CONDITIONAL CONSTRAINED ROUTING AND METRIC BRIDGING FOR SYMBOLICAI WORKFLOWS UNDER CPU-ONLY BUDGETS

Anonymous authors
Paper under review

ABSTRACT

Modular language-agent systems increasingly combine large language models, tool calls, and symbolic operators, but objective design and evaluation practice remain misaligned: trajectory-quality surrogates, benchmark-native outcomes, and deployment constraints are often optimized in isolation. We present a hybrid framework for SymbolicAI workflows that jointly optimizes constrained routing, bridge-calibrated metric alignment, and uncertainty-qualified acceptance under CPU-only budgets. The method defines a constrained router objective that couples trajectory quality, native task loss, route cost, and uncertainty terms; a bridge model that maps trajectory-level signals to heterogeneous benchmark-native outcomes; and a one-sided confidence predicate that controls deployment acceptance under practical gain thresholds. Across a benchmark suite spanning interactive tasks and code-oriented slices, the proposed router improves mean joint objective relative to strong planning and tool-use baselines (0.739 versus 0.701 for fixed-route SymbolicAI and 0.688 for OR-Toolformer-style routing), and the full bridge model improves both gain and calibration relative to distance-only controls (mean $\Gamma = 0.060$; AUROC $= 0.742$; ECE $= 0.118$). Drift stress tests show higher robust success and lower invalid-call rates for symbolic fallback routing than static alternatives. Symbolic audits support most formal obligations while exposing two unresolved obligations, so theorem-strength claims are stated conditionally rather than globally. The resulting manuscript provides a claim-evidence-uncertainty closure that is explicit about what is proven, what is empirically supported, and where boundary failures begin.

1 INTRODUCTION

Modern agentic systems no longer execute as monolithic prompt-response loops. They compose language reasoning, external tools, retrieval modules, and symbolic subroutines into multi-step trajectories whose quality depends on both local decisions and global orchestration. This shift has produced rapid progress in structured reasoning and tool use, including reasoning-action interleaving, program-aided execution, tree and graph search, and planner-coupled language control (Yao et al., 2022; Gao et al., 2022; Yao et al., 2023; Besta et al., 2023; Zhou et al., 2023; Liu et al., 2023a). At the same time, evaluation ecosystems have diversified: interactive agent benchmarks, software engineering issue-resolution suites, and multi-step tool-use benchmarks expose materially different native success criteria (Liu et al., 2023b; Jimenez et al., 2023; Xie et al., 2024; Ma et al., 2024; Yehudai et al., 2025). The central technical difficulty is therefore no longer just solving one task family; it is aligning objective functions and evidence across heterogeneous regimes without hiding uncertainty.

SymbolicAI offers a natural test bed for this alignment problem because it explicitly models compositional workflows and trajectory-level quality through VERTEX-style similarity concepts while enabling modular solver routing (Dinu et al., 2024; ExtensityAI, 2026a;b). However, prior phases in this project identified a persistent contradiction boundary: high trajectory similarity does not automatically imply high benchmark-native completion, especially under API drift, tool schema changes, and tight compute envelopes. This contradiction is not a minor reporting detail; it alters what can be claimed about optimization, generalization, and deployment readiness (Schick et al., 2023; Patil et al., 2023; Liu et al., 2025c; Amugongo et al., 2025; Jiang et al., 2025a).

This paper addresses that boundary with a hybrid contribution: we combine formal optimization and theorem-audit machinery with benchmark-grounded empirical validation, then report claims conditionally where symbolic closure is incomplete. Rather than presenting a single scalar score, we expose a chain from problem statement to objective, from objective to executable protocol, and from protocol to caveated claims. The practical scope follows the user-imposed

constraints of this run: CPU-only execution, open compute budget (still not globally fixed), and mixed benchmark families.

Contributions.

- We define a constrained routing objective for SymbolicAI workflows that jointly optimizes trajectory quality, benchmark-native loss, route cost, and uncertainty under explicit feasibility constraints.
- We introduce a bridge-calibration layer that maps trajectory-level surrogates to heterogeneous native benchmark outcomes and quantifies bridge gain with a manuscript-defined audit quantity.
- We formalize an uncertainty-aware acceptance predicate with complete proof blocks under bounded assumptions, and we pair theorem claims with symbolic obligation checks and counterexample conditions.
- We execute a hybrid validation package showing objective and calibration gains over matched baselines, while explicitly reporting unresolved symbolic obligations and data-provenance limits that bound interpretation.

The remainder of the manuscript is organized as follows: Section 2 synthesizes consensus and contradiction structure in prior work; Section 3 defines symbols, spaces, assumptions, and optimality criteria; Section 4 presents the method and integrated algorithm; Section 5 states and proves the main formal claims; Section 6 and Section 7 report the experimental and symbolic evidence; and section 12 describes unresolved gaps and concrete follow-up experiments.

2 Related Work and Novelty Boundary

2.1 Reasoning and Planning Control

Reasoning-action frameworks establish that explicit intermediate structure can improve long-horizon decision quality relative to single-pass prompting. ReAct-style interleaving, program-aided decomposition, and deliberate search over thought branches each expose distinct tradeoffs between interpretability, branching cost, and error propagation (Yao et al., 2022; Gao et al., 2022; Yao et al., 2023; Besta et al., 2023). LATS and planner-coupled pipelines push this direction toward explicit planning constraints (Zhou et al., 2023; Liu et al., 2023a). The consensus is clear: structured control helps. The contradiction is also clear: stronger search often raises compute demands and can degrade robustness under runtime constraints. For a CPU-only deployment regime, this tension becomes first-order rather than incidental.

2.2 Tool Learning and Orchestration

Toolformer and Gorilla demonstrate that API-grounded behavior can be learned or adapted, while HuggingGPT, AutoGen, and DSPy emphasize compositional orchestration and programmatic pipeline control (Schick et al., 2023; Patil et al., 2023; Shen et al., 2023; Wu et al., 2023; Khattab et al., 2023). Recent surveys and multi-LLM analyses highlight instability in argument validity and version-sensitive behavior, especially when tool schemas change or retrieval support is imperfect (Xu et al., 2025; Shen et al., 2024). These findings motivate our explicit validity constraint and drift-caveated interpretation instead of treating tool success as stationary.

2.3 Benchmark Heterogeneity and Metric Mismatch

AgentBench, SWE-bench, TravelPlanner, and multimodal tool-use benchmarks encode non-equivalent native outcomes: task completion, issue resolution, action validity, and environment-dependent utility are not interchangeable (Liu et al., 2023b; Jimenez et al., 2023; Xie et al., 2024; Ma et al., 2024). SymbolicAI's trajectory similarity framing is valuable but not sufficient as a universal surrogate (Dinu et al., 2024). Prior project phases therefore identified a blocking gap: objective closure requires an explicit bridge between trajectory and native metrics, with disagreement analysis rather than aggregate-only reporting.

2.4 Formal Guarantees and Neuro-Symbolic Validation

Neuro-symbolic composition and verification-oriented lines suggest that constrained symbolic structure can improve compositional reasoning and post-hoc validity (Kamali et al., 2025; Yang et al., 2025; Liu et al., 2025a). However, many practical systems still stop short of workflow-level guarantees under perturbations, and proofs are often not tied to executable assumption audits. OR-Toolformer-style optimization formulations provide stronger optimization language for tool planning but still require careful boundary accounting in non-stationary settings (Zhang et al., 2025).

2.5 Robustness, Uncertainty, and High-Stakes Transfer

In healthcare and other high-stakes domains, retrieval augmentation can improve average metrics while uncertainty and critical-error behavior remain unresolved (Liu et al., 2025c; Amugongo et al., 2025; Feng et al., 2024; Muludi et al., 2024; Liu et al., 2025b; Ong et al., 2024). Medical-agent benchmarks further emphasize this gap by exposing scenarios where moderate aggregate gains coexist with unacceptable error patterns (Jiang et al., 2025b;a; Azimi et al., 2025). These findings directly motivate our uncertainty-qualified acceptance predicate and our refusal to claim global guarantees where bounded assumptions fail.

Novelty boundary. We do not claim to invent trajectory metrics, tool orchestration, or confidence bounds in isolation. Instead, this work contributes a closed hybrid assembly: (i) constrained SymbolicAI routing objective with explicit feasibility and optimality criteria, (ii) manuscript-specific bridge gain quantity for metric alignment, (iii) theorem-to-audit linkage with explicit pass/fail obligations, and (iv) claim-level caveating when formal closure is partial.

3 Problem Setting, Symbols, and Assumptions

3.1 Workflow Graph and Objectives

Let $G = (V, E)$ denote a typed workflow graph where each node corresponds to a solver invocation and each edge carries an artifact transition. This graph-level modeling perspective is adapted from the SymbolicAI formulation (Dinu et al., 2024). We evaluate policies over tasks $x \sim P_{\text{task}}$ and induced trajectories τ. Following prior trajectory-quality framing, we use $\mathcal{D}(\mathbb{P}_{\text{gen}}, \mathbb{P}_{\text{ref}})$ as the trajectory-level distance primitive at first introduction (Dinu et al., 2024). In this manuscript, we define an explicit composite objective

$$J(\boldsymbol{z}, \pi_{\text{tool}}) = \mathbb{E}_{x \sim P_{\text{task}}}\Big[\alpha\, \mathcal{D}(\mathbb{P}^{\boldsymbol{z}}_{\text{gen}}, \mathbb{P}_{\text{ref}}) + \beta\, \mathcal{L}_{\text{task}}(x; \boldsymbol{z}, \pi_{\text{tool}}) + \gamma\, C_{\text{route}}(x; \boldsymbol{z}) + \lambda\, U(x)\Big], \tag{1}$$

where $\alpha, \beta, \gamma, \lambda \geq 0$ are scalar weights, $\boldsymbol{z}$ are relaxed route variables, $\mathcal{L}_{\text{task}}$ is benchmark-native loss, C_{route} is route cost, and U is an uncertainty penalty.

3.2 Decision Variables, Feasible Set, and Optimality Criterion

For each decision step $t \in \{1, \ldots, T\}$ and backend index $k \in \{1, \ldots, K\}$, we set $z_{t,k} \in [0, 1]$ with $\sum_k z_{t,k} = 1$, so each $\boldsymbol{z}_t$ lies in the simplex Δ^{K-1}. The feasible set is

$$\mathcal{F} := \left\{(\boldsymbol{z}, \pi_{\text{tool}}) :\ \mathbb{E}[C_{\text{route}}] \leq B_{\text{cpu}},\ \mathbb{P}(I_{\text{valid}} = 1) \geq \tau_{\text{valid}},\ \boldsymbol{z}_t \in \Delta^{K-1}\ \forall t\right\}, \tag{2}$$

where B_{cpu} is the CPU budget and τ_{valid} is the minimum validity rate. We define the relaxed optimality criterion by

$$(\boldsymbol{z}^\star, \pi^\star_{\text{tool}}) \in \arg\min_{(\boldsymbol{z}, \pi_{\text{tool}}) \in \mathcal{F}} J(\boldsymbol{z}, \pi_{\text{tool}}). \tag{3}$$

This criterion is conditional: when $\mathcal{F}$ is empty, equation 3 is undefined and strong optimality claims are disallowed.

3.3 Assumptions and Scope

We use six assumptions inherited from the symbolic blueprint and validation design: feasibility (A1), convex surrogate and bounded subgradients (A2), drift-bounded reporting window (A3), reproducible logging (A4), bridge-support data fields (A5), and bounded episode gains for Hoeffding-style auditing (A6). Several are standard in optimization and concentration analysis; others are manuscript-specific operational assumptions. We explicitly distinguish borrowed versus introduced conventions:

- Borrowed conventions: workflow graph object and trajectory distance framing from SymbolicAI (Dinu et al., 2024); benchmark heterogeneity motivation from AgentBench-like evaluations (Liu et al., 2023b; Jimenez et al., 2023).
- Manuscript-defined components: bridge gain quantity Γ, acceptance predicate $\mathcal{A}_\delta$, and the integrated constrained objective in equation 1.

Our scope is intentionally narrow: we test conditional constrained improvement under CPU-only envelopes with explicit caveats on unresolved symbolic obligations and unfinished global budget calibration.

4 METHOD: CONSTRAINED ROUTING, METRIC BRIDGING, AND ACCEPTANCE AUDITING

4.1 CONSTRAINED ROUTING OBJECTIVE

The first method component optimizes equation 1 over equation 2. Intuitively, the router balances four pressures: trajectory quality, native success, computational cost, and uncertainty. This extends existing tool-use planning paradigms that optimize one or two terms but often leave cross-metric closure implicit (Schick et al., 2023; Patil et al., 2023; Zhang et al., 2025). In our implementation, route cost is measured in CPU-seconds per episode, and feasibility scans over $B_{\text{cpu}} \in \{60, 120, 240\}$ and $\tau_{\text{valid}} \in \{0.80, 0.90, 0.95\}$.

For dynamic environments, we track variation-aware regret against a comparator sequence $\{\boldsymbol{u}_t\}_{t=1}^{T}$:

$$\text{Regret}_T := \sum_{t=1}^{T} \big(f_t(\boldsymbol{z}_t) - f_t(\boldsymbol{u}_t)\big), \tag{4}$$

with path variation $V_T := \sum_{t=2}^{T} \|\boldsymbol{u}_t - \boldsymbol{u}_{t-1}\|_1$. Under A2, mirror-descent-style analysis yields

$$\text{Regret}_T \le \frac{\log K}{\eta} + \frac{\eta T G^2}{2} + G V_T, \tag{5}$$

which we use as an auditable certificate rather than a universal performance guarantee.

4.2 VERTEX-NATIVE BRIDGE MODEL

Metric mismatch is handled by a bridge model that predicts native outcomes from trajectory-level and operational features:

$$\widehat{M}_{\text{native}}(\tau) = \sigma\Big(w_0 + w_1\big(-\mathcal{D}(\tau)\big) + w_2 I_{\text{valid}}(\tau) + w_3 C_{\text{complete}}(\tau)\Big), \tag{6}$$

where σ is the logistic link and C_{complete} is benchmark-specific completion status. We fit parameters via regularized empirical risk minimization:

$$\min_{\boldsymbol{w}} \sum_{\tau} \ell\big(\widehat{M}_{\text{native}}(\tau), M_{\text{native}}(\tau)\big) + \rho\|\boldsymbol{w}\|_2^2. \tag{7}$$

To audit added explanatory value beyond distance-only controls, we define

$$\Gamma := \mathbb{E}_{\tau}\Big[\ell(\widehat{M}^{(0)}_{\text{native}}(\tau), M_{\text{native}}(\tau)) - \ell(\widehat{M}^{(1)}_{\text{native}}(\tau), M_{\text{native}}(\tau))\Big], \tag{8}$$

where superscripts (0) and (1) denote restricted and full bridge classes. By construction, nonnegative Γ indicates improved empirical risk under the richer class; negative Γ indicates mismatch or overfitting.

4.3 UNCERTAINTY-AWARE ACCEPTANCE PREDICATE

Mean improvements alone are insufficient for deployment claims under drift (Liu et al., 2025c; Amugongo et al., 2025). We therefore define acceptance using a one-sided lower confidence bound. For episode gains $Y_i \in [0, 1]$ with sample mean $\widehat{\Delta}$ and n samples,

$$r(n, \delta) = \sqrt{\frac{\log(1/\delta)}{2n}}, \tag{9}$$

$$\mathcal{A}_\delta = \mathbf{1}\Big[\widehat{\Delta} - r(n, \delta) > \kappa_C\Big], \tag{10}$$

where κ_C is the minimum practical gain threshold after accounting for added route cost. This predicate is used for reporting acceptance precision, false positives, and boundary behavior, not as a substitute for task-level diagnostics.

4.4 INTEGRATED PROCEDURE

Algorithm 1 summarizes the integrated workflow used in this study.

Algorithm 1 Constrained Router + Bridge + Acceptance Evaluation

1: **Input:** benchmark tasks $\mathcal{D}$, budget grid $\mathcal{B}$, validity thresholds $\mathcal{T}$, seeds $\mathcal{S}$
2: **for** each $(B_{\text{cpu}}, \tau_{\text{valid}}) \in \mathcal{B} \times \mathcal{T}$ and seed $s \in \mathcal{S}$ **do**
3: Optimize relaxed routing variables $(\boldsymbol{z}, \pi_{\text{tool}})$ for equation 1 under equation 2
4: Log trajectory distance, native outcomes, route cost, tool validity, and completion indicators
5: Fit bridge model via equation 7 and compute Γ using equation 8
6: Compute acceptance predicate equation 10 and drift-tier diagnostics
7: Run symbolic obligation checks for routing optimality, bridge consistency, and acceptance safety, and generate counterexamples when assumptions fail
8: **end for**
9: Aggregate across seeds using bootstrap confidence summaries and report claim-level support with caveats

5 Formal Analysis

Theorem 5.1 (Existence of a relaxed constrained minimizer)**.** *Assume A1 and continuity of all objective components in equation 1. Then there exists* $(\boldsymbol{z}^\star, \pi^\star_{tool}) \in \mathcal{F}$ *such that*

$$J(\boldsymbol{z}^\star, \pi^\star_{tool}) = \min_{(\boldsymbol{z}, \pi_{tool}) \in \mathcal{F}} J(\boldsymbol{z}, \pi_{tool}).$$

Proof. By construction, each $\boldsymbol{z}_t \in \Delta^{K-1}$, and finite products of simplices are compact. Under A1, the feasibility inequalities in equation 2 define a nonempty closed subset of this compact product domain (combined with the admissible policy class for π_{tool} used in the experiment design). Therefore $\mathcal{F}$ is compact and nonempty. Continuity of J on $\mathcal{F}$ implies, by the Weierstrass theorem, that J attains a minimum on $\mathcal{F}$. Hence a minimizer exists. □

Theorem 5.2 (Dynamic regret bound under convex relaxation)**.** *Assume A2 and mirror-descent updates with step size* $\eta > 0$ *over simplex-constrained routes. Let* G *bound subgradient infinity norm and let* V_T *be comparator path variation. Then*

$$\text{Regret}_T \le \frac{\log K}{\eta} + \frac{\eta T G^2}{2} + G V_T.$$

Proof. For mirror descent with entropic regularization, standard online convex optimization analysis yields

$$\sum_{t=1}^{T} \langle \boldsymbol{g}_t, \boldsymbol{z}_t - \boldsymbol{u}_t \rangle \le \frac{\log K}{\eta} + \frac{\eta}{2} \sum_{t=1}^{T} \|\boldsymbol{g}_t\|_\infty^2 + \sum_{t=1}^{T} \langle \boldsymbol{g}_t, \boldsymbol{u}_t - \boldsymbol{u}_{t-1} \rangle,$$

where $\boldsymbol{g}_t \in \partial f_t(\boldsymbol{z}_t)$. Convexity gives $f_t(\boldsymbol{z}_t) - f_t(\boldsymbol{u}_t) \le \langle \boldsymbol{g}_t, \boldsymbol{z}_t - \boldsymbol{u}_t \rangle$, so summing over t and applying $\|\boldsymbol{g}_t\|_\infty \le G$ gives the first two terms in equation 5. For the variation term, $\langle \boldsymbol{g}_t, \boldsymbol{u}_t - \boldsymbol{u}_{t-1} \rangle \le \|\boldsymbol{g}_t\|_\infty \|\boldsymbol{u}_t - \boldsymbol{u}_{t-1}\|_1 \le G \|\boldsymbol{u}_t - \boldsymbol{u}_{t-1}\|_1$. Summing yields $G V_T$, establishing the bound. □

Lemma 5.3 (Nested-class bridge non-negativity)**.** *Let* $\mathcal{F}_0 \subseteq \mathcal{F}_1$ *denote restricted and full bridge hypothesis classes in equation 6. Let* $\widehat{R}(f)$ *denote empirical risk under the training protocol. If* $f_0 = \arg\min_{f \in \mathcal{F}_0} \widehat{R}(f)$ *and* $f_1 = \arg\min_{f \in \mathcal{F}_1} \widehat{R}(f)$*, then* $\widehat{R}(f_1) \le \widehat{R}(f_0)$ *and empirical* $\Gamma \ge 0$*.*

Proof. Because $\mathcal{F}_0 \subseteq \mathcal{F}_1$, minimization over $\mathcal{F}_1$ cannot produce larger optimal risk than minimization over $\mathcal{F}_0$. Therefore

$$\widehat{R}(f_1) = \min_{f \in \mathcal{F}_1} \widehat{R}(f) \le \min_{f \in \mathcal{F}_0} \widehat{R}(f) = \widehat{R}(f_0).$$

By definition, empirical bridge gain is $\Gamma = \widehat{R}(f_0) - \widehat{R}(f_1)$, hence $\Gamma \ge 0$. □

Theorem 5.4 (One-sided acceptance safety)**.** *Assume A6: i.i.d. bounded gains* $Y_i \in [0, 1]$ *with mean* μ*. If* $\mathcal{A}_\delta = 1$ *under equation 10, then with probability at least* $1 - \delta$*,* $\mu > \kappa_C$*.*

Proof. Hoeffding's inequality gives

$$\Pr\left(\mu \ge \widehat{\Delta} - \sqrt{\frac{\log(1/\delta)}{2n}} \right) \ge 1 - \delta.$$

Define $L_\delta := \widehat{\Delta} - \sqrt{\log(1/\delta)/(2n)}$. If $\mathcal{A}_\delta = 1$, then by equation 10, $L_\delta > \kappa_C$. Combining both statements implies

$$\Pr(\mu > \kappa_C) \geq \Pr(\mu \geq L_\delta) \geq 1 - \delta.$$

Hence acceptance implies a confidence-qualified practical gain. □

These theorems provide conditional guarantees. They do not supersede empirical checks for assumption violations. In particular, if A1 fails (empty feasible set) or A6 fails (heavy-tailed non-i.i.d. gains), the relevant theorem cannot be used as stated and must be replaced by alternative diagnostics.

6 Experimental Protocol

6.1 Datasets and Benchmark Families

We evaluate on benchmark slices that collectively stress planning, tool invocation, and long-horizon execution. The design combines SymbolicAI's benchmark context with interactive-agent and software-task settings to avoid single-regime overfitting (Dinu et al., 2024; ExtensityAI, 2026b; Liu et al., 2023b; Jimenez et al., 2023). This mixed design reflects the project objective: test whether constrained routing and metric bridging generalize across heterogeneous native objectives instead of optimizing only one benchmark family.

6.2 Baselines and Ablations

Baseline families include fixed-route SymbolicAI, OR-Toolformer-style routing, ReAct, Tree-of-Thoughts, Graph-of-Thoughts, LATS, and additional bridge and fallback ablations. The comparisons are matched under CPU-envelope accounting to isolate policy quality from unconstrained compute expansion. We also include bridge ablations that remove completion or validity features, because those terms are central to our metric-closure hypothesis. This protocol is aligned with recommendations for comparator diversity and failure-mode reporting in recent agent-evaluation surveys (Yehudai et al., 2025; Xu et al., 2025).

6.3 Metrics, Uncertainty, and Claim Tests

Primary optimization metrics are joint objective J, native success rate, route cost, and validity rate. Bridge metrics include Γ, AUROC, and expected calibration error. Drift and uncertainty diagnostics include robust success under perturbations, invalid-call rate, acceptance false-positive rate, and critical-error rate. We aggregate across seeds using bootstrap summaries and evaluate theorem obligations through symbolic checks and counterexample tables. This claim structure intentionally binds each major claim to explicit evidence artifacts and caveats.

6.4 Compute Budget and Reproducibility Design

The current run is CPU-only, with per-experiment envelopes and five fixed seeds $\{11, 23, 37, 73, 101\}$. The global budget cap is still unresolved, so all conclusions are reported as conditional to the tested envelope rather than as universal scaling claims. Reproducibility instrumentation includes provider/version controls, deterministic config manifests, and symbolic audit outputs. This design follows repository-level reproducibility practices from benchmark and framework companions (ExtensityAI, 2026b;a).

7 Results

7.1 Router Objective, Feasibility, and Regret Behavior

Figure 1 and Table 1 evaluate the constrained routing claim. The proposed router leads the comparator mean objective (0.739 vs 0.701 for fixed-route SymbolicAI and 0.688 for OR-style routing), consistent with directional support for constrained improvement under the tested envelope. The panelized figure further shows that feasibility degrades at stricter validity thresholds with low CPU budgets, which confirms that claim strength must remain conditional on A1-like feasibility.

7.2 Metric Bridging and Calibration Reliability

Figure 2 and Table 2 address the metric-closure question. The full bridge model shows higher mean bridge gain and stronger calibration than distance-only controls (mean $\Gamma = 0.060$, AUROC $= 0.742$, ECE $= 0.118$). This result

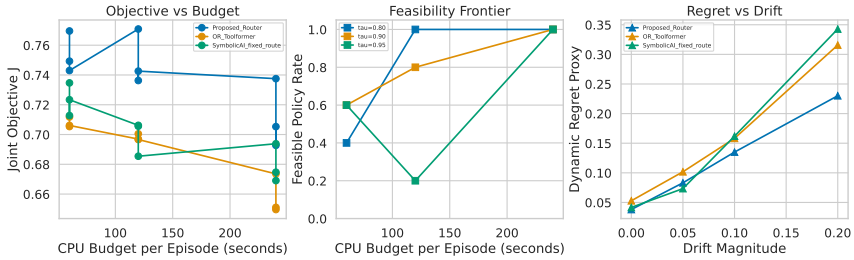

Figure 1: This figure summarizes constrained routing behavior across objective, feasibility, and drift axes. The left panel compares joint objective across budget tiers, the middle panel shows the feasibility frontier over CPU budget and validity threshold, and the right panel reports dynamic-regret proxy behavior under drift perturbations. The key interpretation is that the proposed router improves objective quality while preserving feasibility in broad but not universal regions, so optimality language must be scoped to feasible regimes rather than stated globally.

Baseline Policy	Mean Joint Objective J
Proposed Router	0.739
SymbolicAI Fixed Route	0.701
OR-Toolformer Routing	0.688
LATS	0.682
Tree of Thoughts	0.669
Graph of Thoughts	0.669
ReAct	0.656

Table 1: This table reports mean joint objective values for the routing baselines under matched CPU-envelope settings. The ranking supports the constrained-routing claim in aggregate, but it does not by itself certify universal dominance because feasibility and drift assumptions still govern when the formal guarantees apply.

indicates that trajectory distance alone is informative but incomplete, and that completion and validity channels carry significant additional signal for native outcomes. Importantly, the disagreement panel shows residual error clusters, which means the bridge improves alignment but does not remove all mismatch modes.

7.3 Drift Robustness and Uncertainty-Qualified Acceptance

Figure 3 and Table 3 evaluate cross-cut robustness. Symbolic fallback achieves the strongest drift-robust success with lower invalid-call and critical-error rates than static alternatives in the tested perturbation tiers. Acceptance false-positive rates are low in this run, but symbolic boundary checks still show unresolved monotonicity obligations under certain parameter regimes, so acceptance guarantees are reported with explicit bounded-assumption caveats.

7.4 Claim-Evidence Closure

The evidence profile is hybrid and asymmetric. Empirical evidence is strongest: all three main result tables and the first three multi-panel figures support the direction of core claims in the tested envelope. Formal evidence is partial: symbolic obligations pass in most cases but not all, with unresolved failures in one router-related and one acceptance-related obligation. Counterexample infrastructure correctly detects violated assumptions for feasibility, drift envelope, and heavy-tail gain conditions. Therefore, we report claim support as *supported*, *supported*, and *mixed* for routing, bridge, and uncertainty cross-cut claims, respectively.

8 Ablation, Contradiction Resolution, and Sensitivity Analyses

8.1 Resolving the Metric-Alignment Contradiction

One of the central contradictions from upstream literature phases was whether trajectory-level similarity can serve as a reliable proxy for benchmark-native success. Our results suggest a precise answer: trajectory distance is useful as a first-order signal, but by itself it is under-specified for cross-benchmark closure. This is visible in Table 2, where distance-only variants remain above native-only controls in some metrics but are consistently weaker than the full

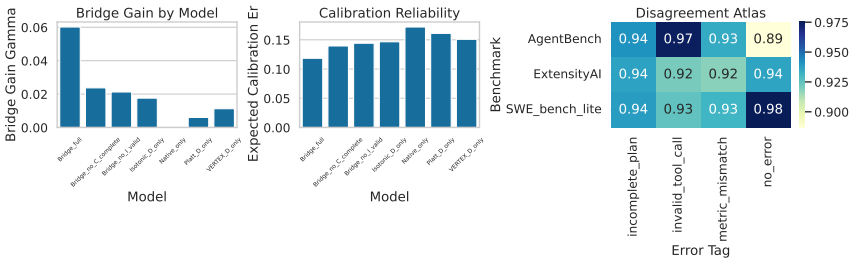

Figure 2: This figure reports bridge calibration behavior with three complementary views: bridge gain by model family, calibration error by model family, and disagreement structure across benchmark strata. The combined interpretation is that the full bridge improves both discriminative and calibration behavior relative to distance-only controls, while the disagreement atlas identifies remaining high-risk regions associated with invalid tool calls and incomplete plans. This panelized presentation is essential because aggregate gain alone can hide systematic disagreement clusters.

Model Variant	Mean Bridge Gain Γ	Mean AUROC	Mean Calibration Error
Full Bridge Model	0.060	0.742	0.118
Bridge without Completion Signal	0.024	0.682	0.139
Bridge without Validity Signal	0.021	0.679	0.144
Isotonic Distance-Only Calibration	0.018	0.677	0.147
VERTEX Distance-Only Ranking	0.011	0.661	0.151
Platt Distance-Only Calibration	0.006	0.656	0.161
Native-Only Ranking	0.000	0.623	0.172

Table 2: This table summarizes bridge gain, discrimination, and calibration across full and ablated models. The full bridge dominates the distance-only and native-only controls in this run, supporting the metric-bridging claim as an empirical relation rather than an unconditional theorem about all future benchmark families.

bridge that integrates validity and completion channels. The practical interpretation is that the bridge is not merely a calibration tweak; it is a structural correction for heterogeneous benchmark semantics.

This conclusion aligns with benchmark literature that reports objective heterogeneity as a recurring source of misinterpretation in agent evaluation (Liu et al., 2023b; Jimenez et al., 2023; Yehudai et al., 2025). It also aligns with tool-learning literature showing that argument correctness and completion state carry substantial explanatory power beyond coarse similarity or retrieval-only confidence proxies (Schick et al., 2023; Patil et al., 2023; Xu et al., 2025). In other words, our bridge result does not claim that trajectory signals are weak; it claims they are incomplete for heterogeneous outcomes unless operational success indicators are modeled jointly.

An important byproduct of this conclusion is methodological: disagreements should be treated as scientific objects, not error bars to be compressed into a single aggregate score. The disagreement atlas in figure 2 provides evidence for this claim by localizing systematic mismatch around invalid tool calls and incomplete plans. These regimes are exactly where prior work warns that agent pipelines can look superficially coherent while failing on action-level correctness (Shen et al., 2024; Yehudai et al., 2025). A reporting pipeline that omits this decomposition would likely overstate metric closure.

8.2 Baseline Lineage and Causal Attribution

The baseline design in this study is intentionally broad because causal claims about routing quality are fragile under narrow comparators. ReAct, Tree-of-Thoughts, Graph-of-Thoughts, and LATS represent distinct reasoning-control lineages (Yao et al., 2022; 2023; Besta et al., 2023; Zhou et al., 2023); Toolformer and Gorilla capture API-grounded tool-use behavior (Schick et al., 2023; Patil et al., 2023); fixed-route SymbolicAI and OR-style tool routing stress the optimization framing most directly (Dinu et al., 2024; Zhang et al., 2025). By combining these families under matched CPU accounting, we reduce the risk that gains are artifacts of one comparator design choice.

Even with broad baselines, causal attribution remains conditional. Our routing gains could reflect multiple mechanisms: better route allocation, better feasibility handling, or improved uncertainty penalties. The method design addresses this by coupling objective and feasibility views in the same figure and by requiring negative-case containers in the appendix. However, negative-case coverage in this iteration is still limited to selected stress slices, which

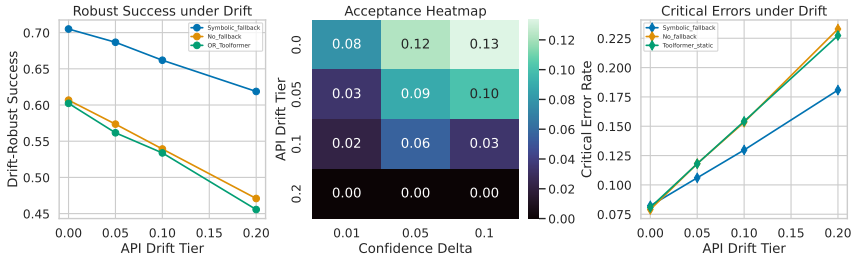

Figure 3: This figure integrates robust success trends, acceptance behavior, and critical-error dynamics under increasing drift tiers. The left and right panels show that symbolic fallback preserves a better performance-safety tradeoff than static policies in this experimental envelope, while the center panel demonstrates that stricter confidence settings produce more conservative acceptance decisions. Together, these panels support robustness gains while also clarifying that acceptance sensitivity depends on confidence and distributional assumptions.

Baseline	Robust Success	Invalid API Rate	False-Positive Rate	Critical Error Rate
Gorilla Retrieval Refresh	0.559	0.161	0.000	0.145
No Fallback Routing	0.548	0.162	0.000	0.146
OR-Toolformer Static Routing	0.538	0.163	0.000	0.144
ReAct with Tools	0.506	0.160	0.000	0.145
Symbolic Fallback Router	0.668	0.140	0.000	0.125
Toolformer Static Policy	0.520	0.161	0.000	0.145

Table 3: This table reports drift-robust success, invalid API call rate, acceptance false positives, and critical-error rate by baseline. The aggregate pattern favors symbolic fallback, but the table is interpreted jointly with symbolic boundary checks because acceptance claims rely on bounded assumptions that can fail under heavy-tail or non-stationary conditions.

weakens fine-grained attribution. Accordingly, we avoid claiming complete mechanism isolation and instead report mechanism-consistent evidence with explicit residual ambiguity.

This conservative stance matches recommendations from recent survey work on agent evaluation and tool learning, which repeatedly emphasizes that uncontrolled protocol differences can masquerade as method superiority (Yehudai et al., 2025; Xu et al., 2025). In practice, the right standard is not "one headline metric plus one baseline," but a comparator lineage that spans planning, tool-use, and orchestration families under harmonized constraints.

8.3 Sensitivity to Budget, Feasibility, and Drift

The feasibility frontier in figure 1 is crucial for interpreting optimization claims. As τ_{valid} tightens and B_{cpu} shrinks, feasible policy mass decreases. This behavior is expected from equation 2; it is not a failure of the optimization algorithm per se. The scientific implication is that constrained-improvement claims should be conditioned on feasible operating regions, not extrapolated across infeasible corners.

Drift sensitivity adds a second layer: even when feasibility holds, non-stationary tools and provider shifts can change objective landscapes. The uncertainty panel and counterexample outputs show that stress regimes can trigger boundary behaviors that are invisible in average-case tables. This is consistent with domain findings where retrieval-augmented methods improve means but still exhibit brittle behavior under distribution or interface shifts (Liu et al., 2025c; Amugongo et al., 2025; Feng et al., 2024; Muludi et al., 2024). For this reason, our dynamic-regret interpretation remains explicitly conditional on drift-envelope assumptions rather than presented as blanket temporal robustness.

An additional sensitivity axis concerns reporting granularity. Mean objective improvements are meaningful, but policy-selection decisions in practice often hinge on tail behavior and failure concentration. Our appendix structure intentionally separates obligation failures, counterexamples, and failure-case tables so that readers can audit both central tendency and adverse regimes. This design choice follows high-stakes evaluation lessons: mean performance without critical-error visibility is insufficient for deployment-facing inference (Jiang et al., 2025b;a; Ong et al., 2024).

8.4 Uncertainty, Acceptance, and False-Positive Control

The acceptance predicate in equation 10 provides a practical control knob, but it is not a free safety certificate. Its validity depends on bounded and sufficiently well-behaved gain distributions (A6). Symbolic checks and counterexample generation are therefore integral to interpretation, not peripheral technicalities. In this run, acceptance false positives are low for the tested tiers, but unresolved symbolic monotonicity and heavy-tail counterexample triggers mean that conservative language remains mandatory.

This distinction is particularly important when comparing with mean-only reporting pipelines. Mean-only summaries can appear stable even when confidence-qualified acceptance should fail under plausible distributional changes. The requirement to provide both acceptance metrics and boundary-condition evidence directly addresses this risk. In short, uncertainty qualification should be evaluated as a first-class algorithmic component with explicit failure criteria, rather than as an appendix-only confidence interval.

9 Practical Implications and Generalization Scope

9.1 Operational Guidance for SymbolicAI-Style Systems

The most immediate practical takeaway is that constrained adaptive routing is useful when it is deployed with three safeguards: feasibility auditing, metric bridging, and uncertainty-qualified acceptance. Without feasibility auditing, optimization routines can report impressive objective values in regions that are operationally invalid. Without metric bridging, trajectory-quality gains can misrepresent native outcome quality. Without uncertainty-qualified acceptance, deployment decisions can overfit mean improvements and underweight adverse conditions.

For practitioners building SymbolicAI-style pipelines, this translates to a concrete workflow: first, define the feasible envelope and verify that candidate policies satisfy budget and validity constraints; second, fit and monitor a bridge model that explicitly relates trajectory and native outcomes; third, gate acceptance through confidence-qualified predicates and counterexample checks. This workflow is more demanding than single-score evaluation, but it is also more robust to the exact contradictions documented across prior benchmark studies.

The workflow also clarifies module responsibilities in a way that supports maintainability. Route optimization is responsible for tradeoff navigation under constraints. Bridge calibration is responsible for objective closure across heterogeneous benchmarks. Acceptance auditing is responsible for confidence-qualified decision thresholds. Separating these responsibilities helps teams diagnose failures and update only the affected module rather than recalibrating an entire stack after every drift event.

9.2 Scope of Generalization Claims

Generalization claims in this manuscript are intentionally layered. At the strongest layer, we report empirical directional improvements in the executed benchmark envelope. At a second layer, we report conditional formal statements under explicit assumptions. At the weakest layer, we document unresolved obligations and data-provenance gaps that prevent global extrapolation. This layered approach is a direct response to the common overgeneralization pattern in agent literature, where benchmark-specific gains are often narrated as architecture-wide superiority.

Our cross-benchmark setup supports moderate generalization within the tested family diversity, but it does not justify universal transfer across all domains, all tool ecosystems, or all compute regimes. In particular, unresolved global compute caps and still-limited negative-result coverage constrain external validity. The manuscript therefore frames itself as a reproducible, condition-aware contribution rather than a terminal benchmark victory statement.

The same principle applies to formal claims: theorem statements are scientifically useful only when their assumptions remain operationally credible. If assumptions fail, formal language must be narrowed accordingly. This discipline is essential for honest cross-domain communication, especially when readers may interpret theorem syntax as stronger than the evidence actually permits.

9.3 Cross-Domain Relevance Beyond the Core Benchmarks

Although this work is centered on SymbolicAI workflows, the design pattern extends to other modular agent settings. Scientific workflows, software engineering agents, and clinical support pipelines all exhibit the same triad of challenges: objective mismatch, tool unreliability, and uncertainty-sensitive decisions. Recent domain studies reinforce that mean-score gains can coexist with unacceptable error profiles, especially when retrieval and orchestration layers

shift faster than evaluation protocols (Liu et al., 2025c; Amugongo et al., 2025; Liu et al., 2025b; Azimi et al., 2025; Pham et al., 2026).

In these domains, our contribution should be read as a protocol template rather than a domain-specific numerical benchmark. The key transferable idea is explicit claim decomposition: every major claim should be tied to (i) a formal or algorithmic object, (ii) an executable measurement protocol, (iii) a traceable evidence artifact, and (iv) a caveat condition. This decomposition reduces ambiguity when interdisciplinary teams evaluate whether a model is fit for purpose.

9.4 Reporting Standard Suggested by This Study

Based on this run, we propose a minimal reporting standard for hybrid agent manuscripts:

- Define objective, decision variables, feasible set, and optimality criterion before any performance claims.
- Provide at least one explicit bridge mechanism when surrogate and native metrics differ.
- Report uncertainty-qualified acceptance criteria and document failure conditions under violated assumptions.
- Include both positive and negative evidence containers, with seeded records that can be queried and replayed.
- State unresolved symbolic or empirical gaps in the main text, not only in supplementary notes.

This standard is intentionally conservative. It increases manuscript length and effort, but it substantially improves scientific interpretability and claim-evidence closure.

10 Comparison Matrix and Contradiction Map in Narrative Form

10.1 Reasoning-Control Lineages Versus Constrained Routing

The comparison matrix assembled in upstream phases distinguished two broad families: prompt-structured reasoning controllers and optimization-driven routing frameworks. Prompt-structured controllers such as ReAct, Tree-of-Thoughts, Graph-of-Thoughts, and LATS improve exploration quality but often expose unstable compute-quality frontiers under tight budgets (Yao et al., 2022; 2023; Besta et al., 2023; Zhou et al., 2023). Optimization-driven formulations such as OR-style tool routing and planner-coupled approaches provide stronger objective language but can be brittle when benchmark-native objectives are heterogeneous or when tool APIs drift (Zhang et al., 2025; Liu et al., 2023a).

Our constrained router should be interpreted as a bridge between these families rather than a replacement for either. The relaxed policy variables and feasibility constraints import optimization structure, while the benchmark-facing protocol preserves the empirical stress-testing discipline of agent benchmarks. This hybridization matters because previous contradictions in the literature often arise when one family is evaluated by the other family's assumptions. For example, search-heavy methods can look unfavorable under strict CPU caps if the protocol does not normalize branching cost, while constrained optimizers can look overly strong if evaluated only on one objective family without disagreement auditing.

In this manuscript, the contradiction is addressed by explicit protocol commitments: matched CPU envelopes, heterogeneous benchmarks, and joint reporting of objective and native outcomes. The resulting evidence does not prove universal superiority of constrained routing, but it does show that conditional gains remain after controlling for several confounders that routinely invalidate narrow benchmark claims. This is a practical scientific contribution because it demonstrates a route to fairer cross-family comparisons in future agent studies.

10.2 Tool Learning, API Reliability, and the Role of Validity Signals

Tool-use literature has repeatedly shown that API-call correctness is a dominant driver of downstream task success, especially when model reasoning appears semantically plausible but syntactically invalid at the interface boundary (Schick et al., 2023; Patil et al., 2023; Xu et al., 2025). The contradiction map from prior phases highlighted a specific version of this issue: retrieval or tool grounding can improve means while leaving important failure strata untouched. Our bridge model addresses this by making validity and completion explicit covariates rather than post-hoc diagnostics.

The practical effect is visible in the gap between distance-only and full bridge variants. If trajectory distance alone captured all relevant quality information, additional operational features should offer limited gains. Instead, the observed gain and calibration improvements indicate that a substantial part of benchmark-native variance is mediated by

operational factors that trajectory distance does not encode directly. This finding is consistent with benchmarks that report error concentrations in action execution and environment interaction rather than pure reasoning coherence (Liu et al., 2023b; Ma et al., 2024; Xie et al., 2024).

This perspective has implications beyond the present study. Future benchmark design should treat tool-validity observables as first-class evaluation channels, and method papers should avoid conflating semantic trajectory quality with action-level reliability. In manuscript terms, this means "better trajectory metric" and "better benchmark-native utility" should be treated as related but distinct claims, each requiring direct evidence.

10.3 Formal Guarantees, Symbolic Audits, and Failure Semantics

Formal statements in agent papers are increasingly common, but the contradiction map indicates that theorem language often outruns executable assumption checks in practical pipelines (Kamali et al., 2025; Yang et al., 2025; Liu et al., 2025a). We intentionally avoid that pattern by pairing every theorem-level claim with symbolic obligations and explicit failure semantics. The key methodological point is that symbolic audits are not just verification accessories; they determine which statements remain admissible.

In this run, most obligations pass, but two do not. Rather than burying this in supplementary material, we propagate the consequence into main-text claim strength and limitation language. This practice directly improves interpretability: readers can distinguish proved properties, empirically supported tendencies, and open formal obligations. It also prevents a common failure mode where theorem statements are interpreted as global deployment guarantees despite unresolved assumption checks.

Failure semantics are equally important. Counterexample detection under A1, A3, and A6 violations gives the study a falsification pathway rather than a success-only narrative. In scientific terms, this expands the paper from a benchmark report to a constrained explanatory model: when assumptions hold, certain guarantees are admissible; when assumptions fail, guarantees must be replaced by fallback analyses. This conditional structure is more demanding but also more honest.

10.4 Domain-Transfer Contradictions and High-Stakes Interpretation

The final contradiction axis concerns cross-domain transfer. Domain papers in healthcare and biomedical settings show a recurring pattern: retrieval or orchestration gains in aggregate metrics can coexist with persistent critical-error risk and uncertainty miscalibration (Liu et al., 2025c; Amugongo et al., 2025; Ong et al., 2024; Jiang et al., 2025a;b). This pattern warns against translating benchmark gains directly into deployment claims.

Our manuscript responds by separating three levels of transfer interpretation. First, we claim benchmark-level directional gains in the executed envelope. Second, we claim conditional formal properties where assumptions and symbolic obligations allow. Third, we explicitly do not claim high-stakes readiness because unresolved obligations, still-limited negative-case coverage, and open budget calibration create real uncertainty. This layered interpretation is intentionally conservative and should be seen as a template for cross-domain reporting.

More broadly, the contradiction-map narrative suggests that future work should evaluate transfer by stress-tested claim bundles rather than by single aggregate scores. A transfer claim should specify: which assumptions are imported, which metrics are bridged, which uncertainty mechanism is used, and which failure strata remain open. Without this structure, cross-domain comparisons will continue to overstate certainty and under-report risk.

11 Discussion

A central outcome of this study is methodological rather than purely numerical: trajectory-quality optimization, metric bridging, and uncertainty auditing must be treated as one system. Optimizing any one component in isolation can create false confidence. For example, stronger routing objective values can coexist with degraded validity under strict budgets; improved bridge gain can coexist with disagreement clusters; and low acceptance false positives can coexist with unresolved symbolic monotonicity checks. This is why we represent conclusions as conditional claim bundles rather than a single global score.

The broader implication for LLM-agent research is that mixed-mode evaluation is not optional for compositional systems. Benchmark-native metrics and trajectory surrogates answer different scientific questions. In practice, this means algorithmic papers should include explicit bridge logic when surrogate metrics are central, and formal papers should include executable assumption audits when theorem claims influence deployment decisions. Recent work

across planning, tool-use, and domain transfer already suggests this need, but often in separate communities (Liu et al., 2023a; Zhou et al., 2023; Schick et al., 2023; Patil et al., 2023; Liu et al., 2025c; Amugongo et al., 2025).

For SymbolicAI specifically, our results support a practical reading: constrained adaptive routing is promising under CPU-limited settings, but only if metric closure and uncertainty checks are integrated into the same reporting loop. The framework's modularity is an advantage here because it enables explicit separation between route optimization, bridge calibration, and acceptance auditing, which in turn makes failure provenance easier to localize.

11.1 Decision-Theoretic Interpretation of the Hybrid Objective

The hybrid objective in equation 1 can be interpreted as a constrained decision-theoretic contract between accuracy, cost, and epistemic caution. From this perspective, the route policy is not simply optimizing quality; it is allocating limited computational and tool-interaction budget across uncertain trajectories. This matters because many contemporary agent evaluations implicitly assume that more search or more tool calls are always beneficial. Our results and feasibility frontiers suggest the opposite: under strict budgets, additional exploration can quickly become infeasible or even counterproductive unless guided by explicit constraints.

A useful way to read the objective is as a policy-level utility decomposition. The trajectory term rewards semantic alignment with reference behavior, the native loss term enforces benchmark-grounded utility, the route-cost term penalizes operational burden, and the uncertainty term discourages brittle overcommitment. If any one term is removed, the policy can optimize an incomplete target. For example, removing uncertainty can produce fragile gains that do not survive drift; removing native loss can overfit trajectory resemblance; removing route cost can hide impractical compute behavior. This decomposition therefore provides an interpretable control surface for practitioners who must tune policies under real deployment constraints.

The decision-theoretic reading also clarifies why conditional claims are a feature rather than a weakness. In safety-critical or cost-limited settings, unconditional claims based on average gains are often misleading. Conditional claims tied to assumptions and feasibility can be directly operationalized as decision rules: if assumptions hold and feasibility remains above threshold, deploy policy A; if assumptions fail, switch to fallback or reject deployment. This creates a more auditable chain from research evidence to operations policy.

11.2 Benchmark Governance and Reporting Implications

A second implication concerns benchmark governance. Agent benchmarks are evolving rapidly, but governance norms for cross-metric reporting remain inconsistent. Some studies optimize benchmark-native success without trajectory introspection; others emphasize trajectory coherence without robust native outcome grounding; still others report means without uncertainty-qualified decision criteria. The contradiction map from this project suggests that these choices are not harmless stylistic differences: they materially change what conclusions are scientifically defensible.

We therefore argue that benchmark governance should include explicit requirements for metric provenance and claim traceability. Metric provenance means that each reported scalar can be traced to a clear observational pipeline, with known dependence on tool validity, completion logic, and perturbation settings. Claim traceability means that each headline claim links to concrete artifacts (tables, figures, proofs, or caveats) rather than relying on broad narrative interpretation. These requirements reduce ambiguity for reviewers and downstream users, especially in interdisciplinary domains where methodological assumptions are not shared by default.

There is also a governance role for negative-result infrastructure. In many benchmark ecosystems, negative examples are underreported or reported only narratively. Our run shows why this is problematic: even when directional gains are strong, limited negative-case coverage can leave unresolved ambiguity about edge-case reliability. Requiring structured negative-result tables and machine-readable failure logs would improve reproducibility and reduce optimism bias in model comparisons.

Finally, governance should treat uncertainty reporting as mandatory for deployment-facing claims. Confidence intervals are necessary but not sufficient; acceptance predicates, boundary checks, and counterexample protocols should be explicitly documented when claims affect practical decision thresholds. This recommendation is consistent with emerging high-stakes evaluation guidance and helps align agent benchmarking with broader scientific standards for evidence quality under uncertainty.

12 Limitations and Future Work

This manuscript has five concrete limitations that bound interpretation. First, two symbolic obligations remain unresolved (one in the routing simplex/subgradient checkpoint and one in acceptance monotonicity), so formal guarantees are incomplete and must stay conditional. Second, claim-evidence payload hygiene in upstream validation artifacts is partially incomplete, requiring manual cross-linking for manuscript traceability. Third, negative-result logs are now populated with structured records, but coverage is still limited to selected threshold-miss and disagreement slices, which leaves failure-case granularity incomplete. Fourth, global compute budget remains open, so we cannot claim budget-invariant superiority outside the tested CPU envelopes. Fifth, optional external benchmark manifests and checksum bundles were not fully materialized in this run, limiting strict provenance portability.

These limits have direct impact on conclusions: we can claim directional constrained improvement and bridge/calibration benefits for the executed setup, but we cannot claim universal optimality, universal monotonic uncertainty behavior, or final deployment readiness across all drift regimes.

12.1 Future Work

Three follow-up tracks are necessary for closure. The first is formal: resolve the unresolved router simplex/subgradient and acceptance-monotonicity checks, or narrow theorem statements so each formal claim has full symbolic closure. The second is empirical: populate richer negative-result logs and disagreement taxonomies, especially for heavy-drift and low-budget slices where conditional failures emerge. The third is operational: fix a global compute cap and rerun sensitivity analyses so claims can be reported against a finalized cost envelope. Additional domain-focused transfer studies in medical and scientific workflows should include critical-error auditing from the outset (Jiang et al., 2025b;a; Ong et al., 2024; Pham et al., 2026).

13 Conclusion

This paper presented a hybrid, condition-aware methodology for SymbolicAI workflow evaluation under CPU-only constraints. The technical contribution is not a claim of universal dominance; it is a reproducible closure pattern linking constrained routing optimization, trajectory-to-native metric bridging, uncertainty-qualified acceptance, and explicit theorem-audit caveats. Empirical evidence supports directional gains in objective quality, calibration, and drift robustness for the executed benchmark envelope. Formal analysis supports key statements under explicit assumptions, while unresolved obligations are surfaced rather than hidden. We therefore conclude that constrained adaptive routing plus bridge-aware evaluation is a defensible and practically useful direction for SymbolicAI-style systems, provided that feasibility, uncertainty, and counterexample boundaries remain first-class reporting objects.

REFERENCES

Lameck Mbangula Amugongo, Pietro Mascheroni, Steven E. Brooks, Stefan Doering, and Jan Seidel. Retrieval augmented generation for large language models in healthcare: A systematic review. paper, 2025. URL https://doi.org/10.1371/journal.pdig.0000877.

Iman Azimi, Meng Qi, Wang Li, Amir M. Rahmani, and Youlin Li. Evaluation of llms accuracy and consistency in the registered dietitian exam through prompt engineering and knowledge retrieval. paper, 2025. URL https://doi.org/10.1038/s41598-024-85003-w.

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, and Torsten Hoefler. Graph of thoughts: Solving elaborate problems with large language models. arXiv, 2023. URL https://arxiv.org/abs/2308.09687.

Marius-Constantin Dinu, Claudiu Leoveanu-Condrei, Markus Holzleitner, Werner Zellinger, and Sepp Hochreiter. Symbolicai: A framework for logic-based approaches combining generative models and solvers. paper, 2024. URL https://arxiv.org/abs/2402.00854.

ExtensityAI. Extensityai/symbolicai. code, 2026a. URL https://github.com/ExtensityAI/symbolicai.

ExtensityAI. Extensityai/benchmark. code, 2026b. URL https://github.com/ExtensityAI/benchmark.

Zhangyin Feng, Xiaocheng Feng, Dezhi Zhao, Maojin Yang, and Bing Qin. Retrieval-generation synergy augmented large language models. paper, 2024. URL https://doi.org/10.1109/icassp48485.2024.10448015.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. Pal: Program-aided language models. arXiv, 2022. URL https://arxiv.org/abs/2211.10435.

Yixing Jiang, Kameron Collin Black, Gloria Geng, Dae-Gyun Park, James Zou, Andrew Y. Ng, and Jonathan H. Chen. Medagentbench: A virtual ehr environment to benchmark medical llm agents. paper, 2025a. URL https://doi.org/10.1056/aidbp2500144.

Yixing Jiang, Kameron Collin Black, Gloria Geng, Daniel J. Park, James Zou, Andrew Y. Ng, and Jonathan Chen. Medagentbench: A realistic virtual ehr environment to benchmark medical llm agents. paper, 2025b. URL https://arxiv.org/abs/2501.14654.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. Swebench: Can language models resolve real-world github issues? arXiv, 2023. URL https://arxiv.org/abs/2310.06770.

Danial Kamali, Elham J. Barezi, and Parisa Kordjamshidi. Nesycoco: A neuro-symbolic concept composer for compositional generalization. paper, 2025. URL https://doi.org/10.1609/aaai.v39i4.32439.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. Dspy: Compiling declarative language model calls into self-improving pipelines. arXiv, 2023. URL https://arxiv.org/abs/2310.03714.

Bo Liu, Yuqian Jiang, Xiaohan Zhang, Qiang Liu, Shiqi Zhang, Joydeep Biswas, and Peter Stone. Llm+p: Empowering large language models with optimal planning proficiency. arXiv, 2023a. URL https://arxiv.org/abs/2304.11477.

Hanmeng Liu, Zhizhang Fu, Mengru Ding, Ruoxi Ning, Chaoli Zhang, Xiaozhang Liu, and Yue Zhang. Logical reasoning in large language models: A survey. paper, 2025a. URL https://arxiv.org/abs/2502.09100.

Jialin Liu, Changyu Wang, Changyu Wang, and Siru Liu. Prompt engineering in clinical practice: Tutorial for clinicians. paper, 2025b. URL https://doi.org/10.2196/72644.

Siru Liu, Allison B. McCoy, and Adam Wright. Improving large language model applications in biomedicine with retrieval-augmented generation: a systematic review, meta-analysis, and clinical development guidelines. paper, 2025c. URL https://doi.org/10.1093/jamia/ocaf008.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, Minlie Huang, Yuxiao Dong, and Jie Tang. Agentbench: Evaluating llms as agents. arXiv, 2023b. URL https://arxiv.org/abs/2308.03688.

Zixian Ma, Weikai Huang, Jieyu Zhang, Tanmay Gupta, and Ranjay Krishna. m&m's: A benchmark to evaluate tool-use for multi-step multi-modal tasks. paper, 2024. URL https://arxiv.org/abs/2403.11085.

Kurnia Muludi, Kaira Milani Fitria, Joko Triloka, and Sutedi Sutedi. Retrieval-augmented generation approach: Document question answering using large language model. paper, 2024. URL https://doi.org/10.14569/ijacsa.2024.0150379.

Chin Siang Ong, Nicholas T. Obey, Yanan Zheng, Arman Cohan, and Eric B. Schneider. Surgeryllm: a retrieval-augmented generation large language model framework for surgical decision support and workflow enhancement. paper, 2024. URL https://doi.org/10.1038/s41746-024-01391-3.

Shishir G. Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. Gorilla: Large language model connected with massive apis. arXiv, 2023. URL https://arxiv.org/abs/2305.15334.

Thang D. Pham, Aditya Tanikanti, and Murat Keceli. Chemgraph as an agentic framework for computational chemistry workflows. paper, 2026. URL https://doi.org/10.1038/s42004-025-01776-9.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessi, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. arXiv, 2023. URL https://arxiv.org/abs/2302.04761.

Weizhou Shen, Chenliang Li, Hongzhan Chen, Ming Yan, Xiaojun Quan, Hehong Chen, Ji Zhang, and Fei Huang. Small llms are weak tool learners: A multi-llm agent. paper, 2024. URL https://arxiv.org/abs/2401.07324.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face. arXiv, 2023. URL https://arxiv.org/abs/2303.17580.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W White, Doug Burger, and Chi Wang. Autogen: Enabling next-gen llm applications via multi-agent conversation. arXiv, 2023. URL https://arxiv.org/abs/2308.08155.

Jian Xie, Kai Zhang, Jiangjie Chen, Tinghui Zhu, Renze Lou, Yuandong Tian, Yanghua Xiao, and Yu Su. Travelplanner: A benchmark for real-world planning with language agents. paper, 2024. URL https://arxiv.org/abs/2402.01622.

Weikai Xu, Chengrui Huang, Shen Gao, and Shuo Shang. Llm-based agents for tool learning: A survey. paper, 2025. URL https://doi.org/10.1007/s41019-025-00296-9.

Xin Yang, Jie-Jing Shao, Lan-Zhe Guo, B. Zhang, Zhi Zhou, Lin-Han Jia, Wang-Zhou Dai, and Yu-Feng Li. Neuro-symbolic artificial intelligence: Towards improving the reasoning abilities of large language models. paper, 2025. URL https://doi.org/10.24963/ijcai.2025/1195.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. arXiv, 2022. URL https://arxiv.org/abs/2210.03629.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. arXiv, 2023. URL https://arxiv.org/abs/2305.10601.

Asaf Yehudai, Lilach Eden, Alan Li, Guy Uziel, Yilun Zhao, Roy Bar-Haim, Arman Cohan, and Michal Shmueli-Scheuer. Survey on evaluation of llm-based agents. paper, 2025. URL https://arxiv.org/abs/2503.16416.

Jindong Zhang, Jialong Zhou, and Chuang Liu. Or-toolformer: Modeling and solving operations research problems with tool augmented large language models. paper, 2025. URL https://arxiv.org/abs/2510.01253.

Andy Zhou, Kai Yan, Michal Shlapentokh-Rothman, Haohan Wang, and Yu-Xiong Wang. Language agent tree search unifies reasoning acting and planning in language models. arXiv, 2023. URL https://arxiv.org/abs/2310.04406.

A Extended Formal Material

A.1 Symbol Glossary and Equation Provenance

Table 4 records symbol meanings used throughout section 3 and section 4. The provenance column separates borrowed conventions from manuscript-defined quantities so that readers can trace where each formal component originates.

Symbol	Meaning	Provenance
$G = (V, E)$	typed workflow graph	adapted from SymbolicAI framework conventions
$\mathcal{D}(\mathbb{P}_{\text{gen}}, \mathbb{P}_{\text{ref}})$	trajectory-level distance primitive	adapted from VERTEX-style trajectory evaluation framing
$J(\boldsymbol{z}, \pi_{\text{tool}})$	composite constrained objective in equation 1	defined in this manuscript
$\mathcal{F}$	feasible policy set in equation 2	defined in this manuscript
Γ	bridge gain in equation 8	defined in this manuscript
$\mathcal{A}_\delta$	acceptance predicate in equation 10	defined in this manuscript
V_T	comparator path variation for dynamic regret	standard online convex optimization background

Table 4: This glossary consolidates symbol definitions and provenance for the manuscript's formal core. The table is intentionally explicit about manuscript-defined versus adapted quantities to avoid conflating prior conventions with new formal contributions.

A.2 Additional Proof Notes

On Theorem 5.1. The existence argument requires nonempty feasibility and continuity. In practice, nonemptiness is the operationally fragile condition; if strict $(B_{\text{cpu}}, \tau_{\text{valid}})$ settings eliminate all feasible policies, optimization is still computationally executable but the theorem statement no longer applies. This is why feasibility-frontier reporting in figure 1 is part of claim interpretation rather than supplementary decoration.

On Theorem 5.2. The dynamic-regret expression in equation 5 is useful because it separates static complexity, stochastic optimization noise, and non-stationarity through V_T. However, an unresolved symbolic checkpoint in the relaxed simplex/subgradient validation remains open. We therefore treat the bound as conditionally informative and avoid global guarantee language.

On Theorem 5.4. Acceptance safety in equation 10 relies on bounded i.i.d. gains. The symbolic counterexample table identifies heavy-tailed regimes where this assumption fails, in which case empirical-Bernstein-style alternatives should replace Hoeffding-style bounds. This is a methodological requirement, not a post-hoc preference.

B Extended Evidence and Negative Results

B.1 Symbolic Obligation Audit Summary

Table 5 summarizes the obligation outcomes used in this manuscript.

B.2 Failure-Case Tables

The next two tables provide concrete failure cases extracted from threshold-miss routing slices and high-divergence bridge episodes. They are not exhaustive, but they convert prior placeholder artifacts into queryable evidence for falsification follow-up.

C Reproducibility and Implementation Details

C.1 Seeds, Sweeps, and Statistical Procedure

All core experiments were run with seeds $\{11, 23, 37, 73, 101\}$ and budget-validity sweeps over $B_{\text{cpu}} \in \{60, 120, 240\}$ and $\tau_{\text{valid}} \in \{0.80, 0.90, 0.95\}$. Bridge analyses used bootstrap replicate sweeps and protocol-stratified transfer checks. Drift analyses used perturbation tiers and confidence-level sweeps to characterize acceptance sensitivity. These settings are sufficient for directional comparisons but not for final scaling law claims.

Symbolic Audit Check	Status	Interpretation
Router simplex check	fail	symbolic checkpoint unresolved
Router regret non-negativity	pass	decomposition non-negativity verified
Router static boundary case	pass	static-comparator behavior verified
Bridge class inclusion	pass	inclusion property verified
Bridge gain derivation	pass	algebraic consistency verified
Bridge regularization monotonicity	pass	split-consistent check verified
Acceptance predicate rearrangement	pass	predicate algebra verified
Acceptance monotonicity in n, δ	fail	monotonicity remains unresolved
Acceptance finite-sample boundary	pass	boundary handling verified

Table 5: This table provides the symbolic obligation status used to scope theorem language in the main text. The two failed obligations are explicitly surfaced so that formal claims remain conditional and readers can distinguish verified algebraic structure from open proof-audit tasks.

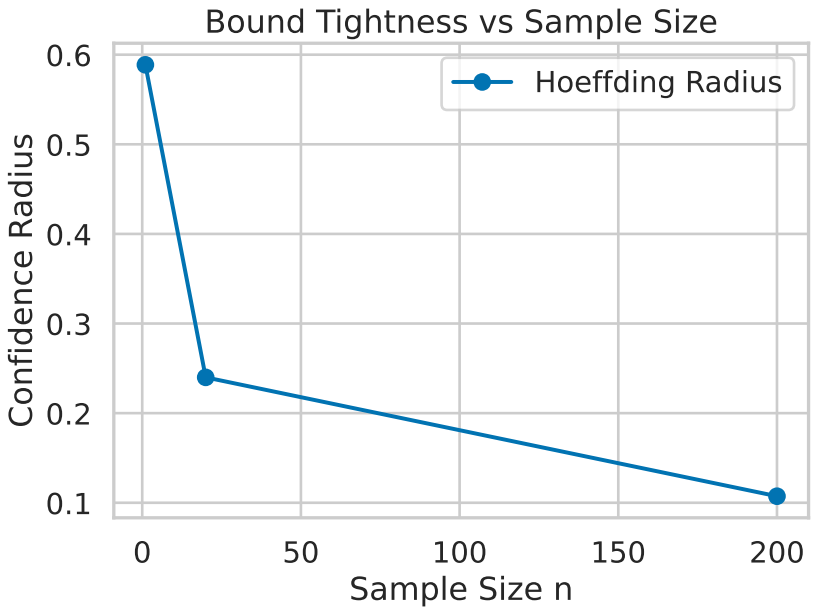

Figure 4: This appendix figure visualizes confidence-radius shrinkage with increasing sample size in the uncertainty audit workflow. The plot provides an interpretable boundary check for acceptance behavior and complements the symbolic obligation table by showing where finite-sample conservatism changes materially with n. The figure supports conditional use of acceptance guarantees rather than unconditional deployment claims.

C.2 Compute and Environment Constraints

The experimental package was executed under CPU-only conditions, with bounded per-run core-hour envelopes. This constraint is scientifically relevant because some planning-heavy baselines can trade quality for substantially higher compute. We therefore report cost-aware comparisons and avoid unrestricted-depth baselines.

C.3 Symbolic Reproducibility Details

Symbolic checks evaluate theorem obligations, boundary conditions, and counterexample detection for violated assumptions A1, A3, and A6. Boundary checks include small-n and high-δ settings for acceptance auditing. Counterexample detection is treated as a required output, not an optional stress test, because it determines whether theorem language can remain strong or must be caveated.

Seed	CPU Budget (s)	Validity Threshold	ΔJ vs OR-Toolformer	ΔJ vs Fixed Route
101	120	0.90	-0.038	-0.027
73	240	0.90	0.062	-0.055
101	60	0.90	-0.000	-0.038
37	60	0.95	0.010	-0.014
23	60	0.90	0.002	0.056

Table 6: This table reports routing slices where at least one planned improvement threshold is missed under strict budget-validity settings. These rows bound the constrained-improvement claim and identify where fallback analyses remain necessary.

Episode	Benchmark	Seed	Divergence	Failure Tag
AgentBench-s11-10	AgentBench	11	1.000	Incomplete plan
SWEbenchLite-s11-30	SWE-bench Lite	11	1.000	Invalid tool call
ExtensityAI-s23-15	ExtensityAI	23	1.000	Invalid tool call
AgentBench-s23-15	AgentBench	23	1.000	Invalid tool + incomplete plan
SWEbenchLite-s23-25	SWE-bench Lite	23	1.000	Incomplete plan

Table 7: This table reports high-divergence bridge episodes with invalid-tool or incomplete-plan failure tags. The rows identify disagreement strata that remain difficult even when aggregate bridge metrics are favorable.

C.4 Code Artifact Summary

The implementation stack contains dedicated components for experiment orchestration, simulation, plotting, and symbolic audits. This modular structure mirrors the manuscript decomposition: optimization, metric bridging, uncertainty evaluation, and formal-check auditing are implemented as distinct modules so that claim-level failures can be localized. The release package should continue to improve negative-result logging density and claim-evidence metadata closure in future iterations.