object will immediately exclude outputs.

We formalize the downstream decision as *hidden-constraint decoding* (HCD): select an output using model evidence and positive demonstrations drawn under an unknown constraint. Building on constraint acquisition and version spaces, HCD makes residual uncertainty relevant to the decoding objective. It distinguishes what the demonstrations identify, how remaining uncertainty is represented, and how that uncertainty affects a query-level decision. Our execution principle is to preserve plausible constraints through candidate selection, so that model evidence can resolve alternatives left open by the demonstrations.

We introduce **TACTIC**, an execution layer that implements this principle. Acquisition produces either a posterior over constraint hypotheses or confidence-weighted rules. TACTIC compiles this object into a bounded compatibility score and combines it with evidence from a frozen diffusion language model. This interface supports exact factorization for independent constraint components and tensor-train (TT) contraction for correlated components.

Code is available in the supplementary material.

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.

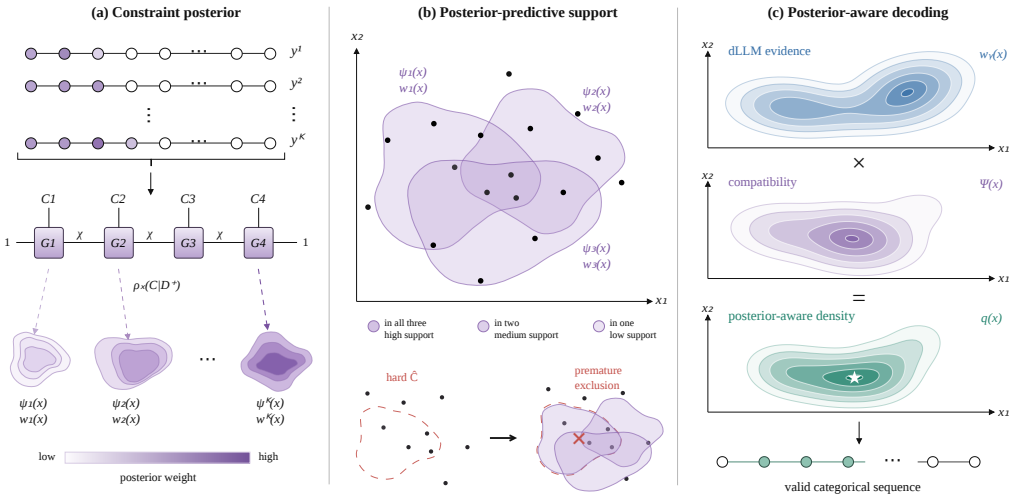


Figure 1: **Hidden-constraint decoding with TACTIC.** Positive demonstrations induce uncertainty over admissible constraints. TACTIC compiles that uncertainty into candidate compatibility, combines it with frozen-model evidence, and selects from the executor's candidate set. Point acquisition followed by hard enforcement is the matched alternative.

The analysis connects false exclusion to support preservation, identification, and decision stability. Across controlled tasks and network records, retaining uncertainty improves on matched hard execution when demonstrations leave useful alternatives unresolved. Support curves reveal when this advantage narrows, and the correlated-posterior study shows how dependence and model specification affect candidate selection.

Our contributions are:

1. We formalize HCD as a downstream decision problem linking constraint acquisition to decoding, and identify how point commitment can exclude valid candidates under unresolved ambiguity.
2. We develop TACTIC, which preserves constraint uncertainty through candidate compatibility, and establish support-preservation, consistency, and decision-stability guarantees for its posterior formulation.
3. We demonstrate gains over matched hard execution under scarce support and characterize how support, domain coverage, and posterior dependence affect those gains.

2 HIDDEN-CONSTRAINT DECODING

For a query or diffusion state y, an executor exposes a finite candidate set $\mathcal{X}_y$. A constraint hypothesis $C \in \mathcal{H}$ has acceptance function and support

$$A_C(x) \in \{0,1\}, \qquad S_C = \{x : A_C(x) = 1\}. \tag{1}$$

The ground-truth constraint $C^\star$ is unavailable. The system observes valid demonstrations, modeled in the analysis as

$$D_n^+ = (X_1, \ldots, X_n), \qquad X_i \overset{\text{iid}}{\sim} P_{C^\star}^+, \quad \operatorname{supp}(P_{C^\star}^+) \subseteq S_{C^\star}, \tag{2}$$

and receives nonnegative candidate evidence $w_y(x)$ from a frozen model.

Definition 1 (Hidden-Constraint Decoding)**.** *Given $(y, \mathcal{X}_y, w_y, D_n^+, \mathcal{H})$ but not $C^\star$ or its validator, hidden-constraint decoding (HCD) selects a candidate using both query evidence and the constraints supported by the demonstrations. Performance is measured by task utility or exact completion, together with validity under $C^\star$ when it is observable.*

HCD evaluates the decision made with acquired constraints while several hypotheses still explain the same demonstrations. Uncertainty matters when these hypotheses rank candidates differently for the current query, motivating the compatibility interface and margin analysis below.

If $C^\star$ were supplied, an idealized executor would rank candidates by $w_y(x)A_{C^\star}(x)$. HCD cannot evaluate this score directly. Its finite-data version space

$$\mathcal{V}(D_n^+) = \{C : P_C^+(D_n^+) > 0\}$$

can contain hypotheses that agree on the demonstrations but disagree on the query. This is the source of constraint uncertainty; it is distinct from uncertainty in the language model.

A point pipeline estimates one constraint and enforces it:

$$q_{\mathrm{hard}}(x \mid y, D_n^+) \propto w_y(x)A_{\widehat{C}(D_n^+)}(x). \tag{3}$$

When plausible supports are non-nested, this operation can assign zero mass to a candidate accepted by another data-compatible hypothesis. HCD therefore calls for an acquisition–execution interface that can retain ambiguity until the query evidence is applied. In proposal-based settings, performance is measured relative to the documented $\mathcal{X}_y$; candidate recall is reported separately from selection quality.

3 TACTIC

TACTIC preserves acquisition uncertainty through candidate selection. The acquisition module returns a posterior or confidence-weighted rules. A shared compatibility interface evaluates each candidate against this information, then combines compatibility with frozen-model evidence for execution.

3.1 POSTERIOR-PREDICTIVE COMPATIBILITY

For a coherent Bayesian acquisition model,

$$\rho_n(C) = P(C \mid D_n^+) \propto \pi(C)P_C^+(D_n^+).$$

Rather than selecting its mode, TACTIC marginalizes the acceptance decision:

$$\Psi_{\rho_n}^{\mathrm{PP}}(x) = \mathbb{E}_{C\sim\rho_n}[A_C(x)] = P_{C\sim\rho_n}(x \in S_C). \tag{4}$$

This posterior membership probability averages the acceptance decisions of plausible constraints (Tenenbaum & Griffiths, 2001). It retains their disagreement in a scalar compatibility score.

Let $Z_1, \ldots, Z_m$ encode local constraint components and let $a_r(z_r, x)$ indicate local acceptance. If $A(z, x) = \prod_r a_r(z_r, x)$ and the posterior factorizes, compatibility is

$$\Psi_{\rho_n}^{\mathrm{PP}}(x) = \prod_{r=1}^{m} \underbrace{\sum_{z_r} \rho_{n,r}(z_r)a_r(z_r, x)}_{\psi_r(x)}. \tag{5}$$

For a binary rule active with probability π_r, a violated rule contributes $1 - \pi_r$; satisfied or irrelevant rules contribute one.

3.2 CORRELATED CONSTRAINT UNCERTAINTY

Independent marginals are insufficient when constraint components co-vary. We represent a correlated posterior as a tensor train (Oseledets, 2011; Cichocki et al., 2016):

$$\rho_\chi(z_1, \ldots, z_m) = G_1[z_1]G_2[z_2]\cdots G_m[z_m], \tag{6}$$

where $G_r[z_r] \in \mathbb{R}^{\chi_{r-1}\times\chi_r}$, $\chi_0 = \chi_m = 1$, and $\chi = \max_r \chi_r$. The represented tensor must be nonnegative and sum to one; individual core entries need not be nonnegative. Compatibility follows by replacing each core with $M_r(x) = \sum_{z_r} G_r[z_r]a_r(z_r, x)$:

$$\Psi_{\rho_\chi}^{\mathrm{PP}}(x) = M_1(x)\cdots M_m(x). \tag{7}$$

Given the cores, forming and contracting $M_r(x)$ costs $O(md_{\max}\chi^2)$ per candidate, where $d_{\max} = \max_r |\mathcal{Z}_r|$. TT rank captures dependence among constraint components (Figure 2); construction and approximation details are in Appendix F.

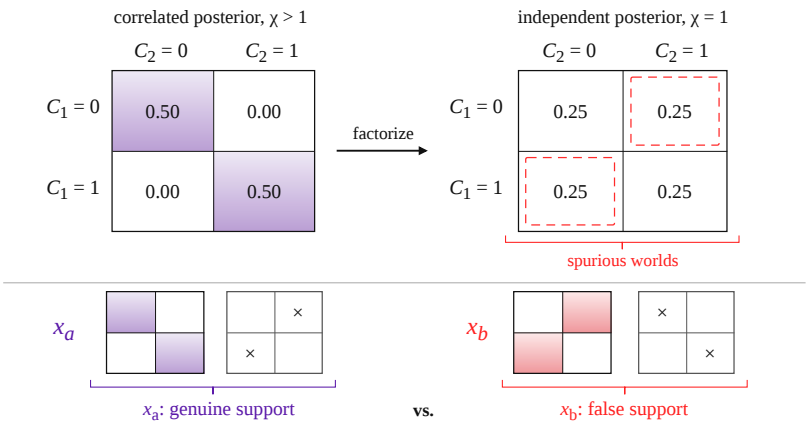


Figure 2: **Why correlated posteriors matter.** Independent marginals assign mass to component combinations absent from the correlated posterior (dashed). A candidate accepted only in these spurious worlds (x_b) then acquires positive compatibility.

3.3 Weighted-rule acquisition

Some acquisition procedures produce rule confidences rather than a joint distribution. For rule r, let $\omega_r \geq 0$ be its confidence, $f_r(x)$ indicate that its antecedent fires, and $b_r(x)$ indicate satisfaction. We use

$$\Psi_D^{\mathrm{WM}}(x) = \begin{cases} \dfrac{\sum_r \omega_r f_r(x) b_r(x)}{\sum_r \omega_r f_r(x)}, & \sum_r \omega_r f_r(x) > 0, \\ 1, & \text{otherwise.} \end{cases} \tag{8}$$

This confidence-weighted satisfaction score supplies CIDDS compatibility.

3.4 Evidence composition and selection

The frozen dLLM supplies evidence for every proposed completion. For masked fields,

$$\log w_y(x) = \sum_{f \in \textsc{Fields}(y)} e_{\theta,f}(x_f; y). \tag{9}$$

Here $e_{\theta,f}$ is a cached field log-score; task-specific extraction and backbones are specified in Appendices C.7 and J.1. Given either posterior-predictive or weighted-rule compatibility Ψ_D, and finite nonnegative exponents, the idealized composed distribution is

$$q_\lambda(x \mid y, D_n^+) \propto w_y(x)^{\lambda_w} \Psi_D(x)^{\lambda_C}. \tag{10}$$

A zero exponent omits its factor. For posterior compatibility, unit weights maximize expected utility for $u(x, C) = w_y(x) A_C(x)$. We use log-additive scoring or development-selected channel scaling (Appendix C).

The protocol fixes channel weights, zero handling, scaling domains, and fallback (default $\perp$). For the product rule, $u = \log w_y$ and $v = \log c$ on positive active channels; a zero-weight channel is omitted. CIDDS scales log scores, while GraphColor scales probabilities before taking logs. Algorithm 1 covers candidate rescoring; JSON Schema thresholds compatibility and BFCL uses it in weighted-automaton generation (Appendix C). Acquisition is reused across queries sharing support.

The two channels resolve different ambiguities: compatibility measures agreement with the acquired constraints, while model evidence distinguishes completions in the query context. Candidates accepted by different plausible hypotheses can therefore remain available for comparison. Their relative compatibility enters the ranking continuously, allowing evidence to resolve the remaining choice without first committing to a single constraint.

Algorithm 1: TACTIC candidate selection

Input: Support D_n^+; acquisition model $\mathcal{M}$; ordered candidates $\mathcal{X}_y$; evidence w_y; fixed scoring protocol $\mathcal{P}$
Output: Selected candidate or abstention $\perp$

$\mathcal{U} \leftarrow \text{ACQUIRE}_{\mathcal{M}}(D_n^+)$
foreach $x \in \mathcal{X}_y$ **do**
  $c(x) \leftarrow \Psi_{\mathcal{U}}(x)$ using equations (4), (7) and (8)
$\mathcal{F}_y \leftarrow \text{ELIGIBLE}_{\mathcal{P}}(\mathcal{X}_y, w_y, c)$
if $\mathcal{F}_y = \varnothing$ **then**
  return $\text{Fallback}_{\mathcal{P}}(w_y)$
$(u, v) \leftarrow \text{SCALE}_{\mathcal{P}}(w_y, c; \mathcal{F}_y)$
foreach $x \in \mathcal{F}_y$ **do**
  $s(x) \leftarrow \lambda_w u(x) + \lambda_C v(x)$
return *first maximizer of s in the supplied candidate order*

4 ANALYSIS

We first characterize false exclusion, then show how posterior consistency and approximation affect candidate selection. Proofs and the extension to min–max scoring are in Appendix A.

Theorem 1 (False exclusion under point commitment)**.** *Fix $D_n^+, y, \mathcal{X}_y$. Suppose $C_1, C_2 \in \mathcal{V}(D_n^+)$ have witnesses $x_1 \in (S_{C_1} \setminus S_{C_2}) \cap \mathcal{X}_y$ and $x_2 \in (S_{C_2} \setminus S_{C_1}) \cap \mathcal{X}_y$. A policy whose output lies in $\{C_1, C_2\}$ has worst-case probability at least $1/2$ of excluding a valid candidate, over these two compatible ground truths. For a deterministic policy this probability is one. If commitment is independent of w_y, the excluded witness can carry arbitrarily close to all normalized model-evidence mass.*

The probability is over the policy's randomization, conditional on the observed version space. A posterior representation can retain both witnesses until query evidence is applied.

Proposition 2 (Posterior support preservation)**.** *Let $\mathcal{H}_x = \{C : A_C(x) = 1\}$. Then $\Psi_{\rho_n}^{\mathrm{PP}}(x) = \rho_n(\mathcal{H}_x)$. If this mass is positive and $w_y(x) > 0$, candidate x has positive probability in equation* (10) *for finite $\lambda_w \geq 0$ and $\lambda_C > 0$. An accepting hypothesis with positive posterior mass therefore suffices.*

Theorem 3 (Posterior consistency and compatibility convergence)**.** *Suppose demonstrations are iid under $C^\star$ in a finite, well-specified family $\mathcal{H}$ with $\pi(C^\star) > 0$. If $0 < D_{\mathrm{KL}}(P_{C^\star}^+ \| P_C^+) < \infty$ for every $C \neq C^\star$ with $\pi(C) > 0$, then, almost surely,*

$$\rho_n(C^\star) \to 1, \qquad \sup_x |\Psi_{\rho_n}^{\mathrm{PP}}(x) - A_{C^\star}(x)| \leq 1 - \rho_n(C^\star) \to 0.$$

The sampling distributions must be distinguishable. On a fixed finite candidate set, consistency implies eventual agreement with a unique positive-score supplied-constraint winner under fixed positive weights (Appendix A.3). Min–max scaling requires separate range control (Appendix A.6). Appendix J.4 maps these assumptions to the benchmark acquisition models.

Posterior dependence can require higher TT rank. The standard Dicke-state Schmidt-rank result (Moreno & Parisio, 2018) gives the following exact reference family.

Theorem 4 (Rank of fixed-cardinality uncertainty)**.** *Let $m \geq 2$, $0 \leq K \leq m$, and $z \in \{0,1\}^m$. For $T_{m,K}(z) = \mathbf{1}[\sum_{r=1}^m z_r = K]$, the minimum exact TT bond dimension at cut $1 \leq i < m$ is $r_i = \min(i, K, m-i, m-K)+1$. The minimum possible maximum bond dimension is $\min(K, m-K)+1$.*

These ranks also hold for the uniform distribution on the same assignments; nonuniform weights can change them.

Table 1: **Matched hard and uncertainty-aware execution.** Scores are exact completion (CatSup, GraphColor, CIDDS), balanced validity accuracy (JSON Schema), or AST accuracy (BFCL), in percent. Δ is TACTIC minus Hard in percentage points. Full support curves and evaluation protocols are in Appendices D, E and G; JSON Schema and BFCL use the additional development calibrations in Appendices J.6 and J.7.

		Matched comparison			References	
Benchmark	N_s	Hard	TACTIC	Δ [95% CI]	Open	Supplied-C
CatSup	4	14.6	**43.3**	+28.8 [22.5, 35.0]	50.0	50.0
GraphColor (n=12)	2	60.0	**70.0**	+10.0 [3.3, 18.3]	0.0	68.3
CIDDS (24-cell macro)	4–32	52.1	**71.5**	+19.4 [18.1, 20.7]	12.0	57.3
JSON Schema	8	83.5	**83.8**	+0.3 [0.05, 0.56]	—	—
BFCL	8	48.2	**50.4**	+2.2 [−1.6, 6.0]	53.5	57.8

Proposition 5 (Compatibility-margin stability)**.** *On a fixed finite candidate set $\mathcal{F}$, suppose $w_y, \Psi, \widehat{\Psi} > 0$, and fix finite $\lambda_w \geq 0$, $\lambda_C > 0$. Let $x^\star$ uniquely maximize $s_\Psi(x) = \lambda_w \log w_y(x) + \lambda_C \log \Psi(x)$ with margin $\gamma > 0$. If*

$$\| \log \widehat{\Psi} - \log \Psi \|_{\infty,\mathcal{F}} < \frac{\gamma}{2\lambda_C},$$

the winner is unchanged. For probability distributions $\rho, \widehat{\rho}$, if both *posterior-predictive compatibilities are at least $\epsilon > 0$ on $\mathcal{F}$, a sufficient condition is $2\lambda_C \|\rho - \widehat{\rho}\|_{\mathrm{TV}} < \epsilon\gamma$.*

The common set $\mathcal{F}$ must contain every candidate competing in either decision; changes in zero support must be handled separately. For min–max scores, Proposition 6 gives the corresponding bound, including the change in scale after approximation.

5 EXPERIMENTS

We examine how uncertainty-aware execution changes completion accuracy, how its value changes with support, and how posterior dependence affects decisions. CatSup and GraphColor provide controlled acquisition models; CIDDS tests network-record completion; JSONSchemaBench and BFCL test schema validity and function calling.

Evaluation design. Matched arms share demonstrations and evaluation units, varying how acquired constraints are enforced. Rescoring tasks also share candidates and frozen evidence; BFCL shares its model and diffusion executor. Open and supplied-constraint execution provide reference points. We measure exact completion, balanced validity accuracy, or official AST accuracy, with task-specific paired intervals (Table 1). Full protocols are in Appendix C.

5.1 PREMATURE COMMITMENT UNDER AMBIGUITY

Categorical support. CatSup asks the model to complete categorical function arguments under hidden mutual-exclusion, dependency, and branch-coupling rules. Candidates enumerate the full argument product; demonstrations reveal legal calls. The original study follows repeated support draws for three rules and six fixed queries. With four demonstrations, TACTIC improves exact completion over hard commitment by 28.8 points. This advantage narrows as demonstrations identify the support, and the trend persists across backbones (Appendix D.1). A separate holdout study excludes queries from support and extends the comparison to independently generated rules: the average gain over hard execution is 31.2 points, while composition and Open have comparable accuracy (Appendix J.5).

Hidden GraphColor. GraphColor hides a graph's edge set and supplies valid colorings as demonstrations. Each query fixes half the vertices of a target coloring and asks for the rest. Exact candidate enumeration separates acquisition and selection from search errors. With scarce support, soft compatibility improves exact completion over hard enforcement (Table 1). Effects shrink and become

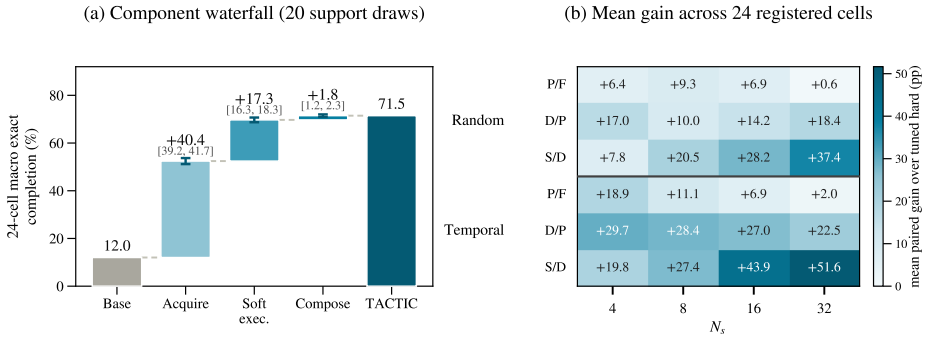


Figure 3: **CIDDS component and robustness analysis.** (a) Acquisition, compatibility, and model evidence each improve completion; whiskers show paired confidence intervals for the increments. (b) Gains over tuned-hard execution persist across mask, support, and split conditions. The horizontal rule separates random and temporal splits.

mixed with additional demonstrations. Evidence-only exact success is zero under both LLaDA-8B and DREAM-7B, while development-selected mixing weights vary by graph size (Appendix D.1). A task-adapted two-part-MDL comparator (Rissanen, 1978; Wiegand et al., 2024) acquires no edge at matched small support; its coding-cost criterion and official-engine comparisons appear in Appendices E and G. Support curves and paired intervals are in Appendix D (fig. 5).

5.2 Real hidden constraints

CIDDS structured completion. CIDDS is a public enterprise NetFlow benchmark (Ring et al., 2017). We mask three field-pair types (P/F, D/P, and S/D), acquire confidence-weighted implications from $N_s \in \{4, 8, 16, 32\}$ valid records, and rank documented completion candidates using DREAM-7B evidence. The primary estimate averages 24 mask, split, and support-size cells within each of 20 independent support draws.

TACTIC improves exact completion over tuned-hard enforcement by 19.4 points with DREAM-7B and 20.8 with LLaDA-8B. Every support draw favors TACTIC. Compatibility supplies most of the gain, while model evidence adds a consistent improvement across backbones (Figure 3 and appendix D.1).

Table 2: CIDDS comparisons with generative ICL and fallback hard execution. Accuracy is in percent; paired differences and confidence intervals are in percentage points. All arms use common support draws and test records.

Arm / contrast	DREAM	LLaDA
Open	12.02	15.58
Compatibility only	69.70	69.70
Original tuned-hard	52.06	52.08
Tuned-hard + fallback, dev-retuned	53.37	55.15
Hard-Ψ + fallback, dev-tuned	40.31	44.78
TACTIC	71.47	72.85
ICL generation	45.48	—
TACTIC − ICL	+25.99 [24.90, 27.08]	—
TACTIC − retuned fallback hard	+18.10 [16.71, 19.50]	+17.70 [16.34, 19.05]

The same-support comparisons in Table 2 test two practical alternatives: placing demonstrations directly in the prompt, and allowing hard execution to fall back when no completion survives. TACTIC retains an advantage over both. Hard-Ψ applies a development-selected threshold to the same compatibility score before ranking accepted candidates by model evidence. Its lower accuracy supports using graded compatibility in the ranking itself. Transition analysis identifies re-ranking

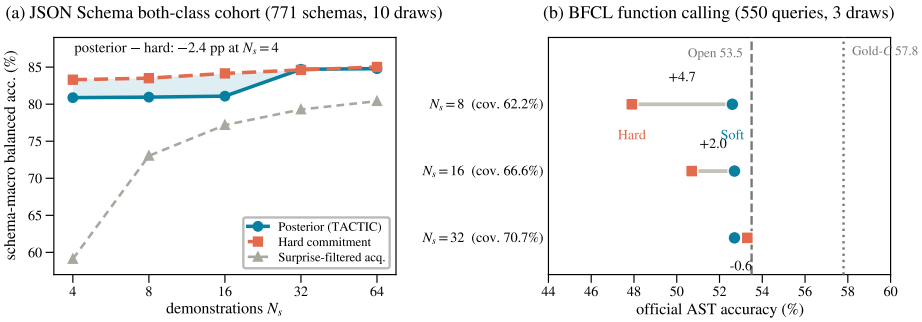


Figure 4: **Support and coverage under the original frozen settings.** (a) JSON Schema validity classification; shading shows the posterior–hard accuracy gap. (b) BFCL function calling, with domain coverage in row labels. Dashed and dotted lines mark Open and supplied-domain execution.

among feasible candidates as the main source of corrected errors, alongside recovery from false exclusion (Appendix J.2).

5.3 Support and coverage transition

JSONSchemaBench. We classify schema validity on a complete cohort with both valid and invalid tests. Rebuilt development data select $\tau = .2$ on schema-macro balanced accuracy at $N_s = 8$. At that support size, posterior execution slightly exceeds hard commitment; it trails at the other sizes. The threshold changes the tradeoff between accepting valid instances and rejecting invalid ones, so calibration at one support budget transfers unevenly to others. The original frozen-threshold curve remains in Figure 4; the recalibrated results are in Appendix J.6.

BFCL hidden enums. BFCL supplies function-calling queries and an official AST-based evaluator (Patil et al., 2025). Official AST accuracy on a separate 45-query pilot selects $p_0 = .85$. On the original evaluation pool at a matched support draw, this yields a smaller soft–hard advantage with a paired interval spanning zero. Open remains stronger than recalibrated soft execution. The original support curve shows hard execution catching up as enum coverage improves; Appendix J.7 reports the disjoint-pilot comparison and its selection protocol.

5.4 Correlated posteriors

We compare an independent-edge model, exact family-conditioned posteriors, and a product of their exact edge marginals (PoM), using common candidates and evidence. PoM retains the marginal changes induced by conditioning while removing joint dependence.

Table 3: Separating marginal changes from joint dependence. Exact and PoM share all edge marginals; Uncond. uses the original independent-edge model. Accuracy is in percent; paired differences are in percentage points.

Family	Uncond.	PoM	Exact	Exact − PoM [95% CI]
Fixed-$K = 4$	32.50	5.83	6.67	$+0.83$ $[0.00, 2.50]$
Fixed-$K = 8$	32.50	18.33	22.50	$+4.17$ $[0.83, 7.50]$
Degree sequence	28.12	42.71	41.67	-1.04 $[-3.12, 0.00]$
Connected	28.12	28.12	28.12	0.00 $[0.00, 0.00]$

PoM reproduces the degree-sequence gain and fixed-cardinality losses relative to the independent model (Table 3), attributing most of the conditioning effect to changed marginals. Joint dependence adds up to 4.2 points, with the largest benefit in the misspecified fixed-cardinality family. TT reproduces exact-posterior decisions at sufficient rank (Appendices F and J.8).

6 DISCUSSION

Soft execution retains candidates lost to hard commitment and improves their ranking through graded compatibility. CIDDS benefits from both mechanisms, while CatSup shows how their value decreases as support becomes identified. The calibration studies add a second consideration: confidence must be translated into a decision at the appropriate scale. JSON Schema varies the acceptance threshold, whereas BFCL varies the cost of choosing an unseen value. Posterior concentration, domain coverage, and score margins therefore suggest a future policy for choosing the execution rule (Appendix C.6).

The correlated-posterior study also separates acquisition quality from representation fidelity. A compact representation can reproduce its target posterior's decisions even when the assumed constraint family is misspecified. Evaluating that family and checking the approximation against exact decisions therefore address distinct sources of error.

7 RELATED WORK

Execution with supplied constraints. Grammars and parsers guide autoregressive generation (Scholak et al., 2021; Poesia et al., 2022; Geng et al., 2023; Beurer-Kellner et al., 2024; Dong et al., 2025). Diffusion executors enforce automata, context-free grammars, lookahead, or verification-based restrictions (Dang & Ermon, 2026; Mündler et al., 2026; Suresh et al., 2025; Zhang et al., 2026). These executors take a supplied constraint; TACTIC infers candidate compatibility from demonstrations.

Constraint acquisition. Constraint acquisition learns a model from examples or queries (Bessière et al., 2017; Raedt et al., 2018). Active methods resolve ambiguity through interaction (Angluin, 1988; Bessière et al., 2013); positive-only methods infer structure from valid solutions (Beldiceanu & Simonis, 2012; Kumar et al., 2022; Wiegand et al., 2024). Statistical and compact-model approaches improve selection or sample efficiency (Prestwich & Wilson, 2024; Bessiere et al., 2025; Tsouros et al., 2025). Version spaces instead retain multiple hypotheses consistent with the data (Mitchell, 1977; Bessière et al., 2005). Posterior membership averaging and the uniform-positive-sampling size principle are classical Bayesian concept-learning constructions (Tenenbaum & Griffiths, 2001). TACTIC builds on this uncertainty-aware view but studies its downstream use: whether to average, approximate, or hard-enforce the acquired constraint at decode time.

Structured uncertainty and generation. Tensor trains and matrix product states provide tractable representations of high-order dependence (Oseledets, 2011; Schollwöck, 2011; Cichocki et al., 2016). TACTIC applies them to a posterior over latent constraint components. This differs from CoDD, which couples simultaneously predicted output tokens inside a diffusion model (Li et al., 2026). Related applications in schema discovery and network generation are discussed in Appendix I.

8 LIMITATIONS AND CONCLUSION

Limitations. TACTIC depends on task-specific acquisition, candidate coverage, and confidence calibration. The appendix details the fixed-query CatSup design, calibration-set differences, and the single-draw BFCL recalibration. TT construction currently uses dense diagnostic tensors, leaving efficient acquisition at larger scales open.

Conclusion. TACTIC carries constraint uncertainty into decoding through compatibility and model evidence. Its support and stability analysis explains how plausible outputs can survive acquisition ambiguity; experiments demonstrate the resulting gains over hard execution under scarce support. Marginal-matched comparisons isolate the contribution of joint dependence, while agreement with exact-posterior decisions provides a concrete target for assessing tensor-train approximations.

AI USE STATEMENT

Generative AI tools assisted with code development, debugging, statistical and mathematical cross-checks, figure and table preparation, and manuscript drafting. The authors conducted the research ideation, manually inspected the relevant literature, verified generated code through tests and result reproduction, traced quantitative claims to the experiment registry, and checked mathematical statements and proofs. Generative AI tools did not determine acceptance criteria or replace author review. The authors take responsibility for the final text, claims, code, and artifacts.

ETHICS STATEMENT

This work studies reliability mechanisms for structured generation and uses public or synthetic benchmarks. It does not involve human subjects or private production traffic. Constraint induction can encode biases or errors present in demonstrations; our method exposes rather than removes that risk by retaining uncertainty and reporting open-world failures. Deployment in security-sensitive or high-stakes settings requires domain review, privacy controls, and abstention policies beyond the present experiments.

REPRODUCIBILITY STATEMENT

The appendix specifies acquisition models, execution rules, model settings, data splits, and statistical procedures. Proofs appear in Appendix A; detailed results and sensitivity analyses are in Appendices D to G and J. The supplement provides source code, configurations, and compact evaluation records; Appendix H summarizes the models, data, and reproduction instructions.

REFERENCES

Dana Angluin. Queries and concept learning. *Machine Learning*, 2:319–342, 1988. doi: 10.1007/BF00116828.

Lyes Attouche, Mohamed-Amine Baazizi, Dario Colazzo, Giorgio Ghelli, and Carlo Sartiani. Witness generation for json schema. *Proceedings of the VLDB Endowment*, 15(13):4002–4014, 2022. doi: 10.14778/3565838.3565852.

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems*, volume 34, pp. 17981–17993, 2021.

Mohamed-Amine Baazizi, Dario Colazzo, Giorgio Ghelli, and Carlo Sartiani. Parametric schema inference for massive json datasets. *The VLDB Journal*, 28(4):497–521, 2019. doi: 10.1007/s00778-018-0532-7.

Nicolas Beldiceanu and Helmut Simonis. A model seeker: Extracting global constraint models from positive examples. In *Principles and Practice of Constraint Programming*, volume 7514 of *Lecture Notes in Computer Science*, pp. 141–157, 2012. doi: 10.1007/978-3-642-33558-7_13.

Christian Bessière, Remi Coletta, Frédéric Koriche, and Barry O'Sullivan. A sat-based version space algorithm for acquiring constraint satisfaction problems. In *Machine Learning: ECML 2005*, volume 3720 of *Lecture Notes in Computer Science*, pp. 23–34, 2005. doi: 10.1007/11564096_8.

Christian Bessière, Remi Coletta, Emmanuel Hebrard, George Katsirelos, Nadjib Lazaar, Nina Narodytska, Claude-Guy Quimper, and Toby Walsh. Learning constraints through partial queries. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, pp. 475–481, 2013.

Christian Bessière, Frédéric Koriche, Nadjib Lazaar, and Barry O'Sullivan. Constraint acquisition. *Artificial Intelligence*, 244:315–342, 2017. doi: 10.1016/j.artint.2015.08.001.

Christian Bessiere, Clément Carbonnel, and Areski Himeur. Learning compact representations of constraint networks. In *European Conference on Artificial Intelligence*, pp. 885–893, 2025. doi: 10.3233/FAIA250893.

Luca Beurer-Kellner, Marc Fischer, and Martin Vechev. Guiding llms the right way: Fast, non-invasive constrained generation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 3658–3673, 2024.

Stefano Bistarelli, Ugo Montanari, and Francesca Rossi. Semiring-based constraint satisfaction and optimization. *Journal of the ACM*, 44(2):201–236, 1997. doi: 10.1145/256303.256306.

Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950. doi: 10.1175/1520-0493(1950)078⟨0001:VOFEIT⟩2.0.CO;2.

Andrzej Cichocki, Namgil Lee, Ivan V. Oseledets, Anh-Huy Phan, Qibin Zhao, and Danilo P. Mandic. Tensor networks for dimensionality reduction and large-scale optimization: Part 1 low-rank tensor decompositions. *Foundations and Trends in Machine Learning*, 9(4–5):249–429, 2016. doi: 10.1561/2200000059.

Meihua Dang and Stefano Ermon. Constrained decoding for diffusion language models via efficient inference over finite automata. *arXiv preprint arXiv:2607.07026*, 2026. doi: 10.48550/arXiv.2607.07026.

Yixin Dong, Charlie F. Ruan, Yaxing Cai, Ziyi Xu, Yilong Zhao, Ruihang Lai, and Tianqi Chen. XGrammar: Flexible and efficient structured generation engine for large language models. In *Proceedings of Machine Learning and Systems*, volume 7, 2025.

Saibo Geng, Martin Josifoski, Maxime Peyrard, and Robert West. Grammar-constrained decoding for structured nlp tasks without finetuning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 10932–10952, 2023. doi: 10.18653/v1/2023.emnlp-main.674.

E. Mark Gold. Language identification in the limit. *Information and Control*, 10(5):447–474, 1967. doi: 10.1016/S0019-9958(67)91165-5.

I. J. Good. The population frequencies of species and the estimation of population parameters. *Biometrika*, 40(3–4):237–264, 1953. doi: 10.1093/biomet/40.3-4.237.

Sumit Gulwani. Automating string processing in spreadsheets using input-output examples. In *Proceedings of the 38th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pp. 317–330, 2011. doi: 10.1145/1926385.1926423.

Zhao-Yu Han, Jun Wang, Heng Fan, Lei Wang, and Pan Zhang. Unsupervised generative modeling using matrix product states. *Physical Review X*, 8(3):031012, 2018. doi: 10.1103/PhysRevX.8.031012.

Hongyu Hè, Minhao Jin, and Maria Apostolaki. Making logic a first-class citizen in generative ML for networking. In *23rd USENIX Symposium on Networked Systems Design and Implementation*, pp. 801–824, 2026. doi: 10.48550/arXiv.2506.23964.

Jennifer A. Hoeting, David Madigan, Adrian E. Raftery, and Chris T. Volinsky. Bayesian model averaging: A tutorial. *Statistical Science*, 14(4):382–417, 1999. doi: 10.1214/ss/1009212519.

Mohit Kumar, Samuel Kolb, and Tias Guns. Learning constraint programming models from data using generate-and-aggregate. In *28th International Conference on Principles and Practice of Constraint Programming*, volume 235 of *Leibniz International Proceedings in Informatics*, pp. 29:1–29:16, 2022. doi: 10.4230/LIPIcs.CP.2022.29.

Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015. doi: 10.1126/science.aab3050.

Ian Li, Zilei Shao, Benjie Wang, Rose Yu, Guy Van den Broeck, and Anji Liu. Breaking the factorization barrier in diffusion language models. *arXiv preprint arXiv:2603.00045*, 2026. doi: 10.48550/arXiv.2603.00045.

Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B. Hashimoto. Diffusion-lm improves controllable text generation. In *Advances in Neural Information Processing Systems*, volume 35, pp. 4328–4343, 2022.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 32819–32848, 2024.

Michael J. Mior. JSONoid: Monoid-based enrichment for configurable and scalable data-driven schema discovery. *arXiv preprint arXiv:2307.03113*, 2023. doi: 10.48550/arXiv.2307.03113.

Tom M. Mitchell. Version spaces: A candidate elimination approach to rule learning. In *Proceedings of the Fifth International Joint Conference on Artificial Intelligence*, pp. 305–310, 1977.

M. G. M. Moreno and Fernando Parisio. All bipartitions of arbitrary Dicke states. *arXiv preprint arXiv:1801.00762*, 2018. URL `https://arxiv.org/abs/1801.00762`.

Niels Mündler, Jasper Dekoninck, and Martin Vechev. Constrained decoding of diffusion LLMs with context-free grammars. In *International Conference on Learning Representations*, 2026. doi: 10.48550/arXiv.2508.10111.

Ivan V. Oseledets. Tensor-train decomposition. *SIAM Journal on Scientific Computing*, 33(5): 2295–2317, 2011. doi: 10.1137/090752286.

Barry O'Sullivan. Automated modelling and solving in constraint programming. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 1493–1497, 2010.

Amichai Painsky. Convergence guarantees for the good–turing estimator. *Journal of Machine Learning Research*, 23(279):1–37, 2022.

Kanghee Park, Timothy Zhou, and Loris D'Antoni. Flexible and efficient grammar-constrained decoding. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 48262–48275, 2025.

Shishir G. Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E. Gonzalez. The berkeley function calling leaderboard (BFCL): From tool use to agentic evaluation of large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 48371–48392, 2025.

Gabriel Poesia, Oleksandr Polozov, Vu Le, Ashish Tiwari, Gustavo Soares, Christopher Meek, and Sumit Gulwani. Synchromesh: Reliable code generation from pre-trained language models. In *International Conference on Learning Representations*, 2022.

Steven Prestwich and Nic Wilson. A statistical approach to learning constraints. *International Journal of Approximate Reasoning*, 171:109184, 2024. doi: 10.1016/j.ijar.2024.109184.

Luc De Raedt, Andrea Passerini, and Stefano Teso. Learning constraints from examples. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, pp. 7965–7970, 2018. doi: 10.1609/aaai.v32i1.12217.

Markus Ring, Sarah Wunderlich, Deniz Scheuring, Dieter Landes, and Andreas Hotho. Flow-based benchmark data sets for intrusion detection. In *Proceedings of the 16th European Conference on Cyber Warfare and Security*, pp. 361–369, 2017.

Jorma Rissanen. Modeling by shortest data description. *Automatica*, 14(5):465–471, 1978. doi: 10.1016/0005-1098(78)90005-5.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T. Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 130136–130184, 2024.

Torsten Scholak, Nathan Schucher, and Dzmitry Bahdanau. PICARD: Parsing incrementally for constrained auto-regressive decoding from language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 9895–9901, 2021. doi: 10.18653/v1/2021.emnlp-main.779.

Ulrich Schollwöck. The density-matrix renormalization group in the age of matrix product states. *Annals of Physics*, 326(1):96–192, 2011. doi: 10.1016/j.aop.2010.09.012.

Lorraine Schwartz. On bayes procedures. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 4:10–26, 1965. doi: 10.1007/BF00535479.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis K. Titsias. Simplified and generalized masked diffusion for discrete data. In *Advances in Neural Information Processing Systems*, volume 37, pp. 103131–103167, 2024.

William Spoth, Oliver Kennedy, Ying Lu, Beda Christoph Hammerschmidt, and Zhen Hua Liu. Reducing ambiguity in json schema discovery. In *Proceedings of the 2021 International Conference on Management of Data*, pp. 1732–1744, 2021. doi: 10.1145/3448016.3452801.

Tarun Suresh, Debangshu Banerjee, Shubham Ugare, Sasa Misailovic, and Gagandeep Singh. DINGO: Constrained inference for diffusion LLMs. In *Advances in Neural Information Processing Systems*, 2025. doi: 10.48550/arXiv.2505.23061.

Joshua B. Tenenbaum and Thomas L. Griffiths. Generalization, similarity, and Bayesian inference. *Behavioral and Brain Sciences*, 24(4):629–640, 2001. URL `https://cocosci.princeton.edu/tom/papers/TGbbs.pdf`.

Dimos Tsouros, Senne Berden, Steven Prestwich, and Tias Guns. Generalizing constraint models in constraint acquisition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 11362–11371, 2025. doi: 10.1609/aaai.v39i11.33236.

Shubham Ugare, Tarun Suresh, Hangoo Kang, Sasa Misailovic, and Gagandeep Singh. SynCode: Llm generation with grammar augmentation. *Transactions on Machine Learning Research*, 2025.

Boris Wiegand, Dietrich Klakow, and Jilles Vreeken. What are the rules? discovering constraints from data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 2024. doi: 10.1609/AAAI.V38I8.28658.

Brandon T. Willard and Rémi Louf. Efficient guided generation for large language models. *arXiv preprint arXiv:2307.09702*, 2023. doi: 10.48550/arXiv.2307.09702.

Yitong Zhang, Yongmin Li, Yuetong Liu, Jia Li, Xiaoran Jia, Zherui Li, and Ge Li. Lookahead-then-verify: Reliable constrained decoding for diffusion LLMs under context-free grammars. *arXiv preprint arXiv:2602.00612*, 2026. doi: 10.48550/arXiv.2602.00612.

A Proofs and Scoring Guarantees

A.1 Proof of Theorem 1

Proof. Write p for the probability that the policy chooses C_1 after the fixed observations. If the truth is C_1, it excludes the valid witness x_1 whenever it chooses C_2, with probability $1 - p$. If the truth is C_2, it excludes x_2 with probability p. Thus the larger of the two exclusion probabilities is

$$\max\{p, 1 - p\} \geq \tfrac{1}{2}.$$

For a deterministic policy, $p \in \{0, 1\}$ and the maximum is one. Both ground truths remain possible because each assigns positive likelihood to the same observed demonstrations.

For the evidence claim, assume the commitment law does not depend on w_y. Choose the ground truth attaining the larger exclusion probability and let x_j be its witness. For any $0 < \eta < 1$, assign normalized evidence mass $1 - \eta$ to x_j and distribute the remaining mass among the other candidates. This changes neither p nor the chosen ground truth. With probability at least $1/2$ (one for a deterministic policy), the committed support removes the valid witness carrying mass $1 - \eta$. □

Scope of the lower bound. The bound concerns the two actions C_1, C_2 after observing the data. A union of their supports is a different action. Evidence independence is needed only for the high-mass claim: with singleton query supports, an evidence-aware policy can retain the higher-mass witness, limiting the excluded witness to at most half the evidence mass.

A.2 Proof of Proposition 2

Proof. Since $A_C(x) = \mathbf{1}[C \in \mathcal{H}_x]$, its posterior expectation is $\rho_n(\mathcal{H}_x)$. Positive evidence and compatibility give x a positive unnormalized score in equation (10). The finite candidate set and finite scores make the normalizer finite and positive, so normalization preserves this support. □

A.3 Proof of Theorem 3

Proof. It suffices to show that the posterior odds of each positive-prior alternative vanish. For such a $C \neq C^\star$, let p_C denote its demonstration density with respect to a common dominating measure, and set $L_i(C) = \log[p_C(X_i)/p_{C^\star}(X_i)]$. Finite KL divergence makes $L_i(C)$ integrable: $\log t \leq t$ gives $\mathbb{E}_{C^\star}[(L_i(C))_+] \leq 1$, and finite KL then bounds the negative part. Bayes' rule and the strong law give

$$\frac{1}{n}\log\frac{\rho_n(C)}{\rho_n(C^\star)} = \frac{1}{n}\log\frac{\pi(C)}{\pi(C^\star)} + \frac{1}{n}\sum_{i=1}^{n} L_i(C) \longrightarrow -D_{\mathrm{KL}}(P^+_{C^\star} \| P^+_C) < 0 \quad \text{a.s.}$$

Thus every such alternative has vanishing odds. Zero-prior hypotheses remain at zero posterior, and finiteness of $\mathcal{H}$ yields

$$\rho_n(C^\star) = \left(1 + \sum_{C \neq C^\star} \frac{\rho_n(C)}{\rho_n(C^\star)}\right)^{-1} \longrightarrow 1.$$

For compatibility, subtract the true acceptance function inside the posterior average:

$$\Psi^{\mathrm{PP}}_{\rho_n}(x) - A_{C^\star}(x) = \sum_{C \neq C^\star} \rho_n(C)\big(A_C(x) - A_{C^\star}(x)\big).$$

Each difference has absolute value at most one, giving the uniform bound $1 - \rho_n(C^\star)$. □

Decisions with fixed score weights. Fix a finite candidate set, finite nonnegative evidence, and $\lambda_w, \lambda_C > 0$. Let

$$u_n(x) = w_y(x)^{\lambda_w}\, \Psi^{\mathrm{PP}}_{\rho_n}(x)^{\lambda_C}, \qquad u_\star(x) = w_y(x)^{\lambda_w} A_{C^\star}(x).$$

The theorem and continuity of $t \mapsto t^{\lambda_C}$ give $u_n \to u_\star$ uniformly on this finite set. If $u_\star$ has a unique winner $x^\star$ with positive score, let $\Delta = \min_{x \neq x^\star}[u_\star(x^\star) - u_\star(x)] > 0$. Eventually $\max_x |u_n(x) - u_\star(x)| < \Delta/2$, which preserves every winner–competitor comparison. Hence the selected candidate eventually agrees with supplied-constraint execution.

Identifiability and zero likelihoods. On a finite demonstration space, the consistency conclusion extends to every alternative with a distinct demonstration distribution. An alternative either has positive finite KL or assigns zero probability to an event of true probability $b_C > 0$. In the latter case, it survives n iid observations with probability $(1 - b_C)^n$ and is eliminated in finite time almost surely. Hypotheses with identical demonstration distributions retain their prior odds.

A.4 Proof of Theorem 4

Proof. Fix a cut i. In the unfolding matrix, a row with left Hamming weight j is the indicator of right assignments with weight $K - j$. It is nonzero exactly for

$$J_i = \{j : \max(0, K - (m - i)) \leq j \leq \min(i, K)\}.$$

Rows with the same weight coincide. Rows with distinct feasible weights have disjoint nonempty supports and are linearly independent. Hence the unfolding rank is

$$|J_i| = \min(i, K) - \max(0, K - (m - i)) + 1 = \min(i, K, m - i, m - K) + 1.$$

Any TT with bond dimension b_i factors this unfolding into a matrix with b_i columns and one with b_i rows, so $b_i \geq |J_i|$.

To attain all these bounds at once, use the partial count as the bond state. Set $J_0 = \{0\}$, $J_m = \{K\}$, and define cores

$$G_r[z_r]_{u,v} = \mathbf{1}[v = u + z_r], \qquad u \in J_{r-1},\ v \in J_r.$$

A binary assignment has exactly one path through these cores when its total is K, and none otherwise. Their product is therefore $T_{m,K}(z)$, with bond dimensions $|J_i|$. This also supplies a nonnegative exact representation.

If $1 \leq K \leq m/2$, cut $i = K$ attains $K + 1$, the largest possible value. Complementing every bit handles $K > m/2$. For $K = 0$ or m, every bond has dimension one. Thus the minimum maximum bond dimension is $\min(K, m - K) + 1$. Dividing by $\binom{m}{K}$ normalizes the tensor without changing its ranks. □

Dependence on the weights. Uniformity is essential to the exact rank formula. For $m = 4, K = 2$, give assignment 1010 weight two and each of the other five weight-two assignments weight one, then normalize by seven. The middle unfolding has two nonzero singleton blocks and a full-rank 2×2 block, so its rank is four rather than the uniform tensor's three, despite identical support.

A.5 Proof of Proposition 5

Proof. Let $\eta = \|\log \widehat{\Psi} - \log \Psi\|_{\infty,\mathcal{F}}$ and $\delta(x) = \log \widehat{\Psi}(x) - \log \Psi(x)$. For every competitor $x \neq x^\star$,

$$\widehat{s}(x^\star) - \widehat{s}(x) = s_\Psi(x^\star) - s_\Psi(x) + \lambda_C[\delta(x^\star) - \delta(x)] \geq \gamma - 2\lambda_C\eta > 0.$$

Thus all comparisons favoring $x^\star$ are preserved.

For the TV statement, use $\|\rho - \widehat{\rho}\|_{\mathrm{TV}} = \sup_B |\rho(B) - \widehat{\rho}(B)|$ and the accepting set $\mathcal{H}_x$. Then

$$|\Psi_\rho(x) - \Psi_{\widehat{\rho}}(x)| = |\rho(\mathcal{H}_x) - \widehat{\rho}(\mathcal{H}_x)| \leq \|\rho - \widehat{\rho}\|_{\mathrm{TV}}.$$

Both compatibilities lie in $[\epsilon, 1]$, where the derivative of $\log$ is bounded by $1/\epsilon$. Consequently $\eta \leq \|\rho - \widehat{\rho}\|_{\mathrm{TV}}/\epsilon$, and the stated condition makes $\gamma - 2\lambda_C\eta$ positive. □

A one-sided compatibility bound. If only $\Psi_\rho \geq \epsilon$ is known, put $t = \|\rho - \widehat{\rho}\|_{\mathrm{TV}}$. For $t < \epsilon$,

$$1 - \frac{t}{\epsilon} \leq \frac{\Psi_{\widehat{\rho}}(x)}{\Psi_\rho(x)} \leq 1 + \frac{t}{\epsilon}, \qquad \eta \leq -\log(1 - t/\epsilon).$$

Thus $2\lambda_C[-\log(1 - t/\epsilon)] < \gamma$ is a sufficient one-sided condition.

A.6 Stability under per-query min–max scaling

For a nonconstant vector v on $\mathcal{F}$, write $N(v)(x) = (v(x) - \min_{\mathcal{F}} v)/(\max_{\mathcal{F}} v - \min_{\mathcal{F}} v)$. For a constant vector set $N(v) = 1/2$, as in the strict log min–max executor of Appendix C. If both log channels have positive ranges R_w, R_C, then the calibrated score is, up to an additive constant,

$$\frac{\alpha}{R_w} \log w_y(x) + \frac{1 - \alpha}{R_C} \log \Psi(x).$$

This identifies effective weights for one fixed pair of score vectors. Replacing Ψ and recomputing its range changes the second weight, so Proposition 5 alone does not cover the full operation.

Proposition 6 (Stability with recomputed min–max scales)**.** *Use the same finite active candidate set $\mathcal{F}$ for the exact and approximate scores, with positive $w_y, \Psi, \widehat{\Psi}$ there. Fix $0 \leq \alpha < 1$ and let $x^\star$ uniquely maximize the calibrated score with margin $\gamma_{\mathrm{cal}} > 0$. Set*

$$v = \log \Psi, \quad \widehat{v} = \log \widehat{\Psi}, \quad R_C = \max_{\mathcal{F}} v - \min_{\mathcal{F}} v > 0, \quad \eta = \|\widehat{v} - v\|_\infty.$$

If $2\eta < R_C$ and $4(1 - \alpha)\eta < R_C\gamma_{\mathrm{cal}}$, recomputing the min–max scale for $\widehat{v}$ preserves $x^\star$.

Proof. Let $\widehat{R}_C$ be the range of $\widehat{v}$. The minimum and maximum each change by at most η, so $|\widehat{R}_C - R_C| \le 2\eta$ and $\widehat{R}_C > 0$. For a competitor x, define $d = v(x^\star) - v(x)$ and $\widehat{d} = \widehat{v}(x^\star) - \widehat{v}(x)$. Subtracting the channel minimum cancels in a pairwise comparison. Since $|\widehat{d} - d| \le 2\eta$ and $|\widehat{d}| \le \widehat{R}_C$,

$$\left| \frac{\widehat{d}}{\widehat{R}_C} - \frac{d}{R_C} \right| \le \frac{|\widehat{d} - d|}{R_C} + \frac{|\widehat{d}|\,|R_C - \widehat{R}_C|}{R_C \widehat{R}_C} \le \frac{4\eta}{R_C}.$$

The evidence channel is unchanged. Each calibrated winner margin therefore remains at least $\gamma_{\mathrm{cal}} - 4(1-\alpha)\eta/R_C > 0$. □

The same TV-to-log bounds proved above can be substituted for η, provided the active support is unchanged. At $\alpha = 1$ the compatibility channel is unused, so changing it cannot affect the result.

Vanishing compatibility range. Consider two candidates accepted by $C^\star$, with $w_y(x_1) > w_y(x_2) > 0$. Let the posterior put mass $1 - \varepsilon_n$ on $C^\star$ and mass $\varepsilon_n > 0$ on an alternative accepting only x_2, where $\varepsilon_n \to 0$. Then $\Psi_n(x_1) = 1 - \varepsilon_n$ and $\Psi_n(x_2) = 1$: the posterior concentrates and both compatibilities converge to one. Nevertheless, min–max maps their log scores to $(0, 1)$ at every finite n, while the evidence channel is $(1, 0)$. For $0 < \alpha < 1/2$, the calibrated decoder always chooses x_2. Under the supplied constraint the compatibility channel is constant, so it chooses x_1. Posterior consistency alone therefore does not ensure convergence of this calibrated decoder. The range condition above states where a perturbation argument is valid.

A.7 Finite-sample affine-hull recovery

This complement concerns an affine-hull estimator; no benchmark in this paper implements it. The following calculation concerns recovery by a specific estimator. It complements posterior consistency with an exact finite-sample probability.

Lemma 7 (Affine-hull recovery)**.** *Let q be a prime power, $0 \le d \le M$, and $n \ge 1$. If $X_1, \dots, X_n$ are iid uniform on an affine subspace $S = a + U \subseteq \mathbb{F}_q^M$ of dimension d, then*

$$\mathbb{P}(\mathrm{aff}\{X_1, \dots, X_n\} = S) = \begin{cases} \prod_{j=0}^{d-1} (1 - q^{j-(n-1)}), & n - 1 \ge d, \\ 0, & n - 1 < d. \end{cases}$$

The empty product is one.

Proof. Condition on X_1. Translation by $-X_1$ maps S bijectively onto U, so the differences $X_2 - X_1, \dots, X_n - X_1$ are conditionally iid uniform on U. Their distribution does not depend on the value of X_1. The sample affine hull is therefore S exactly when these differences span U.

Choose a basis of U and place the differences in a $d \times (n-1)$ matrix. Each such matrix has probability $q^{-d(n-1)}$. If $n - 1 < d$, its rows cannot be independent. Otherwise, choose the rows successively: after j independent rows, their span contains q^j vectors, leaving $q^{n-1} - q^j$ choices for the next row. Dividing the product of these counts by the total number of matrices gives

$$\frac{\prod_{j=0}^{d-1} (q^{n-1} - q^j)}{q^{d(n-1)}} = \prod_{j=0}^{d-1} (1 - q^{j-(n-1)}).$$

For $d = 0$, one observation already recovers the singleton S. □

If the hypothesis family allows larger affine supersets, they can retain positive likelihood even after the sample affine hull equals S. The lemma establishes recovery of the estimator, rather than uniqueness of every data-compatible hypothesis.

Table 4: Benchmark interfaces. PP denotes posterior-predictive compatibility and WM the confidence-weighted mean in equation (8). Headline backbones correspond to Table 1; cross-backbone results are in Appendix D.1.

Benchmark	Latent object	Compatibility	Candidate execution
CatSup	categorical support	PP, working model	full enumeration
GraphColor	edge/graph family	PP, factorized or TT	full enumeration
CIDDS	weighted implications	WM, non-posterior	documented record proposals
JSONSchemaBench	hidden domains/branches	PP + missing mass	valid/invalid instances
BFCL	enum domains	PP + open world	weighted-DFA diffusion

B Benchmark-to-Model Mappings

The mappings share bounded candidate compatibility; their selection or classification rules are specified per benchmark. Model assignments are as follows: CatSup and GraphColor use LLaDA-8B as the headline backbone with DREAM-7B for replication; CIDDS uses DREAM-7B as the headline backbone with LLaDA-8B for replication; BFCL uses DREAM-7B; and JSONSchemaBench uses no language model (its primary classifier evaluates compatibility directly on valid/invalid instances). Priors, likelihoods, atomic states, proposal mechanisms, and missing-mass models remain family-specific; CIDDS, for example, uses neither an independent-rule product nor a TT posterior. The separate correlated-posterior decision study uses LLaDA-8B evidence; its analytic ranks and posterior-fidelity measures do not depend on a language model.

C Protocols and Implementation Details

C.1 Shared protocol

Candidate-rescoring comparisons freeze the model checkpoint, candidate set, and model-evidence cache within an evaluation unit. BFCL instead holds the model, query pool, and chain-diffusion executor fixed while varying domain enforcement; its generation trajectories can differ. JSON Schema compares classifiers on the same evaluation instances without model evidence. Tuned score scales, hard thresholds, and α are selected on development data and held fixed. CatSup uses fixed constants without a mixing grid; BFCL uses a selection-disjoint recalibration, with its original pilot-based curve retained for comparison. Missing evidence keys and vocabulary mismatches raise explicit errors. Unless stated otherwise, accuracy is exact structured completion, not token accuracy or validity.

Composition and the strict log-score executor. At unit weights, equation (10) maximizes expected utility $\mathbb{E}_C[w_y(x)A_C(x)]$. A different objective first normalizes within each constraint, giving $\sum_C \rho_n(C) w_y(x) A_C(x)/Z_C$, where $Z_C = \sum_x w_y(x)A_C(x) > 0$. The product rule uses the former objective. Its normalizing constant must be positive. Channel exponents and scaling are fixed execution parameters.

The strict log min–max instance of Algorithm 1 selects

$$x^\star = \arg\max_{x \in \mathcal{F}_y} \left[\alpha \bar{\ell}_w(x) + (1-\alpha)\bar{\ell}_C(x)\right], \tag{11}$$

where $\mathcal{F}_y$ excludes zeros in positively weighted channels, $\ell_w = \log w_y$, $\ell_C = \log c$, and bars denote min–max normalization over $\mathcal{F}_y$. Constant channels map to $1/2$. The protocol uses weights $(\lambda_w, \lambda_C) = (\alpha, 1-\alpha)$ and returns $\perp$ when $\mathcal{F}_y$ is empty. For each benchmark below, Eligible implements its query mask and zero policy; Scale applies its transformations over the stated domain before restricting scores to eligible candidates.

Score scaling and selection. Write $N(v) = (v - \min v)/(\max v - \min v)$ for per-query min–max scaling. CIDDS scores each completion by $\alpha N(L_w) + (1-\alpha)N(L_C)$, where L_w sums cached field log-probability scores and $L_C = \log\max(\Psi, 10^{-9})$. Both channels are normalized over the completion grid; a constant channel maps to $1/2$. Development selects α from $\{0, 0.3, 0.5, 0.7\}$. GraphColor instead uses $\alpha \log\max(N(w), 10^{-12}) + (1-\alpha)\log\max(N(\Psi), 10^{-12})$, with probability-scale normalization over all 3^n colorings, constant channels mapped to 1, and the query hint mask applied

afterward. Its development grid is $\{0, 0.25, 0.5, 0.75, 1.0\}$. The selected weight is fixed across test cells. These are separate instantiations of Algorithm 1, with different zero, scale, and candidate-domain conventions.

For the strict log min–max rule in equation (11), a positive-weight channel excludes exact zeros before the logarithm and range are computed. A zero-weight channel is absent from both filtering and scoring. This defines the pure-channel endpoints without evaluating $0 \log 0$. Empty active sets return abstention, singleton sets return their sole candidate, and score ties use the supplied deterministic candidate order. These conventions make the candidate-selection rule explicit; a protocol using a fallback reports it separately.

CatSup's recorded composed arm instead adds its cached `evidence_score` to $\log \Psi_D$ without the min–max mixing grid (Appendix D). JSON Schema uses compatibility for thresholded validity classification. BFCL uses the model-coupled weighted-DFA chain-diffusion executor described in Appendices C.5 and J.1. These two executors use the same acquisition and compatibility modules.

Compatibility evaluation. For the factorized posterior, evaluate each local acceptance probability and multiply them as in equation (5). For a normalized nonnegative TT posterior, initialize a row vector $h = 1$ and successively update $h \leftarrow hM_r(x)$ for $r = 1, \ldots, m$; the final scalar is equation (7). An enumerated joint posterior instead gives $\sum_C \rho_n(C) A_C(x)$. The weighted-rule path uses equation (8), including the value one when no positive-weight rule fires. These are the four backends referred to in Algorithm 1.

Main-table references. Table 1 separates the matched hard–soft comparison from Open and supplied-constraint execution. The supplied-constraint column is an execution reference, not an upper bound on exact accuracy. CIDDS reports the 24-cell macro over four support sizes and both split types, averaged over 20 independent support draws. JSON Schema is a validity-classification task, so Open and supplied-constraint generation entries are left blank. The JSON Schema row uses the 771-schema both-class cohort at the rebuilt-development threshold $\tau = .2$, with an unadjusted paired schema-bootstrap interval (Appendix J.6). The BFCL row uses the disjoint-pilot weight $p_0 = .85$ on the original 550 queries at support draw zero; all its reference arms use that same draw (Appendix J.7).

Concrete priors, likelihoods, hard readouts, score conventions, and BFCL decoding steps are in Appendix J.1. The main table reports GraphColor at $n = 12, N_s = 2$ and BFCL at $N_s = 8$; full size and support curves are in Appendices E and G.

C.2 CatSup

Each episode instantiates one of three function-call templates (search, create, config) with 4–5 categorical argument fields whose domains contain 2–3 values. The hidden categorical support takes the form of condition-specific allowed-value sets over argument fields, induced by a compound hidden rule with mutual-exclusion, conditional-dependency, and branch-coupling clauses; the rule partitions the template's full field-product universe (36–162 calls) into legal and illegal calls. Demonstrations are distinct calls sampled uniformly without replacement from the legal set, the query is a natural-language request consistent with one true legal call, and the candidate set is the entire universe, so exact completion requires selecting that call.

The 120-episode experiment uses one fixed hidden rule and two fixed legal queries per template. At requested budget N_s, each episode samples $|D^+| = \min\{N_s, |\mathcal{X}^+|\}$ distinct legal calls uniformly. Queries can occur in D^+, so this is a fixed-query, transductive design. The posterior uses a Dirichlet–multinomial working likelihood, whereas support is sampled without replacement; the iid asymptotic analysis in Section 4 therefore does not directly describe this finite-population experiment. Matched arms share support, queries, candidates, and cached model evidence. Intervals use 5,000 episode-clustered bootstrap resamples and are conditional on the three templates and their fixed queries.

C.3 GraphColor

Each family is a random graph G on n vertices with C=3 colors; the hidden constraint is the edge set of G (adjacent vertices must receive different colors). Edges are added only between differently

colored vertices of a random proper coloring; n=8 draws $|E| \in [10, 14]$, and n>8 draws the edge count from $[\lceil 0.50|A| \rceil, \lceil 0.75|A| \rceil]$ over different-color pairs A to reproduce the realized n=8 density band (the density-matched rule). The candidate set enumerates all 3^n colorings, demonstrations are proper colorings sampled uniformly with replacement from this legal set, and each query fixes $\lceil n/2 \rceil$ vertices of a true legal coloring (hint fraction .5), scoring free vertices with frozen LLaDA-8B color marginals. Exact success requires recovering the complete target coloring; validity requires emitting any proper coloring.

The protocol uses LLaDA-8B, density-matched graph families, hint fraction .5, and exact candidate enumeration for $n \in \{8, 10, 12\}$. Development families `g_s0`–`g_s29` select α from the prespecified grid; test families `g_s30`–`g_s59` are evaluated once with two queries per family. Scores use the GraphColor probability-scale rule specified above. The evidence parser normalizes named colors and rejects missing keys instead of falling back to uniform values. Family-clustered bootstrap intervals use 5,000 resamples.

C.4 CIDDS

The task masks P/F, D/P, or S/D field pairs. Support sizes are $N_s \in \{4, 8, 16, 32\}$ across random and temporal splits. Ordered value implications use the weighted compatibility in equation (8); the rule-retention threshold is .05, with at least one consequent-confirming support record. Canonical development selects $\alpha = .3$ once for the full declared scope. Tuned-hard thresholds use a prespecified grid, selection objective, and deterministic tie-break on development data only. Model evidence is cached and support-draw independent.

The primary statistic has two levels. Within support draw k, it is the unweighted average of 24 mask-by-split-by-N_s cells. Across $K = 20$ independent draws, the point estimate is the mean of those 20 macros. Headline and cross-backbone confidence intervals are paired t intervals on the 20 macro differences. The central percentile range across draws describes acquisition variability; it is not a confidence interval for the mean. Inner paired record-cluster intervals condition on the realized support draws.

C.5 JSONSchemaBench and BFCL

JSONSchemaBench uses $N_s \in \{4, 8, 16, 32, 64\}$ and ten support draws, with the original frozen threshold $\tau = .05$ and a rebuilt-development threshold $\tau = .2$ (Appendix J.6). Construction attempts 2,349 non-development schemas; 2,342 resolve and seven time out. The supplied construction drivers apply a 24-hour limit to each schema's five-draw block. The reference validator checks the nominal invalid tests before evaluation. Balanced accuracy is reported on 771 schemas retaining both classes in every draw; Appendix G.1.1 documents this cohort selection, missing classes, and sensitivity to incomplete draws.

Paired intervals average draws within schema and bootstrap schemas with 10,000 replicates. They are descriptive, unadjusted 95% intervals for the revised cohort. AUROC handles ties by average rank, and Brier/ECE include every record-bearing cell in their stated cohort. Full-census timeout bounds apply only to scores defined on the full census (Appendix G.1.2).

BFCL-HiddenEnum uses DREAM-7B with the weighted-DFA chain-diffusion executor and official AST checker (`bfcl_eval` 2025.12.17) on a frozen 550-query pool from BFCL v4 live simple and multiple. The original $p_0 = .3$ was chosen on a 50-query pilot within that pool. Its three-draw support curve is retained in Table 14. The additional calibration uses 45 queries outside the pool, obtained by the predeclared candidate-argument cap change from six to eight. Official pilot accuracy selects $p_0 = .85$; Appendix J.7 details pilot selection and the paired, single-draw evaluation. Seen and unseen strata refer to whether the target enum value appears in acquired support.

C.6 Choosing the execution rule

The reported benchmarks compare fixed execution rules. Their support trends suggest an adaptive extension based on three observable signals: posterior concentration or rule confidence, estimated coverage of the legal domain, and the composed-score margin between leading candidates. Low concentration, missing mass, or a narrow margin favors preserving uncertainty. Once the acquired

Table 5: CatSup across backbones. Exact accuracy is in percent. Δ_H and Δ_C compare TACTIC with hard and compatibility-only execution; intervals use paired episode-clustered bootstrapping.

Backbone	N_s	Hard	Compat.	TACTIC	Δ_H [CI] (pp)	Δ_C [CI] (pp)
LLaDA-8B	4	14.58	7.08	43.33	+28.8 [22.5, 35.0]	+36.3 [29.2, 43.3]
	8	23.33	13.75	45.00	+21.7 [15.8, 27.9]	+31.3 [23.8, 38.8]
	16	40.83	22.50	49.58	+8.8 [5.0, 12.9]	+27.1 [18.8, 35.4]
	32	50.00	22.50	50.00	0.0 [0.0, 0.0]	+27.5 [20.8, 33.8]
DREAM-7B	4	14.58	7.08	47.92	+33.3 [26.3, 40.8]	+40.8 [33.3, 48.3]
	8	22.50	13.75	45.42	+22.9 [17.5, 28.8]	+31.7 [24.2, 39.2]
	16	39.58	22.50	48.75	+9.2 [5.8, 12.5]	+26.3 [17.9, 34.6]
	32	50.00	22.50	50.00	0.0 [0.0, 0.0]	+27.5 [20.8, 33.8]

constraint stabilizes and candidate coverage is adequate, hard compilation may offer a simpler competitive executor. This would require a development-set policy for switching between rules and an evaluation of its own; the current results do not evaluate such a switching policy. Proposal recall remains separate from the choice between soft and hard constraint execution.

C.7 Backbones and cross-backbone replication

The crossed design evaluates CatSup, CIDDS, and GraphColor with both LLaDA-8B-Instruct and DREAM-7B-Instruct. Episodes, demonstrations, acquisition results, candidate sets, test instances, splits, and development-selection rules are held fixed; only the backbone-specific evidence changes. These replications assess backbone sensitivity within each task and retain the task-specific execution rules of Appendices C and C.5.

Evidence extraction. LLaDA uses mask token 126336 and DREAM uses mask token 151666. CatSup and GraphColor use the respective chat templates; the LLaDA point-evidence path uses float32 and DREAM uses bfloat16. The loader implements LLaDA's two-dimensional key mask and DREAM's four-dimensional additive attention mask. CIDDS uses plain record serialization without a chat template for both backbones. LLaDA evidence extraction uses bfloat16, eager attention, bidirectional full-sequence forward passes, and canonical port encoding.

Frozen selection and evaluation. CatSup retains the original log-evidence plus log-compatibility composition, with no development mixing grid, and evaluates 120 episodes with two fixed, template-specific legal queries at requested support budgets $N_s \in \{4, 8, 16, 32\}$; the realized demonstration count is capped by the size of the template's legal set. Point estimates are observed accuracy differences; confidence intervals use 5,000 episode-clustered bootstrap resamples. CIDDS uses the same 24 cells and 20 support draws. On the canonical 1,000-record development set, LLaDA's macro scores for $\alpha \in \{0, .3, .5, .7\}$ are $.7141, .7411, .7376, .5334$, selecting .3, as for DREAM. Hard thresholds are selected per support size, seed, and mask pattern on development data, then frozen. Threshold selection uses random-pool proxy support also for temporal cells (no temporal test outcomes enter tuning). GraphColor pools $N_s \in \{2, 4, 8, 16, 32\}$ during development selection for each $n \in \{8, 10, 12\}$ and evaluates the same 30 held-out families with two queries per family; intervals resample families. Neither a backbone nor a mixing weight is selected using test performance.

D Full Results and Statistical Details

D.1 Cross-backbone replication

The crossed design holds acquisition and evaluation instances fixed while swapping LLaDA-8B and DREAM-7B evidence (Appendix C.7). Tables 5, 7 and 11 report CatSup, CIDDS, and GraphColor, respectively. The main table retains the original backbone for each benchmark.

CatSup. DREAM reproduces the shrinking soft–hard gap: observed gains are 33.3, 22.9, 9.2, and 0 points as N_s grows from 4 to 32. The compatibility-only predictions are identical across

backbones, with $158, 226, 230, 240$ of 240 queries tied at the respective support sizes. The evidence channel resolves these ties, yielding gains over compatibility-only even at $N_s = 32$, when hard and soft execution agree. Open and supplied-constraint accuracy remain 50% throughout. At $N_s = 4$, LLaDA's composed score is 6.7 points below Open (paired 95% interval $[-11.3, -2.9]$); DREAM reaches 47.92%. The advantage over hard execution transfers across backbones, while Open remains stronger in this fixed-query setting.

GraphColor. Development selects $\alpha^\star = (0, .25, 0)$ for DREAM at $n = (8, 10, 12)$, versus $(0, 0, .25)$ for LLaDA. At $N_s = 2$, DREAM improves over hard execution by 10.0, 3.3, and 8.3 points, respectively; only the $n = 8$ interval excludes zero. Evidence-only exact success is zero for both backbones at every size and support level. Thus the replication preserves the structural benefit while showing that the value of model evidence depends on graph size. DREAM's full curve and calibration details appear in Appendix E.

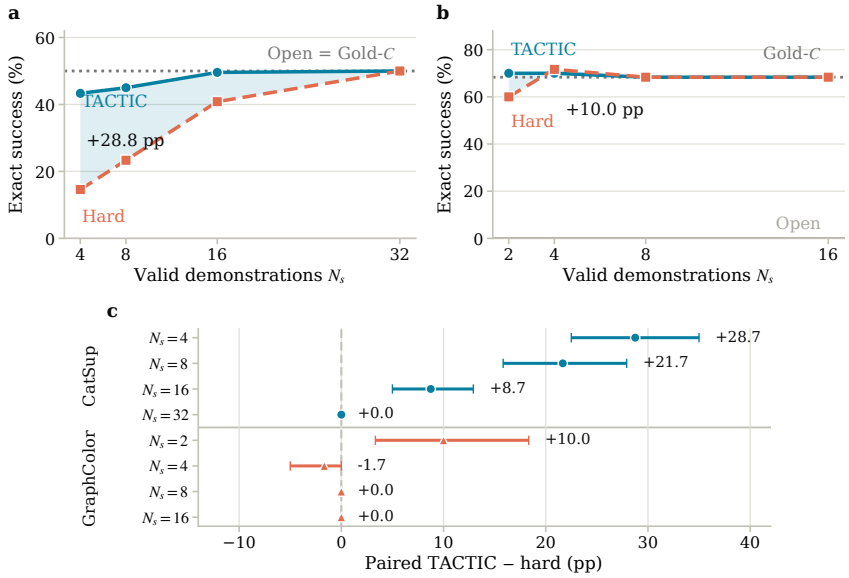


Figure 5: **Support reduces premature-commitment errors.** (a,b) CatSup and GraphColor ($n = 12$) completion accuracy with LLaDA-8B. Shading shows the soft–hard gap; gray lines mark Open and supplied-constraint references. (c) Paired gains with confidence intervals clustered by episode for CatSup and family for GraphColor.

D.2 Family-conditioned posterior results

The diagnostic uses $n = 8, N_s = 2$ and the log-additive score $\log w + \log \Psi$. The marginal-matched control in Appendix J.8 separates first-order changes from joint dependence. Fixed-K assumes four or eight edges while the generating graphs have 10–14 (Appendix J.3).

Table 6: Family-conditioned posterior comparison. TT approximates the conditioned model; the independent arm uses the unconditioned edge model. TV measures approximation error, and Δ reports the paired accuracy change in percentage points.

Family	TV χ=1/16	Indep.	TT χ=16	Δ [95% CI]
Fixed-K=4	$.492/<10^{-14}$	.325	.067	-25.8 $[-34.2, -17.5]$
Fixed-K=8	$.253/<10^{-14}$	.325	.225	-10.0 $[-16.7, -4.2]$
Degree sequence	.537/.0059	.281	.417	$+\mathbf{13.5}$ $[\mathbf{6.3}, \mathbf{21.9}]$
Connected	.053/.00054	.281	.281	0 [0, 0]

Table 7: CIDDS across backbones. Top: macro accuracy and ranges across common support draws, in percent. Bottom: paired gains with t confidence intervals, in percentage points.

Arm	DREAM-7B mean [range]	LLaDA-8B mean [range]
Open	12.0 [12.0, 12.0]	15.6 [15.6, 15.6]
Conditional count	28.8 [27.3, 30.2]	28.8 [27.3, 30.2]
Hard acquired	52.4 [48.2, 56.5]	53.5 [49.7, 56.9]
Dev-tuned hard	52.1 [45.1, 60.8]	52.1 [45.1, 60.8]
Compatibility-only ($\alpha = 0$)	69.7 [64.4, 74.5]	69.7 [64.4, 74.5]
TACTIC ($\alpha = .3$)	71.5 [65.4, 75.7]	72.8 [67.1, 76.7]
Supplied-rule reference	57.3 [57.3, 57.3]	57.2 [57.2, 57.2]
TACTIC minus	**DREAM-7B gain [CI]**	**LLaDA-8B gain [CI]**
Hard acquired	+19.03 [18.01, 20.05]	+19.31 [18.18, 20.44]
Dev-tuned hard	+19.41 [18.12, 20.70]	+20.77 [19.46, 22.08]
Compatibility-only	+1.77 [1.23, 2.31]	+3.15 [2.53, 3.76]
Open	+59.50 [58.30, 60.60]	+57.26 [56.21, 58.32]
Supplied-rule reference	+14.20 [13.00, 15.40]	+15.63 [14.58, 16.68]

D.3 CIDDS UNCERTAINTY HIERARCHY

The 20 support-draw macro differences have mean 19.41 points, outer standard deviation 2.76, range $[12.27, 25.16]$, central 95% range across draws $[13.52, 24.43]$, and a t confidence interval for the mean of $[18.12, 20.70]$. Every macro difference is positive. The pooled inner record-cluster interval is $[19.17, 19.67]$ and conditions on the 20 realized acquisitions. Individual draws also contain cell-level reversals.

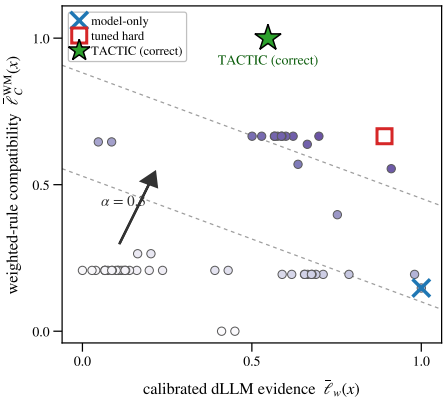


Figure 6: **CIDDS decision geometry for a held-out record.** Axes show calibrated evidence and rule compatibility; color shows their composed score. Markers identify model-only, tuned-hard, and TACTIC selections.

D.4 CIDDS FULL-ARM FACTORIAL

For DREAM, the composition contrast (TACTIC minus compatibility-only) is positive on 19 of 20 draws (two-sided sign-test $p = 4.0 \times 10^{-5}$); per-cell, it is positive or tied on most cells and reverts on a few saturated N_s=32 cells. For LLaDA, both the tuned-hard and compatibility-only contrasts are positive on all 20 draws. Compatibility-only and conditional-count predictions are unchanged by the evidence swap. The DREAM gain over conditional count is 42.6 points with a 95% interval of $[41.6, 43.7]$. Coverage is 100% (the candidate set is the full masked-field product). Tuned-hard abstains (no dev-selected-threshold proposal) on up to 49% of records at small N_s, and plain hard falls back to evidence decoding on up to 46% (no completion satisfies the acquired rule set); the composed TACTIC score never abstains.

The generative ICL arm completes all 20 draws on identical support and test records (Appendix J.2).

CIDDS TEMPORAL CELLS

Table 8: CIDDS temporal completion with DREAM-7B. Entries average accuracy across support draws for the prespecified mask and support conditions.

Arm	P/F N_s=8	D/P N_s=4	S/D N_s=8
Base / fully open	33.4	0.8	0.0
Conditional count	85.8	0.0	0.0
Hard acquired	83.1	39.5	18.5
Dev-tuned hard	79.6	27.8	22.9
Compatibility-only (α=0)	90.5	57.3	46.7
TACTIC (α=.3)	90.8	57.5	50.3
Supplied-rule reference	99.9	57.8	14.4

D.5 COMPONENT ABLATION (COMPATIBILITY-ONLY, α=0)

Table 9: Component ablation within each benchmark. Protocol-specific scoring and tie handling are described below.

Arm	CatSup	GC n=8	GC n=10	GC n=12	BFCL
Model-only / fully open	50.0	0.0	0.0	0.0	53.5
Hard point commitment	14.6	30.0	51.7	60.0	47.9
Compatibility-only (α=0)	7.1†	38.3	58.3	68.3	N/A‡
With evidence composition	43.3†	38.3§	58.3§	70.0§	52.6

† CatSup uses the fixed score $\text{evidence_score}(x) + \log \Psi_D(x)$. Removing evidence gives the compatibility-only arm. This cache-based ablation has tied maxima on 158/240 queries at $N_s = 4$, resolved by the deterministic candidate order. Its exact accuracy is 7.08%; composition reaches 43.3%.

‡ BFCL combines model evidence and weighted-DFA constraints inside chain diffusion. A compatibility-only generator would need a selection rule for tied enum values and for non-enum arguments, which occur in 516 of the 550 functions. Such a decoder is outside the frozen protocol. Here p_0 weights unseen enum values.

§ Dev-selected $\alpha^\star$: 0 (n=8), 0 (n=10), .25 (n=12); the n=8, 10 TACTIC cells coincide with the α=0 row by selection.

GraphColor selects compatibility-only execution for the smaller graphs and adds a modest evidence benefit at the largest size. CatSup relies more strongly on model evidence to resolve compatibility ties. BFCL applies compatibility inside generation; its support curve is reported in Appendix G.

E GRAPHCOLOR SENSITIVITY

Table 10: GraphColor with scarce support. Exact and valid accuracy use family-clustered intervals. The matched two-part-MDL baseline acquires no edge, matching model-only exact success.

n	Base ex./val.	Hard ex./val.	TACTIC ex./val.	MDL obj. ex./val.	$\alpha^\star$	Δ [95% CI]
8	.000/.017	.300/.600	.383/.850	.000/.017	0	+8.3 [−3.3, 20.0]
10	.000/.000	.517/.800	.583/.950	.000/.000	0	+6.7 [−1.7, 15.0]
12	.000/.000	.600/.767	.700/.967	.000/.000	.25	+10.0 [3.3, 18.3]

At $n = 12$, the supplied-constraint execution reference is .683, while the soft method's point estimate is .700. At larger support sizes, effects are mixed. For $n = 8$, differences at $N_s = 4, 8, 16$ are $0, +6.7, +3.3$ points; for $n = 10$, they are $+1.7, -6.7, 0$; for $n = 12$, they are $-1.7, 0, 0$. The $\alpha = 1$ endpoint has zero exact accuracy for every reported size.

Table 11: GraphColor with DREAM-7B. Exact accuracy and paired TACTIC-minus-hard gains are in percent and percentage points. Intervals resample graph families; grouped support sizes have identical results.

n	N_s	Hard	TACTIC	Supplied-C	Δ [95% CI] (pp)
8	2	28.3	38.3	50.0	$+10.0$ $[1.7, 20.0]$
8	4	45.0	51.7	50.0	$+6.7$ $[-6.7, 20.0]$
8	8	41.7	50.0	50.0	$+8.3$ $[-5.0, 21.7]$
8	16	48.3	53.3	50.0	$+5.0$ $[-8.3, 18.3]$
8	32	50.0	55.0	50.0	$+5.0$ $[-8.3, 18.3]$
10	2	50.0	53.3	61.7	$+3.3$ $[0.0, 8.3]$
10	4	56.7	58.3	61.7	$+1.7$ $[0.0, 5.0]$
10	8–32	61.7	61.7	61.7	0.0 $[0.0, 0.0]$
12	2	60.0	68.3	70.0	$+8.3$ $[0.0, 18.3]$
12	4	73.3	75.0	70.0	$+1.7$ $[-6.7, 11.7]$
12	8–32	70.0	73.3	70.0	$+3.3$ $[-5.0, 13.3]$

MDL point-acquisition boundary. Task-adapted two-part-MDL point acquisition (two-part description length over the edge-rule language, exact model count) emits no rule at any matched $N_s \leq 16$. A first edge added to the unconstrained three-color model reduces its valid assignments by a factor $2/3$, giving a data-codelength gain of $N_s \log_2(3/2)$ bits. The acceptance test compares this with the full rule codelength, including $\log_2 \binom{n}{2}$ and grammar overhead; later gains depend on the current graph. Its hard execution therefore coincides with the Open endpoint (exact 0). At $N_s = 32$, $n = 12$, it acquires ≈ 20 edges (≈ 0.54 recall) and reaches 66.7% exact success.

E.1 DREAM-7B REPLICATION

The replication uses the same graph families, demonstrations, candidates, development grid, and family-clustered evaluation as the LLaDA protocol. DREAM's development macro accuracies for $\alpha \in \{0, .25, .5, .75, 1\}$ are $(.427, .410, .320, .163, 0)$ at $n = 8$, $(.703, .720, .677, .483, 0)$ at $n = 10$, and $(.683, .680, .657, .457, 0)$ at $n = 12$. The selected weight is frozen across the five test support sizes. At $n = 8$ and 12, the chosen $\alpha = 0$ makes the TACTIC arm compatibility-only; at $n = 10$, the selected weight is .25.

DREAM's mean color-marginal entropy at $n = 12$ is 1.040, compared with 1.044 for LLaDA and $\log 3 \approx 1.099$ for uniform marginals. The backbone argmax predictions agree on 73% of these marginals. Despite these differences, neither evidence-only endpoint achieves exact completion; the selected structural–neural balance varies with graph size.

F TENSOR-TRAIN CONSTRUCTION AND DIAGNOSTIC COST

For each exact-enumerable posterior, we form the probability tensor over binary local hypotheses, compute exact unfolding ranks, and obtain TT-SVD approximations (Oseledets, 2011; Cichocki et al., 2016) at $\chi \in \{1, 2, 4, 8, 16, 32, 64, 128\}$. Fidelity is total variation to the exact normalized posterior. Decision comparisons reuse identical candidates and frozen evidence and maximize $\log w + \log \max(\Psi, 10^{-300})$. The independent arm is the unconditioned Bernoulli-edge model; the exact and TT arms condition it on a graph family. Comparisons among TT ranks and exact inference hold the family-conditioned distribution fixed. The PoM control in Appendix J.8 additionally holds its edge marginals fixed while removing dependence. Appendix J.3 gives the working likelihood, misspecified fixed-K prior, and decision transitions. The main $N_s = 2$ evaluation contains 60 episodes/120 queries for fixed-K families and 48 episodes/96 queries for degree-sequence and connected families.

Figure 7 uses $m = 17$ in the fixed-cardinality profile and $\chi = 16$ for the decision comparison. Total-variation errors at or below the display floor are marked $\leq 10^{-16}$.

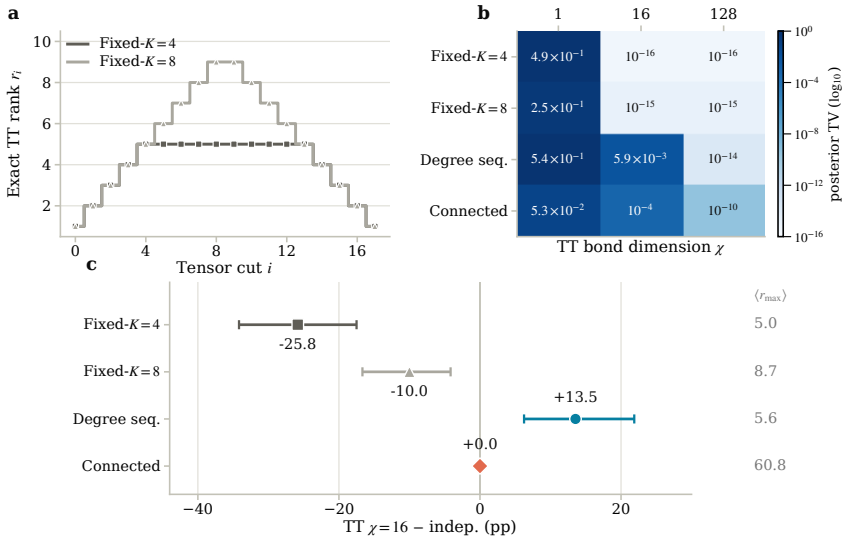


Figure 7: **Posterior rank, fidelity, and decisions.** (a) Exact rank across tensor cuts. (b) Approximation error across bond dimensions, on a logarithmic scale. (c) Paired decision changes relative to the independent model, with clustered confidence intervals; the gray column gives mean maximal exact rank. Accurate representation can have different decision effects across constraint families.

Nonnegativity and normalization. TT-SVD truncation of a probability tensor can produce small negative entries. For the fidelity and decision study, we materialize each approximation, clip negative entries, and renormalize it. The reported total variation and accuracy therefore refer to this valid dense distribution. Clipping can change TT rank. The dense diagnostic therefore measures approximation quality, while the contraction bound in Section 3 applies to a normalized nonnegative TT posterior supplied as cores. The largest observed negative magnitude is 4.5×10^{-3}, and every fidelity value in Figure 7 is measured after clipping.

Fixed-$K = 4$ has exact maximum rank 5, while fixed-$K = 8$ has mean maximum rank 8.7 and maximum 9. Degree sequence has mean maximum rank 5.6 and maximum 28. Connected posteriors are much larger, with mean maximum rank 60.75 and maximum 130 at $N_s = 2$, yet all approximations select the same candidate in this evaluation. Here the additional represented dependence leaves the candidate ranking unchanged.

The timing study includes dense construction, TT-SVD, and contraction. At $\chi = 16$, median latencies span 26.7–122.4 ms/query for $N_s = 2$ and 4.1–12.3 ms/query for $N_s = 8$ across evaluated families. Corresponding p95 values are at most .98 s and .24 s. Whole-process peak resident memory is 268 MB. The analytic bound on per-query tensor storage is at most 78 MB.

The degree-sequence TT curve is non-monotone in task accuracy: at $N_s = 2$, $\chi = 2, 4, 8, 16$ attain approximately .385, .417, .427, .417, while exact-posterior decoding is .417. Total variation falls sharply over the same ranks. Approximation fidelity and decision quality can differ because only relative scores near the winner margin determine the argmax; Figure 7 visualizes this separation.

G Open-World Boundary Analyses

G.1 JSONSchemaBench

The final census attempts 2,349 non-development schemas under the construction protocol in Appendix C.5. Of these, 2,342 resolve normally and seven are recorded as unresolved timeouts, for 0.298% attrition. The seven unresolved cases concentrate among large, recursive, or high-combination-space schemas, so missingness is not assumed random. Primary results are conditional on the resolved pool and accompanied by the full-census attrition; assigning any binary schema-level outcome to all seven unresolved schemas changes a census proportion by at most 0.298 points.

Table 12: **JSON Schema at the original frozen threshold on the complete both-class cohort.** Scores are percentages and differences are percentage points. Paired schema-bootstrap intervals accompany the posterior–hard comparison; cohort construction is detailed below.

	N_s=4	8	16	32	64
Posterior (TACTIC)	80.88	80.95	81.08	84.71	84.79
Hard commitment	83.30	83.51	84.15	84.62	85.02
Point-estimate TACTIC	62.51	76.40	81.03	83.34	84.54
Observed domain	59.62	58.82	58.94	69.26	74.31
Surprise-filtered	59.11	73.03	77.18	79.28	80.41
Δ posterior − hard	−2.43	−2.56	−3.07	+0.09	−0.23
95% CI	[−3.34, −1.53]	[−3.61, −1.56]	[−4.22, −1.95]	[−0.26, +0.43]	[−0.53, +0.09]
Δ vs. point TACTIC	+18.37	+4.55	+0.05	+1.37	+0.25
Δ vs. domain	+21.25	+22.13	+22.14	+15.45	+10.47
Δ vs. surprise-filtered	+21.76	+7.92	+3.90	+5.43	+4.38

Table 13: **JSON Schema ranking and calibration.** Means are over schema–draw cells. AUROC uses both-class cells; Brier and ECE also include the full record-bearing cohort as indicated.

	N_s=4	8	16	32	64
AUROC (tie-aware)	.847	.855	.862	.867	.870
Brier (all evaluable cells)	.262	.314	.343	.361	.381
both-class cells only	.154	.177	.193	.204	.219
ECE (all evaluable cells)	.311	.370	.401	.418	.445
both-class cells only	.179	.215	.237	.250	.275
Mean $p_{0,g}$	.642	.532	.452	.394	.342

G.1.1 CLASS-AVAILABILITY CENSUS AND ESTIMAND CORRECTION

The reference validator filters nominal invalid instances before evaluation. At each support size, the 1,497 schemas with saved evaluation cells contribute 14,934 schema–draw cells: 7,740 contain both classes, 7,186 contain valid tests only, and eight are empty. Across all five support sizes these totals are 74,670 cells, including 35,930 valid-only cells. Missing draws account for the difference from a full grid. Thus the saved-cell analysis covers 1,497 of the 2,342 resolved schemas; the other 845 resolved schemas do not enter these score aggregates.

Balanced accuracy requires both classes. We define its main evaluation cohort as the 771 schemas with both classes in all ten draws at every support size. This choice was made after evaluation; it depends on test composition rather than decoder outcomes. The resulting comparisons are conditional on this complete both-class cohort. Allowing incomplete draws gives 775 schemas and posterior–hard differences of -2.41, -2.55, -3.06, $+0.09$, and -0.23 points as N_s increases, consistent with the complete-draw analysis.

For valid-only cells, the estimable component is accept-valid; empty cells contribute no component score. Assigning zero recall to an absent class does not yield balanced accuracy; single-class cells are therefore excluded from Table 12.

G.1.2 SCOPE OF THE ATTRITION BOUND

For a per-schema score $z_s \in [0, 1]$ defined on the full census of $N = 2349$ schemas, let S_{obs} be the sum on the resolved schemas. The full-census mean lies in $[S_{\text{obs}}/N, (S_{\text{obs}} + 7)/N]$, an interval of width $7/N = 0.298\%$. A paired difference lies in $[-1, 1]$ per schema, so its full-census interval has width $14/N = 0.596\%$. These bounds follow by assigning the smallest or largest possible score to each unresolved schema. They do not bound balanced accuracy on the selected 771-schema cohort or the effects of excluding single-class schemas.

At the original .05 threshold on the both-class cohort, hard commitment leads at $N_s = 4$–16 and the differences approach zero at $N_s = 32$–64. Posterior execution accepts more valid instances but also admits more invalid ones; at low support the latter cost is larger. AUROC improves with support,

Table 14: **BFCL support and domain coverage at the original frozen weight.** Scores use the official AST checker. Coverage measures legal values observed in support; the fixed cohort tracks values unseen at the smallest support size. The disjoint-pilot recalibration is reported in Appendix J.7.

Arm	N_s=8	N_s=16	N_s=32
Hard acquired domain	.479	.507	.533
Soft compatibility (p_0=.3)	.526	.527	.527
Fully open structure-only	.535	.535	.535
Supplied full domain	.578	.578	.578
Coverage	.622	.666	.707
Fixed N_s=8-unseen cohort (207 ids): rescued fraction / accuracy			
Rescued fraction	.072	.147	.229
Hard acquired domain acc.	.279	.327	.380
Soft compatibility acc.	.486	.496	.496
Fully open acc. (reference)	.575	.575	.575

while Brier and ECE worsen (Table 13). Ranking quality and probability calibration therefore move in different directions.

G.2 BFCL

The benchmark and its official AST-based evaluation protocol follow BFCL (Patil et al., 2025). Under the original $p_0 = .3$ setting on the 550-query pool, official AST-checker results separate candidates whose enum values appeared in support from values that remained unseen. As support coverage rises from 62.2% to 70.7% (N_s=8 $\rightarrow$ 32), hard acquired-domain enforcement improves monotonically from 47.9% to 53.3%, closing 54.5% of its observed gap to the supplied full-domain reference (57.8%); the recovery is concentrated on values unseen at low support (the fixed N_s=8-unseen cohort rises from 27.9% to 38.0%). Fixed soft compatibility stays nearly flat (52.6–52.7%): its descriptive interval indicates protection against hard undercoverage at N_s=8 (soft $>$ hard by $+.047$, CI $[+.010, +.084]$), but it does not outperform structure-only open decoding (53.5%), and the soft-versus-open and hard-versus-open differences are unresolved at N_s=32. The supplied full-domain reference reaches 57.8%, exceeding fully open decoding by 4.3 points and providing modest but real headroom; it illustrates the remaining benefit of full domain coverage. Additional demonstrations primarily repair hard undercoverage in this benchmark.

BFCL's soft–hard gap changes from $+4.7$ to -0.6 points as support grows. This differs from JSON Schema's low-support preference for hard commitment: BFCL evaluates generated calls, whereas JSON Schema classifies validity without model evidence. Their different support trends reflect these distinct decision objectives.

The disjoint-pilot evaluation in Appendix J.7 narrows the soft–hard gap while preserving Open's advantage.

G.3 Official URPiLs presence-constraint comparisons

Two adapters apply the official URPiLs v3.0.0 engine (Wiegand et al., 2024) to property- and argument-presence projections. The engine is unmodified. These secondary results characterize the chosen projections. Reported scores condition on successful runs; adapter exceptions and engine crashes are counted separately.

JSB-Presence. The adapter represents property presence using a singleton `Record`, shared `Property` objects, and Boolean `present(Record,Property)` relations. Its universe is fixed from the visible schema. A keyword whitelist admits 709 schemas: six presence-clean and 703 presence-mixed. Of 35,450 acquisition runs, 28,457 succeed, 6,507 record adapter exceptions, and 486 record engine crashes. On successful runs, the official engine's schema-macro scores are .788, .710, .708, .721, and .720 at $N_s = 4, 8, 16, 32, 64$, with 570, 656, 669, 689, and 693 schemas represented. These scores use the adapter's test labels. Its invalid-test counts differ from those of the reference-filtered main experiment, so they describe the presence projection and are not paired comparisons with Table 12.

BFCL-Presence. The argument-presence adapter uses the 550-query pool, three draws, and $N_s \in \{8, 16, 32\}$. Among successful runs, 85% yield non-vacuous models. Learned constraints accept 99.6% of support records but reject 27.6–28.4% of frozen generations; 89.9% of rejected hard-arm generations omit a required argument. A subset of 1,163 planes is redecoded with the acquired presence constraints, after excluding 54 planes whose conditional dependencies cannot be represented by the executor. Accuracy on this subset is .344 (.354/.346/.335 across support sizes), compared with .060 for matched hard execution and .055–.065 for the open reference. Selection targets planes with rejected generations, and the presence executor leaves enum values unrestricted. The improvement therefore concerns this subset and failure mode; it is not a full-pool comparison with enum enforcement.

G.4 BPI NEGATIVE PROBE

A second business-process probe produces a 32.1-point degradation (chain top-1 .719 to evidence-argmax .398 on the same candidate set). The failure is consistent with severe positive-support undercoverage: acquired constraints remove valid values before neural evidence can rank them. We retain this negative result without tuning a rescue method because it tests the same boundary as BFCL.

H REPRODUCIBILITY DETAILS

Models. CatSup and GraphColor use LLaDA-8B-Instruct, with DREAM-7B-Instruct replications. CIDDS uses DREAM-7B-Instruct, with a LLaDA-8B-Instruct replication; BFCL uses DREAM-7B-Instruct. JSON Schema classification uses no language-model evidence. Mask tokens, numerical precision, prompts, and evidence extraction are specified in Appendix C.7; BFCL decoding settings are in Appendix J.1.

Data and evaluation. Appendix C defines the generated CatSup and GraphColor tasks, CIDDS splits and masked fields, JSONSchemaBench cohorts, and BFCL query pool. Support sampling and candidate construction are held fixed within each matched comparison. Development data determine tuned weights and thresholds; CatSup uses fixed constants. Appendices J.5 to J.7 specify the held-out-rule study and additional calibration sets.

Statistical analysis. CIDDS intervals use paired differences across support draws. CatSup resamples episodes in the fixed-rule study and rules in the held-out study. GraphColor resamples graph families, and JSON Schema resamples schemas. BFCL recalibration resamples paired queries at a fixed support draw, and the marginal-matched study resamples graph structures. The corresponding protocols specify resampling counts, cohort exclusions, and failure scoring.

Code and data access. The supplement contains source code, configurations, and compact evaluation records. Large model caches, generated pools, and checkpoints are omitted; the accompanying instructions describe experiment regeneration.

I EXTENDED RELATED WORK

Discrete diffusion and constrained generation. Structured discrete diffusion introduced flexible noising processes over finite state spaces (Austin et al., 2021); subsequent work developed controllable text diffusion (Li et al., 2022), ratio-based objectives (Lou et al., 2024), and simplified masked-diffusion training (Sahoo et al., 2024; Shi et al., 2024). These models supply the evidence channel in our setting, while acquisition supplies compatibility with the hidden constraint. Grammar-constrained decoding has been implemented through language intersections, grammar masks, and non-invasive guidance (Park et al., 2025; Ugare et al., 2025; Willard & Louf, 2023). These techniques implement execution for a supplied constraint, while HCD introduces uncertainty in acquiring that constraint.

Learning and uncertainty. Automated constraint modelling predates recent data-driven acquisition systems (O'Sullivan, 2010). More broadly, programming-by-example and probabilistic concept

learning show how small demonstration sets induce multiple structured explanations (Gulwani, 2011; Lake et al., 2015).

Predictive uncertainty and calibration. Bayesian model averaging provides the general predictive rationale for retaining model uncertainty (Hoeting et al., 1999), while posterior consistency characterizes its asymptotic disappearance under identification (Schwartz, 1965). Semiring-valued constraints provide a complementary foundation for graded feasibility (Bistarelli et al., 1997). Open-domain acquisition also requires estimating probability mass assigned to unseen values. Our constructions follow the Good–Turing tradition (Good, 1953; Painsky, 2022); Brier scores complement calibration error when assessing probabilistic compatibility (Brier, 1950). Together, these connections distinguish uncertainty over the acquired model, coverage of its domain, and the scale of its predictive scores, each of which affects downstream execution.

Tensor probabilities and schema witnesses. Tensor networks have also been used directly as normalized generative models (Han et al., 2018); TACTIC uses them to represent a posterior over constraints. Exact witness generation complements data-driven JSON schema discovery when schemas contain recursive Boolean structure (Attouche et al., 2022).

Schema discovery and network generation. Schema discovery (Baazizi et al., 2019; Spoth et al., 2021; Mior, 2023) and rule-guided network generation (Hè et al., 2026) provide neighboring application settings. HCD focuses on how residual uncertainty in the acquired constraints affects a downstream decision, connecting these acquisition settings to constrained execution.

J ACQUISITION DEFINITIONS AND ADDITIONAL VALIDATION

This appendix specifies acquisition and execution and reports the baseline, failure, scoring, and holdout studies. Missing or failed predictions remain in the evaluation denominators.

J.1 CONCRETE ACQUISITION AND EXECUTION DEFINITIONS

CatSup: conditional categorical supports. For every observed antecedent $x_i = v$ and another field j, form a group $g = (i, v, j)$ with counts $n_{g,u}$ over the known field domain D_j, total n_g, and observed set O_g. Groups with $n_g = 0$ impose no restriction. The latent allowed set is a nonempty $A_g \subseteq D_j$, $K = |D_j|$. The prior assigns .85 to D_j and distributes .15 uniformly across proper nonempty cardinalities and then subsets:

$$\pi(A) = \begin{cases} .85, & A = D_j, \\ .15/[(K-1)\binom{K}{|A|}], & 1 \le |A| < K. \end{cases}$$

The singleton-domain case is deterministic. With symmetric Dirichlet concentration $a = 1$, the working likelihood, up to a factor independent of A, is

$$L_g(A) = \mathbf{1}[O_g \subseteq A] \frac{\Gamma(|A|a)}{\Gamma(n_g + |A|a)} \prod_{u \in A} \frac{\Gamma(n_{g,u} + a)}{\Gamma(a)}.$$

Enumerate the at most $2^K - 1$ states and normalize $\pi(A)L_g(A)$. Let $I_{g,u} = P(u \in A_g \mid D^+)$. The implemented compatibility multiplies $\max(I_{g,x_j}, 10^{-12})$ over groups whose antecedents fire. The composed arm maximizes $\sum_f \log \max(p_f(x_f), 10^{-12}) + \log \Psi(x)$ over the full field product, without a mixing sweep. Field probabilities come from restricted categorical softmaxes at the masked scaffold positions. Hard is the intersection of the observed co-occurrence sets, $x_j \in O_g$ for every firing group, followed by the same evidence argmax; it is *not* the posterior-MAP-set comparator. An empty hard set abstains and counts as wrong; exact ties use the first candidate in the fixed product order.

These conditional groups approximate the clause generator. The original three rules and six fixed queries use support sampled without replacement, so the iid Dirichlet–multinomial likelihood is also a modeling approximation. At $N_s = 32$, the create/config supports exhaust their legal sets and their repeated episodes share the same support set. Open and supplied-C both score 50% on the original

six queries, although 80/240 predictions differ at $N_s = 4$; the two arms therefore have equal accuracy despite different predictions. The independent-rule extension below evaluates generalization beyond these queries.

GraphColor: pairwise working likelihood. The independent-edge prior has $\pi_e = .5$. A pair co-colored in any demonstration has edge probability zero. Otherwise the implementation uses

$$\gamma_e = \frac{.5}{.5 + .5(2/3)^{N_s}}, \qquad \Psi(x) = \prod_{e=(i,j):x_i=x_j} (1 - \gamma_e).$$

This likelihood treats a non-edge as independently co-colored with probability $1/3$ in each demonstration; it omits the legal-coloring normalization $|S_C|^{-N_s}$. Actual demonstrations are uniform proper colorings drawn with replacement. The acquisition likelihood is thus a pairwise approximation to the sampling model. The hard arm enforces pairs with $\gamma_e \geq .5$, maximizes evidence among hint-consistent feasible colorings, and falls back to the hint-consistent evidence argmax if none is feasible. The soft probability-normalization rule, hint-mask order, grids, and data split are specified in Appendix C. Input evidence is the product of cached per-vertex color probabilities.

CIDDS: confidence-weighted implications. The six discretized fields are protocol, source/destination address categories, source/destination port categories, and flags. Completion candidates enumerate the frozen field-vocabulary product for the two masked fields; measured target coverage is one. For every ordered field-value implication $(a = v_a) \Rightarrow (b = v_b)$, let n_a count its antecedent and n_{ab} count joint occurrences in D^+. The soft weight is

$$\omega_r = \frac{n_{ab} + .1}{n_a + .1|V_b|}, \qquad \omega_r > .05, \quad n_{ab} \geq 1.$$

Only retained, firing implications enter the weighted mean in equation (8); no firing weight gives compatibility one. Plain hard uses unsmoothed confidence one and antecedent support at least one, conjoins the implications, and falls back to Open if infeasible. Tuned-hard instead chooses confidence threshold $\tau \in \{.5, .6, .7, .8, .9, 1\}$ and minimum antecedent support $m \in \{1, 2, 3\}$. It selects the highest development exact count among settings with abstention at most $.4$ (or all settings if none qualify), then the lowest abstention count; remaining ties follow the ascending grid order. Selection is per support size, seed, and mask, using random-pool support proxies also for temporal evaluation. At execution, a contradiction among observed fields or an empty feasible completion set produces an abstention, scored zero in the full denominator. The added fallback control below reselects the threshold after changing that convention.

For value v with tokenization $t_1, \ldots, t_L$, CIDDS stores the mean masked-span log probability $e_f(v; y) = L^{-1} \sum_{j=1}^{L} \log p_\theta(t_j \mid y)$, then sums these field scores. This length normalization makes values with different token counts comparable. Both backbones use the fixed `PORT_` serialization; the context-free cache is independent of support draws. The supplied-rule reference uses 111 benchmark Anuta rules converted to symbolic predicates, with zero violations on the 10,000 processed records, and selects the highest-evidence legal completion. Several completions can satisfy these predicates, so the supplied-rule reference can select a different answer from the original masked record.

JSON Schema: open branch compatibility. The acquisition input is the Open-v2 field-only schema with object type. A base validator $B(x)$ adds always-present required properties and acquired uniqueness checks; inferred closed enums are disabled in the posterior arm. Let b range over distinct observed sets of present property names, with count c_b, K distinct sets, and N object positives. If a branch group exists ($K \geq 2$), define

$$p_{0,g} = \frac{1}{1 + N}, \qquad h_g(x) = \max_{b:\, \mathrm{keys}(x) \subseteq b} \frac{c_b + 1}{N + K},$$

with a maximum of zero if no branch matches. An absent group contributes one; the stored fallback .85 is unused in this case. For a demonstrated property-presence dependency $a \Rightarrow b$, n_r counts antecedent occurrences and $S_r = \max(\widehat{P}(b \text{ present}), 10^{-12})$. Set $q_r = .5/[.5 + .5S_r^{n_r}]$. The score is

$$\Psi(x) = B(x)\,[p_{0,g} + (1 - p_{0,g})h_g(x)] \prod_{r:\, a \in x,\, b \notin x} \max(1 - q_r, 10^{-12}),$$

omitting the bracket when no branch group exists. Accept iff $\Psi(x) \geq \tau$, with original $\tau = .05$ and rebuilt-development $\tau = .2$. The maximum-over-branches rule gives a local approximation to schema compatibility. Hard uses the same base, binary membership in some observed branch, and all dependencies as hard predicates.

The other Table 12 rows are deterministic validators with no posterior threshold: point-estimate TACTIC compiles acquired version-space rules; observed domain closes observed per-property values (enumerations when there are 2–20 distinct non-null values), adds always-present required fields and the implementation's array heuristic; surprise-filtered uses the same high-order candidates with empirical independent-presence scores, threshold .05, and description-length tie order. The latter uses a task-specific `urpils` adapter. Required/enum rules and qualifying uniqueness checks are also included by that adapter. Their exact executable definitions are `acq_tactic`, `acq_domain`, and `acq_urpils` in `p4_jsb_baselines.py`. The separate official-engine comparison remains in Appendix G.3.

BFCL: weighted automaton and diffusion schedule. Support comprises ground-truth calls from *other* live queries for the same function, excluding the evaluated query's own answer; a shuffled leave-one-out list supplies nested prefixes capped by its length. All prompts strip enum lists while retaining visible names, types, and required fields. The acquired domain A is the observed set. Soft enum compatibility is one for $v \in A$ and p_0 otherwise (original .3; disjoint-pilot calibration .85). A character trie recognizes acquired strings; a free state admits other quoted strings and puts $\log p_0$ on their closing-quote transitions. Tokenizer composition propagates these penalties to token edges. Call syntax, function/key identity, required fields, and types remain hard in every arm. Open omits enum restrictions; hard accepts only A; supplied-C accepts the full supplied enum domain.

At each diffusion step, the engine combines per-position model log probabilities, fixed-token clamps, and automaton edge weights using backward sum-product, samples an accepted joint token sequence, and computes constrained marginal entropies. It commits the lowest-entropy masked positions under the DREAM transfer schedule; committed tokens are not remasked. The final span is checked for acceptance. Runs use DREAM-7B-Instruct, one sample, 128 steps, generation length 256, temperature one, no nucleus truncation, entropy scheduling with deterministic top-k commit order, and tracked argument cap two. Seeds are $s \cdot 1{,}000{,}003 + i \cdot 10{,}007 + k$ for draw, entry, and sample. Infeasible chains, parse failures, and checker failures count as errors among all 550 queries.

The sparse automaton inference and diffusion integration reimplement the constrained-decoding approach of Dang & Ermon (2026); the soft enum machine and weighted transitions are the additions here. The original pilot and the additional disjoint calibration are documented in Appendices C.5 and J.7.

J.2 CIDDS: SAME-SUPPORT ICL AND HARD-EXECUTION CONTROLS

Generative ICL protocol. The ICL baseline places exactly the same demonstrations as TACTIC in context, uses the same frozen DREAM backbone and masked record serialization, and greedily generates all masked-span tokens in one parallel forward pass, with no rule miner or compatibility score. Token-prefix parsing selects the longest matching candidate value; unmatched generations count as errors. Open instead rescores candidates without demonstrations in context.

Before test evaluation, two prompt variants and two schedules were compared on 200 disjoint canonical-development records at $N_s = 4, 32$ and three masks. The header prompt with parallel generation scored .4383; iterative MaskGIT scored .4542 but took about $44\times$ longer. We selected parallel generation for its lower cost and evaluated this one-pass comparator on the test set. Its port-pair mask was already near zero on development; this serialization dependence limits generalization of the gap.

Both test splits complete all 20 original support draws, four budgets, three masks, and 1,000 records per cell: 480,000 predictions, with no missing cells. Each draw averages its 24 cells; paired t intervals use the 20 draw-level differences. Exact completion accuracies are 45.4792%, 12.0167%, and 71.4690% for ICL, Open, and TACTIC, respectively. These intervals describe support-draw variation on the fixed test records.

The common-draw comparison is summarized in Table 2.

Failure mechanism and fallback. We distinguish the *plain-hard* conjunctive mask from the development-tuned hard comparator. Among plain-hard-wrong/TACTIC-right transitions, gold was excluded by that mask in 23.22% (DREAM) and 26.04% (LLaDA); in the remaining $76.78\%/73.96\%$, gold was feasible and compatibility changed the evidence ranking. The excluded-gold correction path contributes $4.58/5.45$ percentage points over all records. Reverse transitions contribute $.71/1.64$ points. The pre-fallback plain-hard mask excludes gold on 20.69% of all records. From $N_s = 4$ to 32, exclusion falls from 48.9% to 2.3%, while feasible-set ranking errors increase.

For the tuned-hard control, replace abstention with Open fallback and reselect its threshold on development under that convention. TACTIC still leads by $18.10/17.70$ points (Table 2). A second control thresholds the same weighted compatibility Ψ, selects its threshold on development, ranks accepted candidates by evidence, and uses Open fallback. It scores $40.31/44.78\%$, below original tuned-hard ($52.06/52.08\%$). Thus neither abstention alone nor changing the hard acceptance predicate to a threshold on Ψ explains away the advantage. These controls support graded re-ranking without attributing the entire improvement to false-exclusion recovery.

J.3 Correlated working-model diagnostics

Let E_0 contain the pairs never co-colored in D^+ and $r_e = \gamma_e/(1-\gamma_e)$. The family-conditioned working distribution is

$$\rho_F(E) \propto \mathbf{1}[E \subseteq E_0,\ E \in F] \prod_{e \in E} r_e, \qquad \Psi_F(x) = \sum_{E \subseteq \mathrm{Sep}(x)} \rho_F(E).$$

Fixed-K uses elementary symmetric polynomials; degree-sequence/connectedness enumerate subsets, omitting episodes beyond the 2^{19} enumeration cap. Query hints restrict candidate colorings but do not update this distribution. The independent comparator uses the original γ_e without family conditioning, so the comparison changes marginals as well as dependence. The TT/exact comparison, in contrast, holds ρ_F fixed.

The fixed-$K = 4, 8$ rows apply their assumed cardinality to all 60 graphs even though each true graph has 10–14 edges. Exact inference is therefore exact for a misspecified family. At $\chi = 16$, every TT decision agrees with exact inference (120/120 queries per fixed-K row and 96/96 per other row); fixed-K TV is at most 2.7×10^{-14}. This agreement traces the losses to the assumed constraint family. For $K = 4$, 31 queries change from correct to wrong and none from wrong to correct; for $K = 8$, the counts are 14 and two. On all 31/14 losses, the exact family's reduced compatibility margin falls below the wrong candidate's evidence advantage; 30/31 and 13/14 such winners are invalid. Mean gold compatibility advantages shrink from $3.69/2.27$ to $.83/1.13$ nats, against evidence deficits of $1.66/1.52$ nats. Accuracy and validity decline together. Together, these transitions show how family misspecification changes the compatibility margin and the selected answer.

J.4 Score-rule sensitivity and theoretical assumptions

Cache-only comparisons retain the same acquisitions, candidates, evidence, and task-specific zero handling. Fixed-log scoring uses $L_w + \beta L_C$ with β selected on development; the strict variant excludes zeros in positively weighted channels. On CIDDS, development selects $\beta = 30$, yielding 70.80% versus 71.47% for the main scoring rule: paired difference $-.67$ points, CI $[-.97, -.36]$. Unit weights instead score 18.67%. Scale selection matters, even though a tuned fixed-log rule nearly recovers the original min–max result. A strict log min–max variant with both weights positive scores 70.98% (difference $-.49$, CI $[-.83, -.15]$).

For GraphColor, the measured ranges of the two floored, probability-normalized log channels are both 27.63 nats; on these candidates the main scoring rule coincides with a fixed-weight combination of those *transformed* channels, with $\beta = (1-\alpha)/\alpha$ for $\alpha > 0$. The equivalence applies after the probability transformation; posterior-consistency analysis instead concerns the raw compatibility. CatSup and the correlated study already use fixed log-score sums. The min–max counterexample in Appendix A.6 remains valid: shrinking compatibility differences can retain full normalized range.

The analysis separates posterior properties from score transformation. Proposition 2 applies to posterior compatibility under the product rule, and Theorem 3 assumes a correctly specified iid likelihood. CatSup and GraphColor use the working likelihoods above; CIDDS uses weighted

satisfaction. Proposition 5 assumes common positive support, so zero-support decisions require a separate support check. Proposition 6 supplies the additional range condition for strict log min–max scoring. The affine-hull lemma gives a theoretical identification example.

J.5 CatSup with independent rules and held-out queries

The held-out-query design generates ten development and 30 test rules per template (30/90 total), using root seed 20260920 and independent random streams. Each rule combines mutual exclusion, conditional dependency, and branch coupling; equal legal sets are deduplicated within template across both splits. Four distinct legal query calls per test rule are reserved and excluded from every support draw. Five without-replacement draws per rule use requested budgets $4, 8, 16, 32$, capped by the remaining legal pool. The 30 development rules are retained but not used to select a mixing rule. The original LLaDA evidence interface, posterior constants, and composed score are frozen.

Table 15: CatSup generalization to independent rules and held-out queries. Exact accuracy is in percent; evaluated queries are excluded from support.

Arm	$N_s = 4$	8	16	32
Open	56.39	56.39	56.39	56.39
Supplied-C	69.17	69.17	69.17	69.17
Hard observed intersection	1.17	13.94	36.67	49.11
Compatibility only	0.78	1.33	2.39	2.89
TACTIC composition	54.28	56.28	57.11	58.00

There are zero query-in-support violations. Mean coverage of the full legal set is $.157, .308, .522, .664$; only 160/450 draws realize the full requested 32 examples. Aggregate exact accuracy is computed within rule, then within template, then equally across templates. Intervals resample independent rules within template (5,000 replicates), carrying all queries and support draws together. Pooled composed-minus-hard is 31.19 points, CI $[26.90, 35.78]$; composed-minus-Open is .03, CI $[-1.22, 1.39]$. Supplied-C exceeds Open by 12.78 points, CI $[8.89, 16.67]$, showing headroom absent in the original fixed-query design. Compatibility-only has tied maxima on 87.28% of query evaluations. The extension preserves gains over hard execution while composition and Open remain comparable. We report it separately from the original fixed-query study.

J.6 JSON Schema development reconstruction and threshold selection

The protocol-matched development set contains 198 constructed schemas, with two timeouts among 200 attempts. Evaluation at $N_s = 8$ over ten draws produces 1,980 cells and 17,420 records. The predeclared complete-both-class rule retains 144 schemas; the other 54 fail this class-availability requirement.

Selection maximizes schema-macro balanced accuracy, averaging draws within each schema, over the fixed grid $\{.01, .025, .05, .1, .2, .3, .5, .7, .9\}$; ties follow grid order. This development cohort selects $\tau = .2$, with balanced accuracy $.8366$. The legacy 231-schema development rows select $.01$ under the same objective. Both paths are retained alongside the original frozen $.05$ setting. Development tie-aware AUROC is $.8465$.

Table 16: JSON Schema across calibration paths. Accuracy is in percent; the final column gives rebuilt-development posterior minus hard, in percentage points with paired schema-bootstrap intervals.

N_s	Frozen $(.05)$	Legacy dev $(.01)$	Rebuilt dev $(.2)$	$(.2) -$ Hard [95% CI]
4	80.88	80.99	80.77	$-2.53\ [-3.44, -1.63]$
8	80.95	81.13	83.79	$+0.28\ [0.05, 0.56]$
16	81.08	81.20	81.87	$-2.28\ [-2.86, -1.71]$
32	84.71	81.23	81.07	$-3.56\ [-4.33, -2.80]$
64	84.79	81.22	80.55	$-4.46\ [-5.37, -3.60]$

The rebuilt-development threshold yields a small positive posterior–hard difference at $N_s = 8$ and negative differences at the other four support sizes. Calibration at one support size therefore transfers unevenly across support budgets. The legacy .01 path remains below hard at every budget; the original .05 curve approaches hard at high support. Test comparisons average ten draws within each of the same 771 schemas and use paired schema-bootstrap intervals with 10,000 replicates. These intervals are unadjusted for multiple support-size comparisons. Threshold selection uses development outcomes; AUROC and calibration diagnostics retain their threshold-independent definitions.

J.7 BFCL CALIBRATION ON A DISJOINT PILOT

Under the original eligibility criteria, the 550-query pool exhausts all eligible queries. A predeclared change to the candidate-argument cap, from six to eight, provides 45 additional eligible queries, all outside that pool. Other eligibility rules, the support and generation seed formulas, and the diffusion executor are retained. The pilot uses $N_s = 8$, one support draw, one sample, 128 steps, and the fixed grid $p_0 \in \{.05, .3, .5, .85, 1\}$. Selection maximizes official AST correct count, retaining .3 if tied for best and otherwise choosing the smallest tied weight.

Official pilot correct counts are $21, 19, 20, 22, 21$ out of 45 in grid order, selecting .85. A preliminary selection under the stricter in-script score yielded .05; this protocol deviation is reported separately below. The official-checker selection is evaluated on the original 550-query pool at support draw zero, using the same per-entry generation seeds as the reference arms. This single-draw comparison is separate from the original three-draw support curve.

Table 17: BFCL disjoint-pilot recalibration on matched queries and support. Differences compare the selected soft weight with each reference, in percentage points. Intervals use paired query bootstrapping; p is the exact McNemar value.

Reference	Difference [95% CI]	Wins/losses	p
Historical soft (.3)	-2.00 $[-4.00, -0.18]$	$9/20$	0.061
Hard	$+2.18$ $[-1.64, 6.00]$	$63/51$	0.303
Open	-3.09 $[-5.64, -0.55]$	$16/33$	0.021

The selected weight scores $277/550$ correct, compared with $288/550$ for the original .3 weight, $265/550$ for hard execution, $294/550$ for Open, and $318/550$ for supplied-domain execution. Its soft–hard difference is positive but unresolved by the paired interval; Open remains stronger. The .85–.3 bootstrap interval excludes zero, while the exact two-sided McNemar test gives $p = .061$; both summaries are retained. Intervals resample paired queries with 10,000 replicates and seed zero, conditional on the support draw.

The alternative-score run at .05 reaches $287/550$ (52.18%), .18 points below .3. Removing the original 50 pilot IDs from both compared arms gives 51.60% for .85 and 53.60% for .3 on the remaining 500 queries. The new selection has no pilot-query overlap with the evaluation pool; its expanded eligibility criterion and reuse of the established test pool define the calibration setting. These results establish weight sensitivity and a smaller soft–hard margin under disjoint selection.

J.8 PRODUCT OF EXACT MARGINALS CONTROL

Let $P_F(E \mid D)$ be the exact family-conditioned edge posterior of Appendix J.3, with marginals $\rho_e = P_F(e \in E \mid D)$. The product-of-marginals control is

$$Q_{\mathrm{PoM}}(E) = \prod_e \rho_e^{\mathbf{1}[e \in E]} (1 - \rho_e)^{\mathbf{1}[e \notin E]}, \qquad \Psi_{\mathrm{PoM}}(x) = \prod_e \big[(1 - \rho_e) + \rho_e \mathbf{1}[x \text{ separates } e]\big].$$

It preserves every exact edge marginal while removing dependence. For fixed cardinality, $\rho_e = r_e e_{K-1}(r_{-e})/e_K(r)$ is computed by prefix/suffix elementary-symmetric dynamic programming. Degree-sequence and connected families use exact enumeration of edge-containing family weights. Candidates, cached evidence, tie order, and $\log w + \log \max(\Psi, 10^{-300})$ scoring match the original diagnostic.

Paired intervals resample graph structures, retaining both queries per structure (10,000 replicates, seed zero).

Table 3 reports the marginal-matched comparison.

PoM retains the degree-sequence advantage and the fixed-cardinality losses relative to the original unconditioned model. Exact joint structure adds 4.17 points over PoM for $K = 8$; differences in the other families lie within 1.05 points. The PoM and exact winners agree on 108/120, 105/120, 88/96, and 96/96 queries, respectively. Thus joint dependence can change individual decisions, while the large original conditioning effects are primarily reproduced by the changed edge marginals. The fixed-cardinality assumptions remain misspecified relative to the generating graphs.