Agentic Uncertainty Quantification

Anonymous ACL submission

Abstract

Although AI agents have demonstrated impressive capabilities in long-horizon reasoning, their reliability is severely hampered by the "Spiral of Hallucination," where early epistemic errors propagate irreversibly. Existing methods face a dilemma: uncertainty quantification (UQ) methods typically act as passive sensors, only diagnosing risks without addressing them, while self-reflection mechanisms suffer from continuous or aimless corrections. To bridge this gap, we propose a unified Dual-Process Agentic UQ (**AUQ**) framework that transforms verbalized uncertainty into active, bi-directional control signals. Our architecture comprises two complementary mechanisms: System 1 (Uncertainty-Aware Memory, **UAM**), which implicitly propagates verbalized confidence and semantic explanations to prevent blind decision-making; and System 2 (Uncertainty-Aware Reflection, **UAR**), which utilizes these explanations as rational cues to trigger targeted inference-time resolution only when necessary. This enables the agent to balance efficient execution and deep deliberation dynamically. Extensive experiments on closed-loop benchmarks and open-ended deep research tasks demonstrate that our training-free approach achieves superior performance and trajectory-level calibration. We believe this principled framework **AUQ** represents a significant step towards reliable agents [1].

1 Introduction

The transition from LLMs to autonomous agents has unlocked capabilities in long-horizon reasoning and tool use (Yao et al., 2022b; Schick et al., 2023). However, as agents operate in dynamic environments, they face a critical reliability bottleneck: the *"Curse of Recursion"* (Cemri et al., 2025). Unlike single generation, agentic workflows are prone to *"Spiral of Hallucination"* (Dziri et al., 2023; Zhang et al., 2024; Kalai et al., 2025), where a minor grounding error in an early reasoning step propagates through the context window, biasing all subsequent planning towards an irreversible failure state (Zhang et al., 2025). To deploy agents in high-risk real-world scenarios, they need not only reasoning capabilities but also the ability to detect when they are deviating from the correct path before errors propagate.

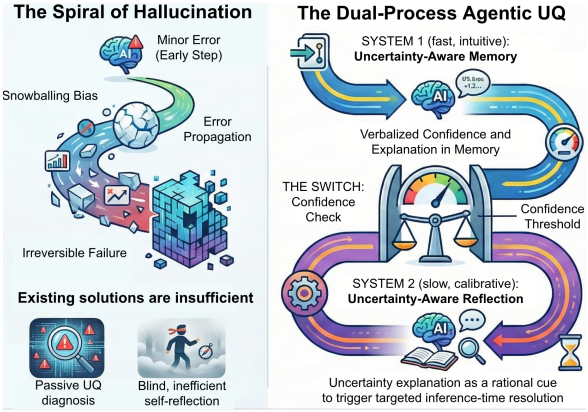

Figure 1: Overview of Dual-Process Agentic UQ Framework. To address the spiral of hallucination challenges in long-horizon agents, **AUQ** transforms verbalized uncertainty into active, bi-directional control signals, comprising two complementary mechanisms: System 1 (Uncertainty-Aware Memory, **UAM**), which implicitly propagates verbalized confidence and semantic explanations to prevent blind decision-making; and System 2 (Uncertainty-Aware Reflection, **UAR**), which utilizes these explanations as rational cues to trigger targeted inference-time resolution only when necessary. This allows the agent to balance efficient execution and deep deliberation dynamically.

Addressing this fragility of long horizons necessitates a shift from blind execution to conscious control. For an agent to be reliable, it must be aware of its own limitations, and know when it knows and when it does not (Kadavath et al., 2022). Recent work like UProp (Duan et al., 2025) and SAUP (Zhao et al., 2025) have formally characterized

[1] All code will be publicly released soon

uncertainty propagation, mathematically demonstrating how local epistemic errors compound into global failures (Zhang, 2021). While they successfully *quantify* the risk, they do not inherently provide a mechanism to *resolve* it. Conversely, current mitigation strategies rely on *self-reflection* mechanisms (Shinn et al., 2023; Madaan et al., 2023). However, without access to ground-truth labels (unavailable in open-ended tasks), standard reflection is often triggered blindly or incessantly, leading to computational inefficiency or the *sycophancy effect* (Huang et al., 2023), where the model hallucinates justifications for its errors (Renze and Guven, 2024). More discussions can be found in the related work section in Appendix A.1.

We believe that reliable long-term intelligent agents need to bridge this gap by transforming uncertainty into bidirectional control signals: *Forward Propagation* for constraint and *Inverse Calibration* for problem-solving. To this end, we propose **Uncertainty-Aware Memory** (**`UAM`**) to explicitly retain verbalized confidence and explanations in the agent's context window. This leverages the Transformer's attention mechanism to naturally suppress overconfidence, creating a soft cognitive constraint that prevents errors from solidifying (Forward). Simultaneously, when suppression signals critical instability, the retained explanation serves as a **rational cue**, transforming vague epistemic anxiety into targeted diagnosis via **Uncertainty-Aware Reflection** (**`UAR`**) (Inverse). Unlike single-step generation (where the process ends at the output), our agent utilizes this cue to trigger targeted actions, such as switching tools or expanding retrieval, effectively transforming epistemic uncertainty into an information-search strategy.

This dual mechanism transforms uncertainty from a passive indicator into an active control signal, implicitly constraining the operation of System 1 (fast, intuitive) using memory and explicitly guiding the operation of System 2 (slow, reflective) using interpretation (Kahneman, 2011; Li et al., 2025). To realize this vision, we propose a unified framework for Agentic UQ based on a dual-process architecture. Our key contributions are threefold:

- We formally decouple Agentic UQ into two complementary mathematical problems: *Forward Uncertainty Propagation* (preventing epistemic errors from solidifying into history) and *Inverse Uncertainty Calibration* (using inference-time compute to correct deviations). To our knowledge, this is the first work to frame agent reliability through this dual-process lens.
- We introduce a novel dual-process architecture, combining *System 1 (Uncertainty-Aware Memory)* and *System 2 (Uncertainty-Aware Reflection)*, as well as a trajectory-level evaluation metric, tailored for long-horizon reliability calibration. Our approach is a *training-free* framework, which enables the agent to dynamically balance efficient execution with deep deliberation.
- We validate the framework across diverse domains, spanning embodied decision-making (`ALFWorld`), web agents (`WebShop`), and open-ended reasoning (`DeepResearch`). Our results demonstrate that **`AUQ`** achieves superior performance, effectively bridging the gap between passive UQ and active agentic control, while yielding superior trajectory-level calibration.

2 Preliminaries

2.1 Long-horizon Agentic Process

We model the long-horizon agent's system as a Partially Observable Markov Decision Process (POMDP) defined by the tuple $\mathcal{E} = (\mathcal{S}, \mathcal{A}, \Omega, \mathcal{T}, \mathcal{R})$. At step t, the agent acts based on a history-dependent policy $\pi(a_t|h_t)$, where $h_t = (o_0, a_0, \ldots, o_t)$ represents the observed trajectory. Unlike single-step LLM, where the full context is visible, autonomous agents operate under *state uncertainty* (the true state s_t is latent). The agent maintains an implicit *belief state* $b_t(s_t) = P(s_t|h_t)$. In this context, reliability failure occurs when the agent's internal belief b_t diverges from the true state s_t, while the policy π continues to act on this flawed belief. Therefore, the core objective of Agentic UQ is to quantify the uncertainty of b_t and its propagation over trajectories.

2.2 Decomposing Uncertainty: From Static to Agentic

Classical UQ distinguishes between aleatoric and epistemic uncertainty (Kendall and Gal, 2017), but these definitions undergo a semantic shift, particularly regarding their temporal interaction.

Aleatoric Uncertainty (Environmental Stochasticity): typically refers to inherent randomness of systems and irreducible noise in data. In agentic settings, it maps to the *stochasticity of the environment interface*. This represents the objective, external friction of the world that the agent cannot eliminate but must robustly handle.

Epistemic Uncertainty (Cognitive Deficiency): Traditionally, this stems from a lack of knowledge and ignorance of model parameters. In Agentic Context: it maps to the *agent's reasoning limitations*, manifesting as hallucinations, logic gaps, or memory failures. Crucially, this is reducible via better reasoning strategies or external knowledge.

The Entanglement in Long Trajectories: A key insight of our work is that these two uncertainties are *not independent* in agentic UQ. When an agent acts with high epistemic uncertainty at step t and commits it to the history h_t, this error becomes part of the "ground truth" context for step $t+1$. This phenomenon, which we term the *Spiral of Hallucination*, effectively transforms the agent's internal cognitive error into an external "environmental" constraint for future steps. This unique dynamics necessitates a dual approach: preventing the propagation of epistemic errors (Forward) and correcting them before they solidify (Inverse).

2.3 Problem Formulation

We decouple Agentic UQ into two complementary mathematical problems, mapping the intuitive concepts of *Propagation* and *Calibration* to rigorous probabilistic formulations.

Forward Problem: Uncertainty Propagation. As early errors infect future steps in long-horizon trajectories, it requires the agent to propagate its confidence through time, ensuring that the current uncertainty estimate c_t accounts for the accumulated risk of the history. We thus define the *Forward Problem* as estimating the joint probability of the trajectory's validity up to time t. Let V_t be a binary variable indicating if the trajectory h_t is valid (free of critical errors). The goal is to estimate:

$$P(V_t|h_t) = f_{\text{p}}(P(V_{t-1}|h_{t-1}), \pi(a_t|h_t)) \quad (1)$$

where $f_{\text{p}}(\cdot)$ denotes the prompt formatting function that wraps the history into the textual input.

Inverse Problem: Uncertainty Calibration. The inverse problem aims to infer the optimal latent process or action a^* given a desired outcome or constraints. We define the Inverse Problem as a **posterior optimization** task. Given a low forward estimate $P(V_t|h_t) < \delta$ (indicating a potential divergence between belief b_t and state s_t), the objective is to find a corrected action a^* that maximizes the likelihood of task success, potentially by inferring a latent reasoning path z:

$$a^* = \underset{a}{\text{argmax}} \int P(a|z, h_t)P(z|\text{Succ}, h_t)dz \quad (2)$$

We view this as *Test-Time Calibration* (Cobbe et al., 2021). When the forward process yields low confidence, we treat the generation of a reliable plan as an inverse search problem (see Appendix A.2).

3 Methodology

We propose a unified Dual-Process UQ Framework that dynamically decouples the agent's mechanism into two distinct modes based on uncertainty: a fast, memory-augmented **System 1** for forward uncertainty propagation, and a slow, reflection-based **System 2** for inverse uncertainty calibration.

3.1 System 1: Forward UQ via Uncertainty-Aware Memory (UAM)

The primary objective of System 1 is to maintain *Cognitive Continuity*. It acts as the default "Fast Path," propagating uncertainty constraints implicitly through the agent's context window. Instead of relying on opaque logit probabilities, which are often miscalibrated in RLHF-aligned models (Tian et al., 2023), we implement a *verbalized confidence sensor* by defining an elicitation mapping $\Phi : h_t \rightarrow (a_t, \hat{c}_t, \hat{e}_t)$, where the agent generates the action a_t, a confidence scalar $\hat{c}_t \in [0, 1]$, and a natural language explanation $\hat{e}_t$. This explanation $\hat{e}_t$ serves to explicitly verbalize latent epistemic uncertainties, see more discussion in Appendix A.5.4).

We formalize **UAM** as an augmented history structure that approximates the agent's belief state over time. Unlike standard agents that only retain the trajectory of observations and actions $h_t = \{(o_i, a_i)\}_{i=0}^{t-1}$, our framework explicitly constructs an *Uncertainty-Aware Memory* that retains the agent's metacognitive states:

$$\mathcal{M}_t = \{(o_i, a_i, \hat{c}_i, \hat{e}_i)\}_{i=0}^{t-1} \quad (3)$$

Here, $\hat{c}_i$ and $\hat{e}_i$ are the verbalized confidence and semantic explanation generated at step i. By preserving $(\hat{c}_i, \hat{e}_i)$, the agent maintains a record of its own reliability, preventing the loss of critical risk signals as the context window slides.

The core mechanism of **UAM** is *Semantic Uncertainty Propagation*, denoted by π_{fwd} policy. While we do not apply hard symbolic rules to block actions, retaining $\hat{e}_t$ in the context window imposes a *Soft Cognitive Constraint* via the Transformer's self-attention mechanism. Mathematically, the probability of the next action $P(a_{t+1}|\mathcal{M}_t)$ is conditioned on the explicit articulation of prior doubts. When the attention heads attend to tokens in $\hat{e}_i$ describing uncertainty, the generative distribution

naturally shifts away from high-commitment actions (Exploitation) towards information-gathering actions (Exploration). This effectively propagates uncertainty forward in time, allowing the agent to dynamically adjust its behavior based on the accumulated uncertainty in its memory. To implement System 1 without fine-tuning, we employ a **Confidence Elicitation Protocol** by appending a structured instruction to the inference prompt, requiring the agent to output a confidence score $\hat{c}_t \in [0, 1]$ and a semantic explanation $\hat{e}_t$ alongside its action

3.2 System 2: Inverse UQ via Uncertainty-Aware Reflection (UAR)

System 2 acts as the slow path intervention. It treats reliability as an *Inverse Optimization Problem*: given the detected diagnostic signal $\hat{e}_t$, we aim to infer a corrected action a^* that maximizes information completeness and logic consistency.

The Reflection Operator. Standard reflection often suffers from degenerate feedback loops (Renze and Guven, 2024). To enforce effective correction, we utilize the verbalized explanation $\hat{e}_t$ as a **Rational Cue** from System 1 as an explicit diagnostic constraint. We define a reflection operator π_{inv} that transforms the unconditioned policy into a conditional distribution parameterized by the identified knowledge gaps. For an initial action a_{init} and explanation $\hat{e}_{init}$, the operator constructs a reflection prompt $\mathcal{P}_{ref}(\hat{e}_{init})$ to guide the resampling: $a_{new}, \hat{c}_{new}, \hat{e}_{new} \sim \pi_{\text{inv}}(\cdot \mid h_t, a_{init})$. Crucially, we inject $\hat{e}_{init}$ into $\mathcal{P}_{ref}$ (see Appendix A.6), explicitly instructing the model to "address the concerns mentioned in: $\hat{e}_{init}$". This reflection occurs *within* the same time step t. If System 2 is triggered, the initial proposal a_{init} is discarded, and the selected candidate a_{new} becomes the realized action a_t sent to the environment.

Consistency-Weighted Reflection. We employ a *Best-of-N* sampling strategy to generate N parallel reflection trajectories $\{(a_{new}^{(k)}, \hat{c}_{new}^{(k)})\}_{k=1}^{N}$ to encourage diversity. Instead of simply selecting the maximum confidence, we calculate a consistency-weighted score $S_{cons}(a)$ for each action candidate:

$$S_{cons}(a) = \frac{1}{N} \sum_{k=1}^{N} \hat{c}_{new}^{(k)} \cdot \mathbb{I}(a_{new}^{(k)} \equiv a) \quad (4)$$

where $\mathbb{I}(\cdot)$ is an indicator function for semantic equivalence, see Appendix A.3.3 for more explanations. This metric rewards answers that are both *high-confidence* and *consistent*. The provisional optimal action is $a_{new}^* = \arg\max_a S_{cons}(a)$.

Adaptive Memory Expansion. To balance efficiency and context retention, our agent typically operates with a *Limited Memory* window ($\mathcal{M}_{limit} = h_{t-k:t}$) during inference. However, epistemic uncertainty often stems from forgetting (Kirsch, 2024; Liu et al., 2024b). We introduce a *Memory Expansion Mechanism*. After the initial reflection, if the aggregated score $S_{cons}(a^*)$ remains below the reliability threshold τ, the agent triggers a *Context Retrieval* operation, loading the *Full Memory* $\mathcal{M}_{full} = h_{0:t}$ and re-executing the reflection. This mechanism ensures that expensive long-context processing is only utilized when local reasoning fails to resolve uncertainty, effectively creating a tiered defense against error propagation. More details are provided in Appendix A.6.

3.3 Dual-Process Agentic UQ

Having defined the two systems, we now formalize the **Dual-Process Policy** π_{dual} that integrates them. To balance reliability and efficiency, we introduce a switching function $S(h_t)$ that governs the transition between systems:

$$\pi_{\text{dual}}(a|h_t) = \begin{cases} \pi_{\text{fwd}}(a|h_t, \mathcal{M}_t), & \text{if } S(h_t) = 0 \\ \pi_{\text{inv}}(a|h_t), & \text{if } S(h_t) = 1 \end{cases}$$

where $S(h_t) = \mathbb{I}(\hat{c}_t < \tau)$ is an indicator function triggered when the agent's self-evaluated confidence $\hat{c}_t$ falls below a reliability threshold τ, which is a threshold determined empirically (typically $\tau \in [0.8, 1)$ based on validation datasets). When $\hat{c}_t \geq \tau$, the agent executes the System 1 action directly, incurring minimal cost. When $\hat{c}_t < \tau$, the agent activates the System 2 loop. The corrected result is then written back to $\mathcal{M}_t$ with its updated confidence, ensuring that the resolved uncertainty is propagated to future steps. This mechanism realizes *dynamic inference budgeting*, allocating expensive reflection only to high-uncertainty steps. The complete execution flow is detailed in Algorithm 1.

3.4 Trajectory-Level Evaluation Metrics

Standard calibration metrics (e.g., ECE) are ill-suited for long-horizon agentic tasks because they decouple local confidence from global success. A single failure step often invalidates the entire trajectory, regardless of the confidence of other steps. To address this, we introduce a generalized framework for **Trajectory-Level Calibration**. We define a trajectory as a sequence of confidence scores

Dataset	Method	End-State Calibration (Φ_{last})		Overall Quality (Φ_{avg})		Process Reliability (Φ_{min})	
		ECE (↓)	Brier Score (↓)	ECE (↓)	Brier Score (↓)	ECE (↓)	Brier Score (↓)
ALFWorld	ReAct	0.306	0.258	0.286	0.272	0.255	0.236
	Reflexion	0.279	0.237	0.234	0.264	0.199	0.219
	Self-Reflection	0.264	0.227	0.223	0.254	0.185	0.218
	CoT-SC	0.185	0.224	0.177	0.215	0.178	0.206
	Forward UQ (**UAM**)	**0.109**	0.220	**0.160**	0.222	0.104	0.217
	Inverse UQ (**UAR**)	0.205	**0.202**	0.207	0.205	0.131	0.191
	Dual-Process (**AUQ**)	0.174	0.218	0.162	**0.198**	**0.093**	**0.176**
WebShop	ReAct	0.335	0.288	0.342	0.305	0.329	0.298
	Reflexion	0.352	0.275	0.334	0.296	0.272	0.284
	Self-Reflection	0.310	0.278	0.295	0.288	0.298	0.276
	CoT-SC	0.225	0.263	0.269	0.253	0.242	0.250
	Forward UQ (**UAM**)	**0.185**	0.245	0.221	0.228	**0.138**	0.268
	Inverse UQ (**UAR**)	0.209	**0.231**	0.242	**0.222**	0.262	0.248
	Dual-Process (**AUQ**)	0.188	0.242	**0.215**	0.236	0.210	**0.235**

Table 1: Trajectory-Level Reliability Analysis. We report Trajectory-ECE (T-ECE, lower is better) and Brier Score (T-BS, lower is better) on ALFWorld and WebShop. **Forward (UAM)** achieves the best calibration (lowest ECE) by effectively dampening overconfidence, while **Inverse (UAR)** achieves the best sharpness (lowest BS) by resolving uncertainty through reflection. The Dual-Process framework strikes an optimal balance.

$\mathbf{c} = \{\hat{c}_1, \ldots, \hat{c}_T\}$. We posit that the effective "trajectory belief" $C(\tau)$ is determined by a trajectory confidence aggregation function Φ: $C(\tau) = \Phi(\mathbf{c})$, consisting of three distinct aggregation strategies to capture different dimensions: (1) **End-State Belief (Φ_{last}):** $C(\tau) = \hat{c}_T$. This assumes reliability depends solely on the last-step final decision; (2) **Process Reliability (Φ_{min}):** $C(\tau) = \min_{t=1}^{T} \hat{c}_t$. Based on the *Weakest Link Principle* (Zhong et al., 2024), a trajectory is only as reliable as its most uncertain step; and (3) **Overall Quality (Φ_{avg}):** $C(\tau) = \frac{1}{T}\sum_{t=1}^{T} \hat{c}_t$. This represents the average confidence throughout the task. The detailed metrics of trajectory-ECE, Brier Score (BS), and AUROC are described in Appendix A.3.4.

4 Experiments

4.1 Experimental Setup

Datasets and Benchmarks. To demonstrate the broad applicability of our framework across distinct agentic regimes, we evaluate on three diverse benchmarks: (1) **ALFWorld** (Shridhar et al., 2021): A precise embodied decision-making suite. (2) **WebShop** (Yao et al., 2022a): A realistic e-commerce environment characterized by high observation noise. (3) **DeepResearch Bench** (Du et al., 2025): A benchmark for open-ended information seeking, consisting of 100 PhD-level research tasks. Detailed task descriptions and evaluation protocols are provided in Appendix A.3.1.

Baselines. We compare our framework against three categories of methods to isolate specific agentic capabilities: (1) **ReAct** (Yao et al., 2022b): The standard System 1 baseline. The agent reasons and acts in an interleaved manner without explicit self-evaluation or reflection. (2) **Reflexion** (Shinn et al., 2023): A strong *inter-episode* learning baseline representing System 2 self-correction without uncertainty awareness. (3) **Self-Reflection** (Renze and Guven, 2024): An *intra-episode* baseline that triggers reflection on every step (or uses a heuristic), by simply asking the model to validate its action for unguided compute scaling. (4) **CoT-SC** (Wang et al., 2022): An ensemble baseline, verifying whether our gains stem purely from sampling diversity rather than targeted reflection. To dissect the Dual-Process architecture, we evaluate three variants of our method: Forward (**UAM**-only), Inverse (**UAR**-only), and Dual-Process (**AUQ**). The detailed descriptions are provided in Appendix A.3.2.

Models and Implementation Details. We conduct a comprehensive evaluation across a spectrum of closed and open-source LLMs, including **GPT-5.1**, **GPT-4.1**, and **GPT-4o** (Achiam et al., 2023); **Gemini-2.5-Pro** and **Gemini-2.5-Flash** (Comanici et al., 2025); **Qwen3-235B** (Yang et al., 2025) and **DeepSeek-V3.1** (Bi et al., 2024), representing the frontier of open-source capabilities. Our Inverse UQ module employs a **Best-of-N** strategy ($N = 3$ parallel paths). For scenarios with limited memory context ($h = 5$), we activate the **Memory Expansion** mechanism upon reflection failure. All experiments on ALFWorld and WebShop are capped at a maximum of 50 steps. To demonstrate generalizability, we integrated **AUQ** into the *Enterprise Deep Research (EDR)* framework (Prabhakar et al.,

Method	ALFWorld				WebShop			
	Success Rate (↑)	AUROC (↑)			Success Rate (↑)	AUROC (↑)		
	(%)	Φ_{last}	Φ_{avg}	Φ_{min}	(%)	Φ_{last}	Φ_{avg}	Φ_{min}
ReAct	63.6	0.913	0.783	0.667	29.3	0.863	0.758	0.608
Reflexion	67.9	0.925	0.820	0.721	30.7	0.840	0.742	0.630
Self-Reflection	66.4	0.922	0.831	0.718	33.4	0.855	0.735	0.615
CoT-SC	69.5	0.948	0.847	0.763	37.1	0.861	0.769	0.663
Forward UQ (UAM)	65.7	0.963	0.841	**0.807**	31.4	0.865	0.775	0.725
Inverse UQ (UAR)	72.9	0.958	0.856	0.774	38.6	0.874	**0.791**	0.710
Dual-Process (AUQ)	**74.3**	**0.968**	**0.905**	0.791	**42.9**	**0.888**	0.782	**0.755**

Table 2: Performance and Discrimination Analysis. We compare Success Rate (SR) and Discriminative AUROC. Dual-Process (**AUQ**) consistently outperforms strong baselines.

2025). See more details in Appendix A.3.3.

Evaluation Metrics. We adopt an evaluation protocol to assess agent performance across three dimensions: (1) **Performance:** We report Success Rate (SR) for ALFWorld and WebShop. For the open-ended DeepResearch Bench, we utilize their RACE rubric. We employ Gemini-2.5-Pro as an impartial LLM-as-a-Judge to score the generated reports. (2) **Calibration:** We assess how well the agent's confidence aligns with reality using **Trajectory-ECE (T-ECE)** and **Trajectory Brier Score (T-BS)**. (3) **Discrimination:** We report the (**AUROC**) to quantify the system's ability to distinguish between success and failure modes purely based on internal confidence signals.

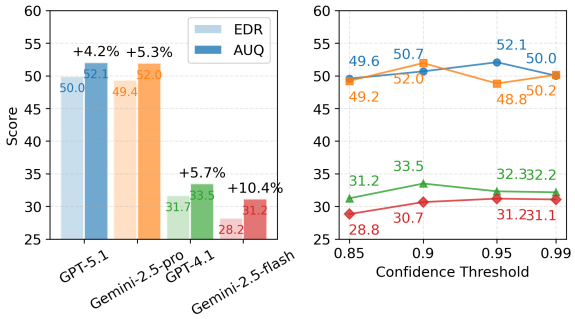

Figure 2: Generalization on Deep Research. (Left) **AUQ** outperforms the Enterprise Baseline (EDR) across four diverse LLM backends. **AUQ** consistently outperforms the Enterprise Baseline (EDR). (Right) Sensitivity analysis of confidence threshold τ. The performance is robust across models in the high-confidence regime.

4.2 Main Results

4.2.1 Closed-Loop Decision Making

Reliability (Calibration): Table 1 highlights the calibration dynamics via Trajectory-ECE and BS. **ReAct** agents exhibit severe miscalibration on the process reliability metric (Φ_{min}), confirming that snowballed hallucinations often go undetected by the agent itself. In contrast, our **Forward (UAM-Only)** variant achieves superior T-ECE. By retaining verbalized uncertainty in memory, **UAM** effectively aligns the agent's confidence with its actual capabilities. However, while **UAM** is calibrated, **Inverse (UAR-Only)** is decisive. It achieves the lowest BS by actively resolving knowledge gaps via reflection, polarizing the belief distribution towards certainty. **AUQ** integrates these strengths, maintaining high calibration while significantly improving resolution compared to the baseline.

Utility (Success): Table 2 demonstrates that superior calibration translates directly into task performance. **Dual-Process (AUQ)** consistently outperforms baselines, achieving a remarkable **74.3% SR (+10.7%)** and **42.9% SR (+13.6%)** on ALFWorld and WebShop respectively. This gain is particularly significant compared to **CoT-SC**, indicating that our *Consistency-Weighted* reflection provides quality gains beyond simple ensembling. Furthermore, Dual-Process and its variants achieve superior **AUROC** against baselines, proving that its internal confidence signals reliably distinguish between success and failure. This discriminative power confirms that the system effectively allocates its System 2 budget to the trajectories that need it most, rather than reflecting blindly or randomly.

4.2.2 Open-Ended Deep Research

This task demands high-fidelity reasoning, where success is defined by the depth and coherence of the final report rather than binary completion. Table 3 benchmarks our **Dual-Process AUQ** framework against widely deployed *Deep Research Agents* and open-source frameworks. **AUQ** achieves a state-of-the-art Overall Score of **52.09**, outperforming the strongest closed baseline (**49.71**) and the best open-source competitor (**50.62**). Critically, this ad-

Agent systems	Overall	Comprehensiveness	Insight	Instruction	Readability
langchain-open-deep-research	43.44	42.97	39.17	48.09	45.22
doubao-research	44.34	44.84	40.56	47.95	44.69
kimi-research	44.64	44.96	41.97	47.14	45.59
Claude-research	45.00	45.34	42.79	47.58	44.66
Openai-deepresearch	46.45	46.46	43.73	49.39	47.22
Gemini-2.5-pro-deepresearch	49.71	49.51	49.45	50.12	50.00
WebWeaver (Qwen3-235b)	50.62	51.29	51.00	49.98	48.89
WebWeaver (Claude-sonnet-4)	50.58	51.45	50.02	50.81	49.79
Enterprise Deep Research (Gemini-2.5-pro)	50.62	49.70	51.24	50.52	50.61
AUQ (Ours, Gemini-2.5-Pro)	51.97	49.19	53.64	**51.67**	**51.26**
AUQ (Ours, GPT-5.1)	**52.09**	**51.60**	**54.21**	50.69	50.13

Table 3: Evaluation on Open-Ended DeepResearch Bench. We compare our approach against enterprise-grade systems and recent agent frameworks using the RACE rubric (Du et al., 2025). The Dual-Process architecture achieves superior performance in Comprehensiveness, Insight, Instruction Following, and Readability.

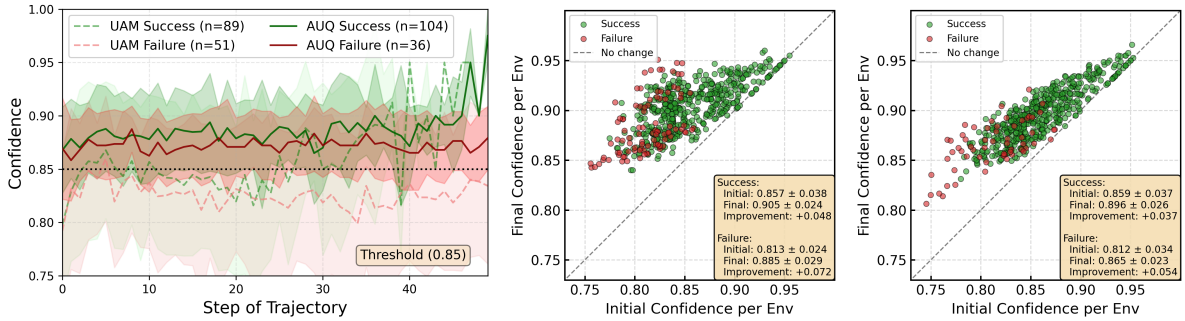

Figure 3: Internal Belief Dynamics on ALFWorld. (Left) Trajectory confidence evolution comparing **UAM-Only** (Dashed) and **Dual-Process AUQ** (Solid). (Right) Scatter plots of Pre- vs. Post-Reflection confidence (GPT-4o & GPT-5.1). Yellow boxes show summary statistics.

vantage stems from superior *Insight* (**54.21**) and *Comprehensiveness* (**51.60**). Baseline agents often exhibit "information satisficing", retrieving surface-level facts and stopping. In contrast, our Inverse UQ mechanism triggers a deep dive reflection when it detects epistemic gaps, transforming potential shallow summaries into rigorous, evidence-backed arguments. Figure 2 (Left) illustrates **AUQ**'s impact across four diverse backends. We observe universal gains, with the relative improvement (e.g., **+10.4%** on Gemini-2.5-flash, avg **+6.4%** over all models). Figure 2 (Right) confirms hyperparameter stability: performance forms a stable plateau for τ, indicating that verbalized confidence is a robust signal that does not require intensive tuning.

Qualitative Analysis. Beyond quantitative metrics, we analyze the agent's behavioral shifts in Appendix A.7. Specifically, we provide detailed trajectories for **Deep Research** (Appendix A.7.2 and A.7.3), illustrating how System 2 refines query decomposition; and **ALFWorld** (Appendix A.7.1), demonstrating how **AUQ** detects and breaks snowballed hallucination in embodied tasks.

4.3 Analysis and Discussion

Internal Dynamics of Uncertainty. To demystify how **AUQ** alters the reasoning trajectory, we analyze the evolution of belief in Figure 3. Comparing the confidence traces of **UAM-Only** and **Dual-Process** reveals distinct cognitive roles. **UAM** trajectories remain consistently lower, suppressing blind commitment to create a crucial *Discriminative Margin* between safe and risky states. In contrast, **AUQ** trajectories show a significant rebound, confirming that System 2 functions actively consume compute to eliminate uncertainty. However, the scatter plots (Figure 3 Right) reveal a nuanced *"Delusion Gap"* in this resolution process. While System 2 consistently boosts confidence, failure cases often exhibit significantly larger gains (Δ) than successes. This is because successful trajectories often start with high confidence, where System 2 merely acts as a validator; while in intractable failure cases, aggressive reflection can sometimes lead to *Delusional Confirmation*, creating overconfidence in a hallucinated plan. Despite this risk, the net impact remains overwhelmingly positive. As detailed in our com-

parative analysis (Appendix A.5.1), our framework corrects 14.3% of ReAct's failures while regressing on only 3.6% of its successes.

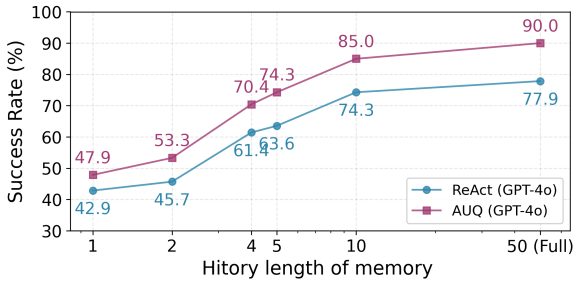

Figure 4: Effect of memory length on **AUQ** and ReAct.

Role of Memory Length and Model Generalization. Figure 4 shows the performance degradation of the agent as we shorten the historical window h from the full history to a single step. A significant difference emerges: ReAct's performance drops sharply with limited memory history because it cannot access the previous observations needed to maintain consistency. In contrast, **AUQ** demonstrates superior resistance to forgetting. Even with $h = 1$, it still maintains significantly higher performance (+5.0%), indicating that the stored uncertainty metadata $(\hat{c}, \hat{e})$ encapsulates the risk state of the trajectory into a compact signal that persists even when the original observation log is truncated. Also, the performance improvement provided by **AUQ** increases with the length of the memory.

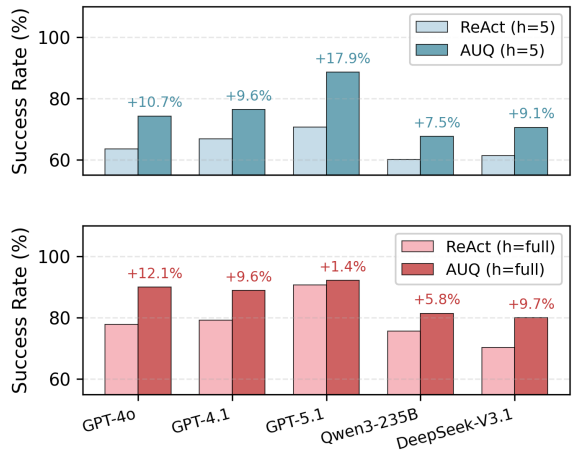

Figure 5: Model generalization and memory expansion.

This robustness is further amplified by our memory expansion architecture, validated in Figure 5. In the constrained $h = 5$ setting, agents often fail due to missing dependencies. **AUQ** recovers this loss via **Adaptive Memory Expansion**, triggering full-context retrieval only when System 2 detects epistemic gaps. This mechanism yields massive gains (e.g., **+17.9%** for GPT-5.1), proving that dynamic retrieval is superior to static windowing. Crucially, these benefits are universal. As shown in the cross-model comparison, **AUQ** consistently outperforms ReAct across diverse models (e.g., **+11.0%** for limited memory, and **+7.7%** for full memory).

Cost-Efficiency Analysis. Figure 6 presents the Pareto frontier of Success Rate versus Inference Cost (avg. API calls). The curve exhibits a distinct inflection point at $\tau \approx 0.9$, representing the optimal efficiency sweet spot. Further increasing the threshold to $\tau = 0.95$ leads to diminishing returns in accuracy while costs increase exponentially, as the agent begins to over-verify even trivial steps. Counterintuitively, our detailed analysis (see Appendix A.5.3) shows that System 2 does not always increase the total computation; in many cases, by detecting low confidence early, the agent can prevent the lengthy and futile hallucination loops common in ReAct, thus transforming wasted failed computations into useful verification computations. Therefore, **AUQ** effectively shifts the paradigm from cheap but wasteful generation to strategic and efficient reasoning (see Appendix A.5.6).

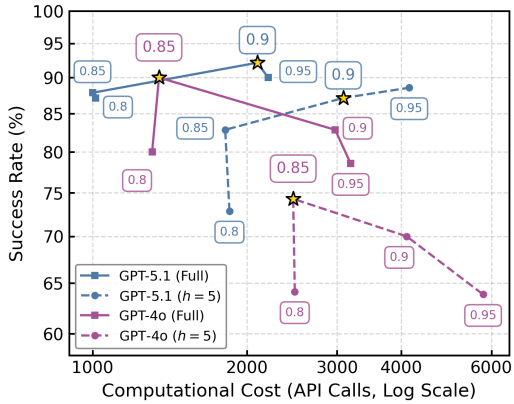

Figure 6: Pareto Efficiency. Success Rate vs. Computational Cost (log scale). **AUQ** achieves a superior Pareto frontier, with optimal efficiency at $\tau \approx 0.9$ (starred).

5 Conclusion

We propose a dual-process agentic UQ framework that bridges the gap between calibration and autonomous reasoning. By decomposing uncertainty management into fast, memory-aware propagation (System 1) and slow, reflective calibration (System 2), we effectively mitigate the hallucination spiral problem that plagues long-horizon tasks. Our extensive experiments demonstrate that this approach not only achieves superior performance but also excels in calibration and self-awareness. These results suggest that principled agentic UQ is an effective approach to building more reliable and adaptive LLM agents.

Limitations

Although our dual-process framework demonstrates significant reliability improvements across various benchmarks, we also acknowledge some limitations that define the scope of our current work and point to future research directions. (1) Our framework is based on the premise that the underlying large language model (LLM) possesses the latent ability to express uncertainty. While we observe a strong correlation between verbally expressed confidence and correctness in strong LLM models (e.g., GPT-5.1, Gemini-2.5-Pro), this capability diminishes in smaller models (e.g., models with fewer than 7 billion parameters). (2) The activation of System 2 (uncertainty-aware reflection) inevitably introduces additional inference latency due to the parallel execution of "best-of-N"sampling and iterative critique loops. While our analysis shows that this latency is typically offset by a reduction in the total number of wasted steps (thus preventing prolonged failure loops), instantaneous latency spikes may be unacceptable for strictly real-time, low-latency applications (e.g., real-time conversational agents).

Ethical Considerations

This work introduces a framework for enhancing the reliability of autonomous agents. We identify two primary ethical implications: (1) *Automation Bias and Over-Reliance.* While our AUQ framework improves calibration, there is a risk that users may over-rely on the agent's verbalized confidence ($\hat{c}_t$), perceiving high confidence as a guarantee of factual correctness. In high-stakes domains (e.g., medical or legal research), even calibrated agents can hallucinate. We emphasize that our system serves as a decision-support tool, and human oversight remains essential for final verification. (2) *Computational Impact.* Our System 2 reflection mechanism (Best-of-N sampling) increases inference-time compute, potentially raising energy consumption. However, as noted in our Efficiency Analysis, this is often offset by preventing lengthy, futile failure loops in unguided agents. We advocate for "Adaptive Risk Budgeting" to deploy such compute-intensive reflection only when necessary, minimizing the environmental footprint.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Zhe Fu, and 1 others. 2024. Deepseek llm: Scaling open-source language models with longtermism. *arXiv preprint arXiv:2401.02954*.

Mert Cemri, Melissa Z Pan, Shuyi Yang, Lakshya A Agrawal, Bhavya Chopra, Rishabh Tiwari, Kurt Keutzer, Aditya Parameswaran, Dan Klein, Kannan Ramchandran, and 1 others. 2025. Why do multi-agent llm systems fail? *arXiv preprint arXiv:2503.13657*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Mingxuan Du, Benfeng Xu, Chiwei Zhu, Xiaorui Wang, and Zhendong Mao. 2025. Deepresearch bench: A comprehensive benchmark for deep research agents. *arXiv preprint arXiv:2506.11763*.

Jinhao Duan, James Diffenderfer, Sandeep Madireddy, Tianlong Chen, Bhavya Kailkhura, and Kaidi Xu. 2025. Uprop: Investigating the uncertainty propagation of llms in multi-step agentic decision-making. *arXiv preprint arXiv:2506.17419*.

Nouha Dziri, Ximing Lu, Melanie Sclar, Xiang Lorraine Li, Liwei Jiang, Bill Yuchen Lin, Sean Welleck, Peter West, Chandra Bhagavatula, Ronan Le Bras, and 1 others. 2023. Faith and fate: Limits of transformers on compositionality. *Advances in Neural Information Processing Systems*, 36:70293–70332.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yujiu Yang, Nan Duan, Weizhu Chen, and 1 others. 2023. Critic: Large language models can self-correct with tool-interactive critiquing. In *The Twelfth International Conference on Learning Representations*.

Tobias Groot and Matias Valdenegro-Toro. 2024. Overconfidence is key: Verbalized uncertainty evaluation in large language and vision-language models. *arXiv preprint arXiv:2405.02917*.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR.

Jiuzhou Han, Wray Buntine, and Ehsan Shareghi. 2024. Towards uncertainty-aware language agent. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 6662–6685.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2023. Large language models cannot self-correct reasoning yet. In *The Twelfth International Conference on Learning Representations*.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, and 1 others. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

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

Adam Tauman Kalai, Ofir Nachum, Santosh S Vempala, and Edwin Zhang. 2025. Why language models hallucinate. *arXiv preprint arXiv:2509.04664*.

Alex Kendall and Yarin Gal. 2017. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30.

Michael Kirchhof, Gjergji Kasneci, and Enkelejda Kasneci. 2025. Position: Uncertainty quantification needs reassessment for large language model agents. In *Forty-second International Conference on Machine Learning Position Paper Track*.

Andreas Kirsch. 2024. (implicit) ensembles of ensembles: Epistemic uncertainty collapse in large models. *arXiv preprint arXiv:2409.02628*.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*.

Zhong-Zhi Li, Duzhen Zhang, Ming-Liang Zhang, Jiaxin Zhang, Zengyan Liu, Yuxuan Yao, Haotian Xu, Junhao Zheng, Pei-Jie Wang, Xiuyi Chen, and 1 others. 2025. From system 1 to system 2: A survey of reasoning large language models. *arXiv preprint arXiv:2502.17419*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*.

Hao Liu, Zi-Yi Dou, Yixin Wang, Nanyun Peng, and Yisong Yue. 2024a. Uncertainty calibration for tool-using language agents. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16781–16805.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024b. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Panagiotis Lymperopoulos and Vasanth Sarathy. 2025. Tools in the loop: Quantifying uncertainty of llm question answering systems that use tools. *arXiv preprint arXiv:2505.16113*.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, and 1 others. 2023. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36:46534–46594.

Akshara Prabhakar, Roshan Ram, Zixiang Chen, Silvio Savarese, Frank Wang, Caiming Xiong, Huan Wang, and Weiran Yao. 2025. Enterprise deep research: Steerable multi-agent deep research for enterprise analytics. *arXiv preprint arXiv:2510.17797*.

Matthew Renze and Erhan Guven. 2024. Self-reflection in llm agents: Effects on problem-solving performance. *arXiv preprint arXiv:2405.06682*.

William Saunders, Catherine Yeh, Jeff Wu, Steven Bills, Long Ouyang, Jonathan Ward, and Jan Leike. 2022. Self-critiquing models for assisting human evaluators. *arXiv preprint arXiv:2206.05802*.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36:68539–68551.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36:8634–8652.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Cote, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. 2021. Alfworld: Aligning text and embodied environments for interactive learning. In *International Conference on Learning Representations*.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2024. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on*

Empirical Methods in Natural Language Processing, pages 5433–5442.

Yao-Hung Hubert Tsai, Walter Talbott, and Jian Zhang. 2024. Efficient non-parametric uncertainty quantification for black-box large language models and decision planning. *arXiv preprint arXiv:2402.00251*.

Hanlin Wang, Jian Wang, Chak Tou Leong, and Wenjie Li. 2025. Steca: Step-level trajectory calibration for llm agent learning. *arXiv preprint arXiv:2502.14276*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. 2022a. Webshop: Towards scalable real-world web interaction with grounded language agents. *Advances in Neural Information Processing Systems*, 35:20744–20757.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2022b. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*.

Jiaxin Zhang. 2021. Modern monte carlo methods for efficient uncertainty quantification and propagation: A survey. *Wiley Interdisciplinary Reviews: Computational Statistics*, 13(5):e1539.

Muru Zhang, Ofir Press, William Merrill, Alisa Liu, and Noah A Smith. 2024. How language model hallucinations can snowball. In *Proceedings of the 41st International Conference on Machine Learning*, pages 59670–59684.

Shaokun Zhang, Ming Yin, Jieyu Zhang, Jiale Liu, Zhiguang Han, Jingyang Zhang, Beibin Li, Chi Wang, Huazheng Wang, Yiran Chen, and 1 others. 2025. Which agent causes task failures and when? on automated failure attribution of llm multi-agent systems. In *Forty-second International Conference on Machine Learning*.

Qiwei Zhao, Dong Li, Yanchi Liu, Wei Cheng, Yiyou Sun, Mika Oishi, Takao Osaki, Katsushi Matsuda, Huaxiu Yao, Chen Zhao, and 1 others. 2025. Uncertainty propagation on llm agent. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6064–6073.

Ming Zhong, Aston Zhang, Xuewei Wang, Rui Hou, Wenhan Xiong, Chenguang Zhu, Zhengxing Chen, Liang Tan, Chloe Bi, Mike Lewis, and 1 others. 2024. Law of the weakest link: Cross capabilities of large language models. In *The Thirteenth International Conference on Learning Representations*.

Kunlun Zhu, Zijia Liu, Bingxuan Li, Muxin Tian, Yingxuan Yang, Jiaxun Zhang, Pengrui Han, Qipeng Xie, Fuyang Cui, Weijia Zhang, and 1 others. 2025. Where llm agents fail and how they can learn from failures. *arXiv preprint arXiv:2509.25370*.

A Appendix

Contents

A.1 Related Work 12
A.2 Formal Mathematics in Problem Formulation 13
A.2.1 Forward Problem: Recursive Validity Estimation . 13
A.2.2 Inverse Problem: Latent Variable Calibration . . . 14
A.3 Detailed Experimental Setup . . . 14
A.3.1 Datasets and Evaluation Protocols 14
A.3.2 Baselines and Variants . . 15
A.3.3 Implementation Details . . 16
A.3.4 Evaluation Metrics and Protocols 18
A.4 Dual-Process Agentic UQ Framework (AUQ) 18
A.5 Additional Experimental Results and Analysis 18
A.5.1 Extended Analysis of Internal Dynamics and Risks 18
A.5.2 Error Analysis and Failure Modes 20
A.5.3 Extended Cost-Efficiency Analysis 21
A.5.4 Discussion: Why Verbalized Confidence? 22
A.5.5 Discussion: On the Dynamics of Thresholding . . 22
A.5.6 Discussion: Efficiency and Cost Analysis 23
A.6 Prompt Templates 24
A.6.1 Baseline Agent 24
A.6.2 System 1: Uncertainty-Aware Memory (Forward UQ) 24
A.6.3 System 2: Uncertainty-Aware Reflection (Inverse UQ) 24
A.7 Qualitative Examples 25
A.7.1 Qualitative Analysis: Embodied Decision Making (ALFWorld) 25
A.7.2 Qualitative Case Study of Deep Research (Chinese Query) 26
A.7.3 Detailed Qualitative Case Study of Deep Research (English Query) 28
A.8 LLM Usage 29

A.1 Related Work

Uncertainty Quantification in LLMs. Accurate UQ is the cornerstone of reliable AI deployment. While classical calibration methods like temperature scaling (Guo et al., 2017) are effective for discriminative tasks, applying them to the open-ended generation of LLMs remains non-trivial (Kadavath et al., 2022). Recent advancements have largely bifurcated into logit-based and linguistic-based approaches. Logit-based methods, such as Semantic Entropy (Kuhn et al., 2023), attempt to aggregate token probabilities over equivalent meanings, yet they often face accessibility challenges with black-box APIs. Consequently, *verbalized uncertainty* where models explicitly express their confidence via natural language prompting, has emerged as a promising alternative (Lin et al., 2022; Tian et al., 2023). Studies show that sufficiently aligned models can produce well-calibrated verbal confidence (Groot and Valdenegro-Toro, 2024), a finding that serves as the theoretical basis for our **Forward UQ (System 1)** module. However, these methods predominantly treat UQ as a static, post-hoc metric for isolated queries. They do not address how such signals can be operationalized to dynamically control the branching and backtracking decisions in the continuous, multi-step trajectories of autonomous agents (Kirchhof et al., 2025).

Autonomous Agents and Error Propagation. While the reasoning-acting paradigm, pioneered by frameworks like ReAct (Yao et al., 2022b) and Reflexion (Shinn et al., 2023), has empowered LLMs to solve long-horizon tasks, these systems exhibit significant brittleness in dynamic environments. Unlike single-turn generation, where errors are isolated, autonomous agents suffer from the "Curse of Recursion" or "Error Propagation". Recent failure analyses have systematically codified this phenomenon. For instance, Zhu et al. (2025) and Cemri et al. (2025) identify that the primary cause of task failure is not merely lack of knowledge, but *"Spiral of Hallucination"* (Zhang et al., 2024; Dziri et al., 2023) where a minor hallucination in an early reasoning step (e.g., misinterpreting a tool output) pollutes the context window, biasing all subsequent planning steps towards an irreversible failure state.

To address this, a surge of recent work has focused on *Automated Failure Attribution*. Approaches like *AgentDebug* (Zhu et al., 2025) and failure causality analysis (Zhang et al., 2025) treat

agent trajectories as diagnosable artifacts, utilizing separate critique models to trace root causes (e.g., planning brittleness vs. grounding errors) after the episode concludes. Other works have explored the dynamics of coordination collapse in multi-agent settings (Cemri et al., 2025), identifying how individual delusions can propagate to system-wide failures. While these contributions provide a valuable taxonomy of *why* and *where* agents break, they remain predominantly diagnostic or rely on distinct "debugging layers" (Zhu et al., 2025). They function as autopsies, explaining the death of the trajectory after it has occurred. Our Dual-Process framework fundamentally shifts this paradigm from post-hoc attribution to **runtime prevention**. By integrating UQ directly into the cognitive loop, we detect the onset of error propagation (System 1) and arrest the snowball effect via immediate reflection (System 2) before it cascades into total task failure.

Uncertainty and Calibration in Agentic Systems. The study of uncertainty in LLM agents is an emerging but critical field, distinct from static text generation (Kirchhof et al., 2025). Pioneering works have begun to formalize the unique challenges agents present, particularly the sequential nature of confidence (Han et al., 2024; Tsai et al., 2024). Frameworks like **UProp** (Duan et al., 2025) and **SAUP** (Duan et al., 2025) were among the first to explicitly model how uncertainty **propagates** through the sequential steps of an agent's trajectory, mathematically characterizing how local errors compound into global failures. Concurrently, other research has focused on quantifying the **external uncertainty** introduced by tool interactions, analyzing how API failures or noisy tool outputs impact overall reliability (Liu et al., 2024a; Lymperopoulos and Sarathy, 2025). While these studies laid the essential groundwork by identifying the core mechanics of propagation and external interaction, they primarily focus on *high-level modeling* or *passive diagnosis*. They quantify the risk but do not necessarily provide architectural mechanisms to resolve it during runtime. Our work bridges this gap by operationalizing these uncertainty signals. Instead of merely observing the propagation of doubt (as in UProp), our Dual-Process framework uses it as a trigger for **active intervention**, switching from fast System 1 execution to slow System 2 reflection to arrest the propagation before it becomes irreversible.

Reflection and Self-Correction Mechanisms. The ability to self-correct is a hallmark of autonomous agents. Early frameworks like **Self-Refine** (Madaan et al., 2023), **Self-Reflection** (Renze and Guven, 2024), and **Reflexion** (Shinn et al., 2023) demonstrated that LLMs can improve their outputs through iterative refinement loops. However, these methods typically rely on *explicit failure signals* or ground truth oracles to trigger correction. In open-ended reasoning tasks where such environmental feedback is absent, they often resort to "blind" reflection, leading to inefficient loops or the *self-correction fallacy* (Huang et al., 2023) where the model confidently validates its own errors.

To address this, *critique-and-refine* approaches like **CRITIC** (Gou et al., 2023) and self-critiquing pipelines (Saunders et al., 2022) introduce verification steps using external tools or self-generated critiques. Yet, these methods suffer from the "severity of hallucination": if the model lacks the knowledge to solve the problem, it often lacks the knowledge to critique it, resulting in sycophantic confirmation. More recent advances like **STeCa** (Wang et al., 2025) attempt to enforce calibration through trajectory-level reward modeling and supervised fine-tuning. While effective, STeCa requires expensive training on expert trajectories and is not easily adaptable to new base models. Our work aligns with the emerging trend of **Scaling Test-Time Compute** (Snell et al., 2024), but with a critical distinction in *activation*. Unlike Reflexion, which acts incessantly, or STeCa, which requires parameter updates, our **Dual-Process** framework offers a training-free, Pareto-optimal alternative. We utilize internal epistemic uncertainty as a calibrated "Stop" or "Switch" signal, ensuring that expensive reflective compute (System 2) is deployed *only* when the agent detects a genuine risk of failure, thereby balancing reliability with token efficiency.

A.2 Formal Mathematics in Problem Formulation

We provide the rigorous probabilistic definitions and derivations corresponding to the Problem Formulation in Section 2.3.

A.2.1 Forward Problem: Recursive Validity Estimation

In Equation (1), we defined the forward problem as estimating the trajectory validity $P(V_t|h_t)$. Here,

we explicate the recursive function f_{p}. Let $V_t \in \{0, 1\}$ be a binary random variable indicating that the trajectory up to step t is free of critical epistemic errors. By the chain rule of probability, the validity of the current state depends on the validity of the history and the correctness of the current action:

$$\begin{aligned} P(V_t = 1 \mid h_t) &= P(V_t = 1 \mid V_{t-1} = 1, a_t, h_{t-1}) \\ &\quad \cdot P(V_{t-1} = 1 \mid h_{t-1}) \\ &\approx \underbrace{P(\text{Corr}(a_t) \mid h_t)}_{\text{local conf. } c_t} \cdot \underbrace{P(V_{t-1} = 1 \mid h_{t-1})}_{\text{hist. validity}}. \end{aligned} \tag{5}$$

This recursive product implies that $P(V_t = 1)$ is monotonically non-increasing with respect to t. This mathematically formalizes the **Spiral of Hallucination**: a single failure at step k ($P(\text{Correct}_k) \approx 0$) drives the joint probability to zero for all $t > k$, permanently invalidating the trajectory. Our Forward UQ approximates this joint probability by aggregating verbalized confidence scores: $P(V_t|h_t) \approx \prod_{i=0}^{t} \hat{c}_i$ (or via a conservative minimum function $\min(\hat{c}_{0:t})$).

A.2.2 Inverse Problem: Latent Variable Calibration

We view this as *Test-Time Calibration* (Cobbe et al., 2021). When the forward process yields low confidence, we treat the generation of a reliable plan as an inverse search problem. By utilizing inference-time compute (e.g., reflection (Shinn et al., 2023)), we approximate the posterior distribution of correct reasoning without updating model parameters.

In Equation (2), we framed the inverse problem as maximizing the posterior via a latent reasoning path z. Here, we derive this form from Bayesian decision theory. We treat the task success as an optimality variable $\mathcal{O}$ (where $\mathcal{O} = 1$ implies Succ). We introduce a latent variable z representing the *reasoning trace* (e.g., a Chain-of-Thought or a Reflection explanation). We aim to find the action a^* that maximizes the posterior:

$$a^* = \arg\max_a P(a|h_t, \mathcal{O} = 1) \tag{6}$$

By marginalizing over the latent reasoning z, we expand the posterior:

$$\begin{aligned} P(a \mid h_t, \mathcal{O} = 1) &= \int P(a, z \mid h_t, \mathcal{O} = 1)\, dz \\ &= \int P(a \mid z, h_t) \\ &\quad \cdot P(z \mid h_t, \mathcal{O} = 1)\, dz. \end{aligned} \tag{7}$$

Using Bayes' rule on the second term:

$$P(z|h_t, \mathcal{O} = 1) \propto \underbrace{P(\mathcal{O} = 1|z, h_t)}_{\text{Likelihood (Consistency)}} \cdot \underbrace{\pi_{\text{prior}}(z|h_t)}_{\text{Generation Prior}} \tag{8}$$

Substituting this back, we recover the objective in Equation (2). Since the integral over all possible thoughts z is intractable, our **Best-of-N Reflection** strategy performs a Monte Carlo approximation:

1. **Sample:** Generate N reasoning paths $\{z_i\}_{i=1}^{N}$ from the prior $\pi(z|h_t)$.
2. **Reweight:** Estimate $P(\mathcal{O} = 1|z_i, h_t)$ using self-consistency or verbalized confidence as a proxy.
3. **Argmax:** Select the action a associated with the highest weighted z.

This derivation proves that our method is a particle-based approximation of the optimal inverse calibration.

A.3 Detailed Experimental Setup

A.3.1 Datasets and Evaluation Protocols

We selected three benchmarks that span the spectrum of agentic capabilities, ranging from rigid, logic-heavy planning to open-ended, creative synthesis. This diversity ensures that our Agentic UQ framework is not overfit to a specific modality but is generally applicable.

ALFWorld: Embodied Decision Making (Logic & Planning). ALFWorld aligns TextWorld with the ALFRED benchmark, creating a text-based simulated household. The agent must solve high-level goals (e.g., "clean the apple and put it in the fridge") by executing a sequence of low-level actions (e.g., `open fridge`, `put apple`).

- It represents **Deterministic Planning**. The environment is logically consistent but requires long-horizon dependency tracking. A single missing step (e.g., forgetting to open the fridge before putting the apple) causes failure. This tests the agent's ability to maintain *Cognitive Continuity*.
- We utilize the **Seen Evaluation Set**, comprising **140 unique environments**. This setting tests the agent's reliability in handling known domain structures. The evaluation metric is Success Rate (SR).

WebShop: Noisy Web Navigation (Robustness & Search). WebShop simulates a large-scale e-commerce website with 1.18 million products. The agent is given a user instruction (often containing implicit constraints like "under $50") and must search, browse, and select the correct product options.

- It represents **Stochastic Environment Interaction**. Unlike ALFWorld, WebShop is highly noisy: search engines return irrelevant results, and product descriptions are verbose and unstructured. This tests the agent's ability to use *Inverse UQ* to filter noise and verify information before committing.

- To maintain sample size consistency with ALFWorld, we randomly sampled **140 episodes** from the standard **Development Set**. The evaluation metric is Success Rate (SR).

DeepResearch Bench: Open-Ended Synthesis (Synthesis & Depth). Created by 100+ domain experts, this benchmark consists of **100 PhD-level research tasks** (50 in Chinese, 50 in English) spanning 22 distinct fields. The agent must perform autonomous web research to generate a comprehensive report.

- It represents **Open-Ended Reasoning**. There is no single "correct" sequence of actions. Success depends on the depth of insight and the coherence of the final report. This tests whether *Confidence-Aware Reflection* can drive deeper investigation rather than superficial summarization.

- We utilize the **Reference-based Adaptive Criteria-driven Evaluation (RACE)** framework (Du et al., 2025) with Dynamic Weighting. Unlike static grading, RACE dynamically adjusts evaluation criteria based on the reference report's complexity, assessing the agent on dimensions such as comprehensiveness, insight, instruction following, and readability. We focus on the RACE score to evaluate generation quality, excluding the FACT (citation) metric to isolate reasoning capabilities.

A.3.2 Baselines and Variants

Here we provide the exact operational details for each method to ensure reproducibility.

Baselines Implementation

- **ReAct** (Yao et al., 2022b): The standard System 1 baseline. The agent reasons and acts in an interleaved manner without explicit self-evaluation or reflection. We use the standard prompt template provided in the original paper. The temperature is set to 0.0 for deterministic greedy decoding.

- **Reflexion** (Shinn et al., 2023): A strong *inter-episode* learning baseline. It persists through verbal reinforcement from past failures to improve future trials. We evaluate Reflexion after 2 accumulated failure trials to measure its few-shot adaptation capability. We implement the standard Reflexion loop: `Act` $\rightarrow$ `Fail` $\rightarrow$ `Reflect` $\rightarrow$ `Store in Long-term Memory`. For a fair comparison with our inference-time method, we allow Reflexion to accumulate experience over 2 failure trials on the same instance before measuring the performance on the 3rd trial.

- **Self-Reflection** (Renze and Guven, 2024): An *intra-episode* baseline that triggers reflection on every step (or uses a heuristic). Unlike our method, it lacks confidence calibration and simply asks the model to "double-check" its action, serving as a baseline for unguided compute scaling. At every step, the agent generates a thought and action, then enters a mandatory "Check"phase: *"Review your proposed action. Is it correct? If not, generate a new one."* This method often suffers from "double-checking fatigue," where the model blindly confirms its own action, or "over-correction," where it changes a correct action due to anxiety.

- **CoT-SC** (Wang et al., 2022): A statistical ensemble baseline. For every decision step, we sample $N = 6$ independent trajectories (Action + Thought) with temperature $T = 0.7$. We apply majority voting on the final Action string. If there is a tie, we select the one with the highest average log-probability. This tests whether our gains stem purely from sampling diversity rather than targeted reflection.

Dual-Process Variants Implementation All our variants use the same base prompt structure but differ in their control flow and memory content.

- **Forward (`UAM`-Only):** The agent generates and stores verbalized confidence in memory to constrain future exploration, but *never* triggers the

System 2 reflection loop. This tests the efficacy of predictive uncertainty propagation.

  - *Process:* Step t generates $(a_t, \hat{c}_t, \hat{e}_t)$.
  - *Memory Update:* $\mathcal{M}_{t+1} \leftarrow \mathcal{M}_t \cup \{(o_t, a_t, \hat{c}_t, \hat{e}_t)\}$.
  - *Constraint:* The System 2 loop is strictly disabled (Threshold $\tau = -\infty$). The agent must rely on the presence of $\hat{e}_t$ in the context window to adjust its behavior for step $t+1$.

- **Inverse (UAR-Only):** The agent can trigger System 2 reflection to correct low-confidence steps, but does *not* persist the confidence metadata $(\hat{c}, \hat{e})$ into long-term memory. This tests the benefit of local correction without cognitive continuity.

  - *Process:* If $\hat{c}_t < \tau$, trigger Reflection + Best-of-N to get corrected action a_t^*.
  - *Memory Update (Crucial Difference):* $\mathcal{M}_{t+1} \leftarrow \mathcal{M}_t \cup \{(a_t^*, o_t)\}$.
  - *Constraint:* The confidence scores and explanations are **discarded** after the step is finalized. Future steps do not see the "cognitive meta-data"of the past, simulating a "forgetful" but capable agent.

- **Dual-Process (Full, AUQ):** The complete framework combining UAM propagation and UAR correction with adaptive switching.

  - *Process:* If $\hat{c}_t < \tau$, trigger Reflection.
  - *Memory Update:* $\mathcal{M}_{t+1} \leftarrow \mathcal{M}_t \cup \{(o_t, a_t^*, \hat{c}_{new}, \hat{e}_{new})\}$.
  - *Constraint:* Combines both local correction and long-term uncertainty propagation.

A.3.3 Implementation Details

Calculation of Consistency-Weighted Confidence

Given N sampled reasoning paths $\{r_1, ..., r_N\}$, each producing a final action a_i and a verbalized confidence $\hat{c}_i$, we first aggregate the paths into semantic clusters $C_1, ..., C_K$ such that all actions in a cluster are semantically equivalent. The score for a candidate action a is calculated as:

$$S(a) = \underbrace{\frac{|C_a|}{N}}_{\text{Consistency}} \times \underbrace{\frac{1}{|C_a|} \sum_{i \in C_a} \hat{c}_i}_{\text{Mean Confidence}} \tag{9}$$

This formulation separates diversity (consistency) from epistemic strength (confidence). Our consistency-weighted score $S_{cons}(a)$ (Equation 4) implicitly approximates the negative entropy of the System 2 distribution. A low consistency score indicates high variance among the N sampled paths, serving as a proxy for *Conflictual Uncertainty* (Epistemic), whereas a uniformly low $\hat{c}_t$ across consistent answers would indicate *Task Difficulty* (Aleatoric). By filtering candidates based on S_{cons}, we specifically target epistemic hallucinations. The semantic equivalence function $a_i \approx a_j$ is implemented differently depending on the task domain:

- Structured Environments (ALFWorld, WebShop): Since these environments require valid API calls, we employ *Normalized String Matching*. We extract the content within the `<action>` tags, convert to lowercase, and strip trailing whitespace.

$$\mathbb{I}(a_i, a_j) = \begin{cases} 1 & \text{if norm}(a_i) = \text{norm}(a_j) \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

- Open-Ended Reasoning (Deep Research): For high-level planning tasks where identical intents may be phrased differently (e.g., "Search for X then Y"vs. "First find X, followed by Y"), strict string matching is insufficient. We employ a *Model-Based Equivalence Check*. We define $\mathbb{I}(a_i, a_j) = 1$ if the model (System 1) predicts "Yes" to the prompt: *"Do these two plans represent the same core information-seeking strategy?"*. In practice, to reduce latency, we often strictly enforce output formatting (e.g., specific JSON keys) to revert to string matching where possible.

Model Specifications. We access proprietary models via their official APIs and host open-weights models using vLLM on a cluster of H200 GPUs. Table 4 lists the specific versions used in this study.

Hyperparameter Settings. We enforce strict hyperparameter control to ensure fair comparison. Table 5 details the specific settings for the Dual-Process framework.

Reflection and Memory Expansion Logic. The Inverse UQ process follows a hierarchical search:

1. **Parallel Sampling:** Upon triggering System 2, we generate $N = 3$ parallel responses using a temperature of 0.7 to encourage diverse reasoning paths.

Model Family	Specific Checkpoint/Version	Role in Experiments
OpenAI	`gpt-5.1-preview`	Strong Reasoning Baseline
	`gpt-4.1`	Strong Baseline
	`gpt-4o`	Efficiency Baseline
Google	`gemini-2.5-pro`	Long-Context Specialist
	`gemini-2.5-flash`	High-Speed/Low-Cost
Open-source LLMs	`Qwen3-235B-Instruct`	Strong Open Model
	`DeepSeek-V3.1-Chat`	Reasoning-Dense Open Model

Table 4: List of LLMs evaluated in our experiments.

Parameter	Value	Description
General Inference		
Max Trajectory Steps	50	Early stop for ALFWorld/WebShop
Temperature (Greedy)	0.0	Used for ReAct and System 1 (Forward)
Temperature (Sampling)	0.7	Used for System 2 (Inverse/Reflection)
Dual-Process Control		
Threshold τ (Standard)	$\{0.8, 0.85, 0.9, 0.95\}$	Ablated range; 0.85 is default
Threshold τ (Research)	$\{0.8, 0.85, 0.9, 0.95\}$	Ablated range; 0.95 for DeepResearch Bench
Inverse UQ (System 2)		
Sampling Width N	3	Number of parallel candidate paths
Reflection Depth D	3	Max iterative correction rounds per path
Memory Expansion	True	Triggered if $h =$ limit and $\hat{c} < \tau$ after reflection

Table 5: Hyperparameters for the Agentic UQ Framework.

2. **Iterative Refinement:** For each path, if the verbalized confidence is below τ, the agent is prompted to critique and refine its action. This repeats for a maximum of $D = 3$ turns.

3. **Expansion Trigger (Conditional):** In the *Limited Memory* setting ($h = 5$), if the best candidate from the reflection loop still fails to meet the threshold τ, the **Adaptive Memory Expansion** protocol is activated. The agent retrieves the full trajectory history from an external buffer and re-runs the reflection loop once. This ensures that expensive long-context processing is only reserved for the most stubborn uncertainties.

Integrate Agentic UQ with Enterprise Deep Research (EDR). The standard EDR architecture operates as a hierarchical multi-agent system (Prabhakar et al., 2025): a **Planner** ("master research agent") first decomposes the user topic into parallel sub-questions, which are then executed by **Researcher** ("specialized agent") agents using tools (e.g., Tavily/Google Search), followed by an **Analyst** ("research report") that synthesizes the results. We integrate our Dual-Process framework specifically at the **Planner-Researcher Interface**, utilizing a control strategy:

1. **Planner Augmentation:** We modify the Planner's system prompt to include our *Confidence Elicitation Protocol*. Instead of directly outputting a JSON list of sub-queries, the Planner is instructed to first output a `<confidence>` score and an `<explanation>` of potential knowledge gaps.

2. **Execution Interception (The Switch):** Before the Planner's output is parsed into the task queue for Researcher agents, the **`AUQ`** module acts as a gatekeeper:

   - **Pass-through (System 1):** If confidence $\geq \tau$, the decomposition is parsed immediately, and Researcher agents are dispatched in parallel. This incurs zero latency overhead.
   - **Intervention (System 2):** If confidence $< \tau$, the **`AUQ`** module blocks the dispatch of Researchers. It initiates the *Reflection Loop*, feeding the Planner's own explanation back to it to regenerate a refined plan (e.g., switching from General Search to Academic Search tools, or adding granular sub-queries).

3. **Seamless Integration:** Once the System 2 loop resolves the uncertainty (or reaches max

depth), the *corrected* decomposition plan is injected back into the EDR pipeline. The Researcher agents then execute this improved plan, unaware that an intervention occurred.

This architecture demonstrates that **AUQ** is **framework-agnostic**: it improves the quality of the EDR system's output (Insight/Comprehensiveness) solely by optimizing the planning instructions, without requiring changes to the complex asynchronous search or synthesis logic.

A.3.4 Evaluation Metrics and Protocols

Trajectory-Level Calibration Metrics. Standard token-level metrics are insufficient for sequential tasks. We define a trajectory confidence sequence $\mathbf{c} = \{\hat{c}_1, \ldots, \hat{c}_T\}$. We map this to a scalar trajectory belief $C(\tau) = \Phi(\mathbf{c})$ using three aggregation strategies: *End-State* (Φ_{last}), *Overall Quality* (Φ_{avg}), and *Process Reliability* (Φ_{min}).

- **Trajectory-ECE (T-ECE).** We partition test trajectories into M bins $\{\mathcal{B}_m\}_{m=1}^{M}$ based on their aggregated belief $C(\tau)$. T-ECE measures the weighted absolute difference between confidence and accuracy:

$$\text{T-ECE}_\Phi = \sum_{m=1}^{M} \frac{|\mathcal{B}_m|}{N} \left| \underbrace{\text{acc}(\mathcal{B}_m)}_{\text{Avg } Y} - \underbrace{\text{conf}(\mathcal{B}_m)}_{\text{Avg } C(\tau)} \right| \tag{11}$$

  where $Y \in \{0, 1\}$ is the binary task success.

- **Trajectory Brier Score (T-BS).** To jointly evaluate calibration and sharpness (decisiveness), we utilize the Mean Squared Error of the probabilistic prediction:

$$\text{T-BS}_\Phi = \frac{1}{N} \sum_{i=1}^{N} (C(\tau^{(i)}) - Y^{(i)})^2 \tag{12}$$

- **Discriminative AUROC.** We calculate the AUROC by treating the aggregated confidence $C(\tau)$ as a binary classifier score for task success. A score of 0.5 indicates random guessing, while 1.0 indicates perfect separation between success and failure trajectories.

Deep Research Judge Protocol. Evaluating open-ended research reports requires nuanced judgment. We utilize **Gemini-2.5-Pro** as the evaluator. The model follows the RACE protocol (Du et al., 2025), scoring on a 1-10 Likert scale across four dimensions (comprehensiveness, insight/depth, instruction following, readability), which are then normalized to a 0-100 scale.

A.4 Dual-Process Agentic UQ Framework (AUQ)

We provide the formal execution flow of the Dual-Process Agentic UQ framework in Algorithm 1. The algorithm proceeds in three phases at each time step t:

- **System 1 - Uncertainty Propagation:** The agent observes the current state o_t and conditions on the accumulated uncertainty-aware memory $\mathcal{M}_t$. It jointly generates a tentative action $\hat{a}_t$, a scalar confidence $\hat{c}_t$, and a semantic explanation $\hat{e}_t$. The explanation serves as a "rationale cue" for potential debugging.

- **System 2 - Uncertainty Reflection:** The framework evaluates the confidence against the threshold τ.
  - If $\hat{c}_t \geq \tau$, the system trusts the fast intuition and proceeds immediately.
  - If $\hat{c}_t < \tau$, System 2 is activated. The agent uses the generated explanation $\hat{e}_t$ as a specific query to guide a *Best-of-N* sampling process. The final action a_t is selected via confidence-weighted consistency voting, effectively filtering out hallucinated paths.

- **Memory Consolidation:** Crucially, the final executed tuple $(o_t, a_t, \hat{c}_t, \hat{e}_t)$ is appended to $\mathcal{M}$. This ensures that future steps are conditioned on the agent's past epistemic states, enabling the "Cognitive Damper" effect where past doubts influence future caution.

A.5 Additional Experimental Results and Analysis

A.5.1 Extended Analysis of Internal Dynamics and Risks

The scatter plots in Figure 3 (Right) revealed a counter-intuitive phenomenon where failure cases often exhibit larger confidence gains (Δ) than successes. To explain this, we categorize the System 2 intervention outcomes into three distinct modes based on initial confidence c_{init} and final status:

1. Validation (High c_{init} → High c_{final}, Success). This is the dominant mode for successful trajectories. The agent starts with high confidence (typically $c_{init} > 0.9$). The reflection loop serves as a

Algorithm 1 Dual-Process Agentic Uncertainty Quantification

Require: Task Instruction I, Threshold τ, Sampling Count N, Max Steps T_{max}

Initialize: Memory $\mathcal{M}_0 \leftarrow \emptyset$, Observation $o_0 \leftarrow$ Env.reset(I)
for $t = 0$ **to** T_{max} **do**
  // Phase 1: System 1 Fast Execution
  Construct prompt $P_t \leftarrow f_{prompt}(I, \mathcal{M}_t, o_t)$
  Generate initial proposal:
    $\hat{a}_t, \hat{c}_t, \hat{e}_t \sim \pi_\theta(\cdot|P_t)$
  // Phase 2: Uncertainty Switch
  if $\hat{c}_t \geq \tau$ **then**
    // High Confidence: Pass-through
    Set final action $a_t \leftarrow \hat{a}_t$
  else
    // Low Confidence: System 2 Reflection (Triggered)
    Define candidate set $\mathcal{A}_{cand} \leftarrow \emptyset$
    Construct reflection prompt P_{ref} using rational cue $\hat{e}_t$
    for $n = 1$ **to** N **do**
      Sample reasoning path: $r^{(n)}, a^{(n)}, \hat{c}^{(n)} \sim \pi_\theta(\cdot|P_{ref})$
      Add to candidates: $\mathcal{A}_{cand} \leftarrow \mathcal{A}_{cand} \cup \{(a^{(n)}, \hat{c}^{(n)})\}$
    end for
    // Inverse UQ: Consistency-Weighted Selection
    Calculate score for each unique action a:
      $S(a) = \sum_{(a^{(i)}, \hat{c}^{(i)}) \in \mathcal{A}_{cand}} \mathbb{I}(a^{(i)} \approx a) \cdot \hat{c}^{(i)}$
    Select optimal action: $a_t \leftarrow \arg\max_a S(a)$
    Update confidence/explanation to match selected path's values
  end if
  // Phase 3: Execution & Memory Update
  Execute action: o_{t+1}, done $\leftarrow$ Env.step(a_t)
  Update Uncertainty-Aware Memory:
    $\mathcal{M}_{t+1} \leftarrow \mathcal{M}_t \cup \{(o_t, a_t, \hat{c}_t, \hat{e}_t)\}$
  if done **then**
    Break
  end if
end for

sanity check, confirming the plan is sound. Since the confidence is already near the ceiling (1.0), the potential gain Δ is minimal. This explains the tight clustering of green points in the top-right of the scatter plots.

2. True Correction (Low c_{init} $\rightarrow$ High c_{final}, Success). This represents the ideal System 2 intervention. The agent starts in the "Ambiguous Zone"($c_{init} \approx 0.6 - 0.8$), correctly identifying a gap in its reasoning. Reflection synthesizes a corrected plan, boosting confidence significantly (e.g., $\Delta \approx +0.3$).

3. Delusional Confirmation (Low c_{init} $\rightarrow$ High c_{final}, Failure). This is the root of the "delusional gap."In challenging situations, the agent initially has low confidence. However, instead of realizing the task is impossible, it generates a seemingly plausible but actually incorrect explanation through its reflection mechanism. The agent then adopts this fabricated solution with extremely high confidence. This significant increase in confidence (e.g., from 0.5 to 0.9) distorts the average statistics, creating the illusion that failing agents are "more confident"in their improvements than successful agents.

Threshold Sensitivity and Calibration Profiles

We further analyze how the sensitivity threshold τ and model scale influence the trade-off between the correction modes defined above.

The Trigger-Efficiency Trade-off. The choice of threshold modulates the system's behavior:

- **Low Threshold ($\tau = 0.8$, "Loose Filter"):** The mechanism only activates for obvious errors. While it minimizes the risk of Regression (breaking good steps), it misses many "True Corrections" in the Ambiguous Zone, limiting the overall Success Rate gain.
- **High Threshold ($\tau = 0.95$, "Strict Forcer"):** The system acts aggressively, verifying even moderately confident steps. This maximizes the error correction rate (recall), but also increases the probability of incorrect confirmations. Crucially, however, Figure 7 shows that **System 2 rarely downgrades a correct high-confidence step** (few green points below the diagonal). This "safe failure"characteristic indicates that increasing sensitivity increases computational cost but does not significantly compromise reliability.

Model-Specific Calibration. Comparing GPT-4o and GPT-5.1 reveals distinct personalities:

- **GPT-4o (Figure 7):** Shows a wider spread in initial confidence. The reflection mechanism often functions as a "booster," lifting confidence from the ambiguous zone to the certain zone.
- **GPT-5.1 (Figure 8):** Exhibits higher intrinsic confidence. System 2 serves less as a confidence booster and more as a "validity check."The gains are subtler numerically but represent the resolution of complex, long-tail uncertainties.

Convergence of Belief. Across all settings, we observe *Belief Polarization*. Before reflection, confidence scores are distributed along a continuum. After reflection, they snap towards extremes (0.0 or 1.0). This aligns with our Brier Score analysis, confirming that Dual-Process reduces epistemic entropy, converting "unknowns" into either "known knowns" (Success) or "known unknowns" (Decisive Stop).

Comparative Outcome Analysis (`AUQ` vs. ReAct)

To rigorously quantify the net utility of the framework, we perform a quadrant analysis of trajectory outcomes compared to the ReAct baseline (visualized in Figure 9).

- **Shared Success (60.0% - "The Easy"): Decisive Efficiency.** These are tasks solvable by System 1 alone. A critical finding is that `AUQ` is **more efficient** in this regime (13.7 steps) compared to ReAct (16.2 steps). By utilizing verbalized confidence, our agent can detect goal satisfaction with high certainty and trigger the `stop` action earlier.
- **Shared Failure (22.1% - "The Intractable"):** These represent tasks beyond the underlying model's capability, where both agents fail (often hitting the max step limit). This defines the hard upper bound of the LLM's reasoning power.
- **Correction (14.3% - "The Net Gain"):** This quadrant represents the primary contribution of Inverse UQ. ReAct failed these cases (often due to early hallucinations spiraling out of control), while `AUQ` successfully recovered them via reflection. The higher average step count (18.1) confirms that these victories required active computational investment.
- **Regression (3.6% - "The Cost"):** These are cases where ReAct succeeded, but `AUQ` failed. This highlights the risk of **Over-reflection**: a correct System 1 intuition can occasionally be undermined by a hyper-critical System 2. However, the **Net Repair Ratio of roughly 4:1** (14.3% Correction vs. 3.6% Regression) validates that our switching logic is highly conservative and effective.

A.5.2 Error Analysis and Failure Modes

While `AUQ` achieves a high Net Repair Ratio, we conduct a rigorous analysis of the failure cases to understand the limitations of our framework. We categorize errors into two types: *Regressions* (where `AUQ` degrades performance) and *Shared Failures* (where both systems fail).

Regressions (3.6%): The Cost of Over-Correction

The 3.6% regression rate primarily stems from the **"Over-Correction Fallacy."** In these instances, System 1 initially proposed a correct action, but the uncertainty score ($\hat{c}$) marginally dipped below τ, triggering System 2. Qualitative inspection reveals that during reflection, the model sometimes hallucinates non-existent constraints or interprets the "Reflection Request" as an implicit signal that its initial thought was wrong (a phenomenon known as *sycophancy*).

- **Example (ALFWorld):** The agent correctly identified a "Pen" on the desk. However, low

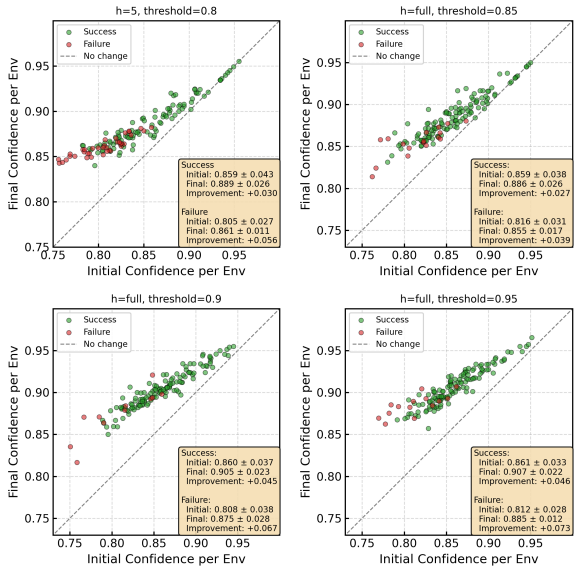

Figure 7: **Detailed Confidence Dynamics for GPT-4o across Thresholds.** We visualize the reflection gain for $\tau \in \{0.8, 0.85, 0.9, 0.95\}$. Lower thresholds (e.g., 0.8) trigger sparsely on obvious errors, while higher thresholds (e.g., 0.95) induce broad recalibration. The positive shift (points above the diagonal) remains consistent across all settings.

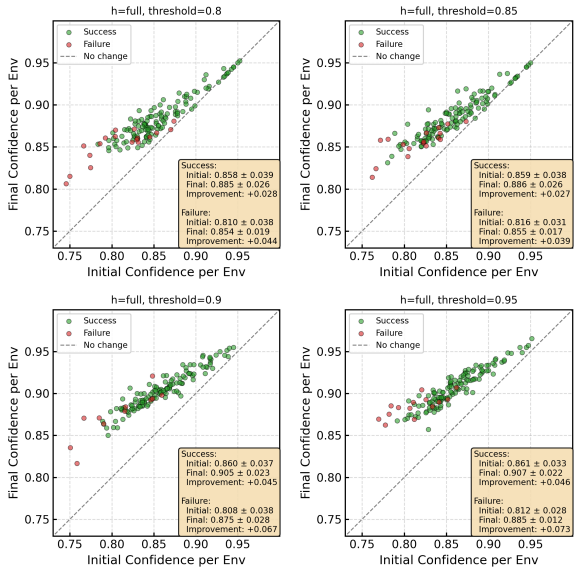

Figure 8: Detailed Confidence Dynamics for GPT-5.1 across Thresholds. Compared to GPT-4o, GPT-5.1 exhibits tighter clustering and higher intrinsic confidence. Even at strict thresholds ($\tau = 0.95$), the mechanism effectively nudges uncertain predictions towards certainty (1.0) or rejection (0.0).

confidence triggered reflection, leading the agent to doubt its perception ("Maybe it's a pencil?") and navigate away to a drawer, ultimately running out of steps.

Figure 9: **Comparative Outcomes and Efficiency Analysis (AUQ vs. ReAct).** (Left) Outcome distribution matrix. **AUQ** demonstrates a strong positive net repair: correcting **14.3%** of ReAct's failures while only regressing on **3.6%** of its successes (a 4:1 ratio). (Right) Average steps per category. Surprisingly, for commonly successful trajectories ("Easy" cases), **AUQ** is more efficient (13.7 steps vs. 16.2), indicating that high confidence enables decisive early stopping.

- **Mitigation:** This suggests that future work could benefit from a "Confidence Hysteresis" or a stricter verification step inside System 2 to confirm that the new plan is strictly better than the old one before switching.

Shared Failures (22.1%): The Hard Capability Boundary

The shared failure cases represent the fundamental capability upper bound of the base LLM. **AUQ** acts as a reasoning amplifier, but it cannot generate knowledge that does not exist in the model's weights or the environment. This analysis confirms that while **AUQ** significantly improves *calibration* and *reasoning robustness*, it does not solve fundamental *grounding* or *retrieval* impossibilities.

A.5.3 Extended Cost-Efficiency Analysis

In Section 4.3, we argued that Dual-Process optimizes the computational cost of agentic reasoning. Here, we provide a detailed breakdown of the metrics presented in Table 6.

Efficiency via Prevention

A naive assumption is that reflection always adds cost. However, Table 6 reveals a more nuanced reality.

- **The Cost of Failure:** Standard ReAct agents are remarkably inefficient when they fail. They tend to exhaust the maximum step limit (e.g., 50 steps) chasing hallucinations, accumulating a massive "Failure Cost"without delivering value.

- **The Savings of Correction:** `AUQ` introduces a "Reflection Overhead"(extra API calls per step). However, by correcting an error at Step 5, it prevents the subsequent 45 steps of futile wandering. For difficult trajectories, this **Early Correction** mechanism effectively reduces the *Total Steps to Solution*.

Comparison with Reflexion

It is also crucial to compare costs against *inter-episode* baselines like Reflexion. Reflexion achieves high performance by running multiple full trials (e.g., Trial 1 Fail → Reflect → Trial 2 Fail → ...). This effectively multiplies the cost by the number of trials (K). In contrast, our Dual-Process approach performs **Inference-Time Correction** within a single episode. Table 6 suggests that `AUQ` achieves comparable or superior results to multi-trial Reflexion but at a fraction of the total token consumption (since it does not need to re-generate the entire valid prefix of a trajectory).

A.5.4 Discussion: Why Verbalized Confidence?

A critical design choice in our framework is the reliance on verbalized confidence ($\hat{c}$) and explanations ($\hat{e}$) rather than token-level probabilities (logits). While we acknowledge that verbalization is not perfectly calibrated in smaller models (as discussed in Limitations), it represents the most practical and effective metric for agentic systems for the following key reasons.

Model Agnosticism and Accessibility. The primary goal of this work is to enhance strong agents powered by frontier models (e.g., GPT-5.1, Gemini-2.5-Pro). Currently, the majority of these top-tier models operate as "black boxes"via APIs that do not expose access to log-probabilities (Achiam et al., 2023). Relying on logits would restrict our framework to open-weights models (e.g., Qwen, Deepseek, Llama), severely limiting its applicability to the most capable agents deployed in real-world scenarios. Verbalized uncertainty serves as a universal interface compatible with any instruction-tuned LLM.

Semantic vs. Statistical Uncertainty. There is a fundamental misalignment between *token-level probability* and *reasoning uncertainty*.

- **The Token Trap:** An LLM can be statistically confident (high logit) in predicting the next grammatical token (e.g., "The"), while being epistemically uncertain about the factual content. Averaging log-probabilities over a long Chain-of-Thought (CoT) sequence introduces significant noise and length bias, often washing out the signal of a specific logical flaw.
- **The Semantic Summary:** Verbalized confidence acts as a metacognitive compression. It forces the model to introspect on the *entirety* of its reasoning step and output a scalar that represents semantic validity rather than statistical fluency. As noted by Lin et al. (2022), verbalized confidence often correlates better with correctness for reasoning tasks than raw probabilities.

The Necessity of Explanations for System 2. Crucially, our Dual-Process framework requires uncertainty to be *actionable*. A raw logit score (e.g., 0.45) serves only as a switch; it provides no information on *how* to fix the error. By eliciting a verbal explanation ($\hat{e}$), we obtain a "Rational Cue"(e.g., "I am unsure about the specific date") that guides the System 2 reflection loop. This semantic signal allows the agent to target its query expansion or tool use, a capability impossible to achieve with scalar logits alone.

A.5.5 Discussion: On the Dynamics of Thresholding

A recurring question concerns the selection of the confidence threshold τ and its static nature. We address the implications of threshold sensitivity and adaptability below.

Why Static Thresholding? Our current framework employs a static τ per task to maintain a *training-free* architecture. Implementing a dynamic, instance-level threshold (e.g., τ_t varying by step type) would typically require training a separate *meta-controller* or reward model (as seen in STeCa (Wang et al., 2025)), which introduces significant computational overhead and data dependency. Our static approach serves as a strong baseline, demonstrating that even a fixed sensitivity gate can yield superior performance when paired with the powerful mechanism of System 1.

The Heterogeneity Challenge. We acknowledge that heterogeneous tasks involve steps with varying risk profiles; for instance, a "Search" action might tolerate higher uncertainty than a "Final Answer" action. A static τ forces a uniform risk tolerance. However, our sensitivity analysis (Figure 2 and 6)

Threshold (λ)	Success Rate (%)	Trigger Rate (%)	Conf. Inc. Rate (%)	Initial Conf.	Final Conf.	Total LLM API Calls	Total Steps
Setting: Limited History ($h = 5$)							
0.80	72.9	46.5	99.5	0.743	0.848	1850	3551
0.85	82.9	43.3	98.8	0.750	0.871	1815	3094
0.90	87.1	78.6	95.0	0.807	0.898	3085	2879
0.95	88.6	85.5	94.2	0.817	0.917	4149	2868
Setting: Full History ($h = full$)							
0.80	87.1	35.2	95.0	0.737	0.833	1013	2880
0.85	87.9	35.5	95.7	0.741	0.835	1000	2814
0.90	92.1	77.3	92.7	0.807	0.875	2097	2717
0.95	90.0	82.8	91.5	0.815	0.882	2201	2661

Table 6: **Cost-Efficiency Ablation Analysis.** We compare the computational overhead (Average Steps, API Calls, and Token Cost) of ReAct vs. Dual-Process (**AUQ**) across different thresholds. While **AUQ** introduces an inference overhead for reflection, it achieves a superior **Conversion Rate**: ReAct often wastes tokens on long, failing trajectories (Avg Steps 26.4 for failure), whereas **AUQ** invests tokens to secure successes. At $\tau = 0.9$, **AUQ** represents the optimal trade-off, delivering maximal reliability gains before the cost curve becomes exponential.

suggests that the system is robust to this limitation. We hypothesize that this is because System 1's *explanation* ($\hat{e}$) acts as a secondary, semantic filter: even if the threshold is slightly misaligned, the explicit rationale generation forces the model to ground its confidence, partially mitigating the rigidity of the scalar cutoff.

Future Direction: Adaptive Risk Budgeting. To fully address task heterogeneity, future work could model τ as a function of the *step type* (a_t) and the remaining *inference budget* (B). For example, an *Adaptive Risk Budgeting* module could lower τ (be more cautious) for high-stakes actions (e.g., API calls that expend money) and raise τ (be more lenient) for reversible reasoning steps.

A.5.6 Discussion: Efficiency and Cost Analysis

A rigorous evaluation of agentic systems requires analyzing not just token consumption, but also the *wall-clock latency* and the *economic cost of reliability*. Here, we address the trade-offs of our Dual-Process architecture.

Inference vs. Environmental Latency

Critically, the latency profile of an agent depends heavily on whether the task is *compute-bound* or *I/O-bound*.

- **System 2 Overhead (T_{inf}):** Generating a reflection and performing Best-of-N sampling typically consumes 0.5–2 seconds of GPU inference time per intervention.
- **Environmental Latency (T_{env}):** In real-world tasks like Deep Research, tool execution (e.g., scraping a heavy webpage, waiting for a server response) often takes multiple seconds.

Our framework trades "cheap" inference time to prevent "expensive" environmental interactions. For example, if a System 2 intervention prevents the agent from executing a futile search plan (which would incur $5 \times T_{env}$ latency), the net Time-to-Solution (TTS) decreases, even if the reflection itself added non-zero inference time.

Theoretical Cost Model

We formalize this trade-off. Let L be the trajectory length of a baseline agent. The total time is $T_{base} = L(T_{inf} + T_{env})$. For our Dual-Process agent, let p be the fraction of steps triggering System 2, and k be the overhead factor during reflection (e.g., sampling N paths). Due to better planning, the trajectory length is reduced to $L' < L$. The condition for **AUQ** to be faster is:

$$L' \cdot \Big[(1-p)(T_{\text{inf}} + T_{\text{env}}) + p(kT_{\text{inf}} + T_{\text{env}})\Big] < L(T_{\text{inf}} + T_{\text{env}}). \tag{13}$$

In I/O-heavy domains where $T_{env} \gg T_{inf}$ (e.g., Deep Research), this inequality holds easily even for modest reductions in trajectory length (L'), as the dominance of T_{env} dilutes the impact of inference overhead (kT_{inf}).

The Economics of Reliability: Success-Weighted Cost

A naive comparison of raw token costs is misleading because it ignores the *penalty of failure*. A baseline trajectory that fails after consuming 50 steps represents a 100% waste of resources. We

propose analyzing the **Effective Cost per Success** (Cost_{eff}):

$$\begin{aligned}\text{Cost}_{\text{eff}} &= \frac{\text{Total Cost of All Attempts}}{\text{Number of Successful Tasks}} \\ &= \frac{\text{Avg. Cost per Trajectory}}{\text{Success Rate (SR)}}\end{aligned} \tag{14}$$

In ALFWorld, although **AUQ** increases the per-trajectory token cost by $\approx 1.4\times$ due to reflection, it boosts the Success Rate significantly (e.g., +20%). This reduces the Cost_{eff}, meaning that to achieve a fixed number of solved tasks, **AUQ** is financially more efficient than the baseline at the campaign level.

A.6 Prompt Templates

We present the complete prompt templates used in our experiments. Our framework requires no parameter updates; all capabilities are elicited through structured in-context learning.

A.6.1 Baseline Agent

The baseline agent utilizes a standard ReAct-style prompt. It reasons and acts but does not generate or attend to uncertain information.

Baseline System Prompt

```
You are an expert agent operating in the ALFRED Embodied
Environment. Your task is to: {task_description}

Prior to this step, you have already taken {step_count}
step(s). Below are the most recent {history_length}
observations and the corresponding actions you took:
{action_history}

You are now at step {current_step} and your
current observation is: {current_observation} Your
admissible actions of the current situation are:
[{admissible_actions}].

Now it's your turn to take an action. You should
first reason step-by-step about the current situation.
This reasoning process MUST be enclosed within <think>
</think> tags. Once you've finished your reasoning, you
should choose an admissible action for the current step
and present it within <action> </action> tags.
```

A.6.2 System 1: Uncertainty-Aware Memory (Forward UQ)

To enable Forward UQ, we inject a *Confidence Elicitation Instruction* and modify the history format to propagate uncertainty states.

Confidence Elicitation Instruction

This instruction is appended to the user prompt at every inference step to extract $\hat{c}$ and $\hat{e}$.

Elicitation Suffix

```
After your action, you MUST provide:
1. Your confidence level (0.0-1.0) in
<confidence>...</confidence> tags
2. An explanation of your confidence in
<explanation>...</explanation> tags
- Explain what makes you confident
- Explain what concerns or uncertainties you have
- What information might be missing or unclear
- What alternative actions you considered
- DO NOT output empty <explanation></explanation> tags
- you MUST provide actual text inside
```

Uncertainty Propagation Formats

We define how past uncertainty is formatted in the `{action_history}` slot.

Variant A: Confidence Score Only. A minimal constraint setting where only the scalar score is retained.

```
... Step {t-1}: Observation: {obs_prev} Action:
<think>...</think> <action>examine desk 1</action>
<confidence>0.85</confidence> ...
```

Variant B: Semantic Propagation (Confidence + Explanation). Our primary **UAM** setting where the full explanation is retained in the context window.

```
... Step {t-1}: Observation: {obs_prev}
Action: <think>I should check the desk
first.</think> <action>examine desk 1</action>
<confidence>0.65</confidence> <explanation>I see a
bowl, but I do not see the desklamp required for the
task. It might be in a closed container, or I might
need to look elsewhere.</explanation> ...
```

A.6.3 System 2: Uncertainty-Aware Reflection (Inverse UQ)

When confidence falls below the threshold ($\hat{c} < \tau$), we trigger the reflection mechanism.

Uncertainty-Aware Reflection Prompt

This prompt feeds the agent's own explanation back to it as a "Rational Cue" for correction.

Reflection Prompt Template

```
**REFLECTION REQUEST**
Your previous response had confidence {confidence}. You
mentioned the following concerns:
{explanation}

Given these concerns and the full context below,
please reconsider your reasoning and provide a better
response.
—
**FULL CONTEXT (including history):**
{full_context}
—
**YOUR PREVIOUS RESPONSE:**
{previous_response}
—
**REFLECTION INSTRUCTIONS:**
Please provide a NEW response that addresses the
confidence concerns you mentioned. Your new response
should:
1. Include updated reasoning in <think>...</think> tags
2. Include a new action in <action>...</action> tags
3. Include your new confidence level in
<confidence>...</confidence> tags (0.0-1.0)
4. Include an updated explanation in
<explanation>...</explanation> tags - Specifically
explain how you addressed the previous concerns
- What makes you more or less confident now
```

Memory Expansion Prompt

For tasks requiring long-range dependency resolution, we use a specialized prompt that emphasizes retrieving information from the extended history.

Memory Expansion Instructions

```
**MEMORY EXPANSION INSTRUCTIONS:**

Please carefully review the expanded history above and
use it to address the confidence concerns you mentioned.
Your new response should:
1. Include updated reasoning in <think>...</think> tags
2. Include a new action in <action>...</action> tags
3. Include your new confidence level in
<confidence>...</confidence> tags (0.0-1.0)
4. Include an updated explanation in
<explanation>...</explanation> tags

**Specifically explain**: How the expanded history
({history_length} steps) influenced your confidence
- What information from the history was most useful (or
not useful)
- What makes you more or less confident now compared to
before
- What concerns remain or have been resolved

Think carefully about:
- What patterns or information in the expanded history
are relevant to your current decision
- How the expanded context helps address your previous
concerns
- Whether there are better actions to take based on the
full history
- What makes you more or less confident now
- Be critical: not all historical information may be
useful; use your judgment
```

A.7 Qualitative Examples

A.7.1 Qualitative Analysis: Embodied Decision Making (ALFWorld)

In embodied tasks, agents often succumb to "Action Loops", repeatedly executing the same sub-optimal actions due to local grounding errors. We illustrate how **AUQ** breaks such a loop in the task: **"examine a bowl with the desklamp."**

Task Context. The agent starts at `desk 1`, where a `bowl 1` is present. However, the required `desklamp` is not at the current location and must be found elsewhere in the room.

Trajectory Divergence Analysis

Table 7 contrasts the baseline agent's trajectory with the **AUQ** agent. The divergence occurs at **Step 3**, where the Baseline commits to a local loop, while **AUQ** triggers a reflection to expand its search horizon. The full results are shown in Table 8.

The Critical Moment: Breaking the Loop. At Step 3, the **AUQ** agent initially considered repeating a local action (`look` or `examine`) with low confidence ($\hat{c} = 0.80$). This triggered the System 2 resolver.

System 2 Reflection Trace (Step 3)

System 1 Proposal: `look` ($\hat{c} = 0.80$)
Uncertainty Explanation: "I am at desk 1 and I see the bowl required for the task, but I do not see the desklamp. Simply looking again or examining the desk is unlikely to help since I have already done so. I am uncertain about the location of the desklamp."
Reflection (Best-of-N):

- Path A (Action: examine desk 1): "Re-examine to ensure I didn't miss it."→ Rejected (Redundant).
- Path B (Action: go to sidetable 1): "Search nearest receptacle."→ Plausible.
- Path C (Action: go to shelf 1): "Explore vertical storage likely to hold lamps."→ **Selected** ($\hat{c}_{new} = 0.85$).

Final Decision: Switch strategy from exploitation (interacting with bowl) to exploration (finding lamp). Action: go to shelf 1.

Key Improvements Analysis

1. **Precondition Verification:** The Baseline agent focused immediately on the `bowl` (which it found) and ignored the `desklamp` (which it lacked). **AUQ**'s uncertainty signal flagged the missing object ("Where is the lamp?") before committing to manipulation actions.

2. **Exploration vs. Exploitation Switch:**

   - **Baseline:** Fell into a "Greedy Trap." Because it had the bowl, high probability tokens favored interacting with it (`take`, `put`), leading to a loop of moving the bowl between the desk and shelf.
   - **AUQ:** The low confidence in Step 3 inhibits the high-likelihood but incorrect local ac-

Step	Baseline Action	Conf.	Outcome / Observation	AUQ Action (System 2)
0	`go to desk 1`	0.90	On desk 1: alarmclock, **bowl 1**, mug...	`go to desk 1`
1	`examine desk 1`	0.85	(Same observation)	`examine desk 1`
2	`look`	0.8	Facing desk 1. Next to it: nothing.	`look`
3	`examine desk 1`	**0.8**	*(Repetitive)* On desk 1: bowl 1...	**Trigger Reflection** → `go to shelf 1`
...	...	...	...	...
End	**FAILED (Timeout)**	-	Stuck in manipulation loop.	**SUCCESS (Step 8)**

Table 7: **Trajectory Divergence in ALFWorld (env_127).** The Baseline agent (left) gets stuck interacting with the bowl at the desk without the lamp. The **AUQ** agent (right) detects the missing precondition at Step 3, triggering exploration to shelf/dresser.

tion. System 2 realized that holding the bowl is useless without the lamp, forcing a navigation action (`go to shelf/dresser`).

3. **Efficiency:** **AUQ** completed the task in 8 steps. The Baseline wasted 30+ steps moving the bowl back and forth until the context window was exhausted.

A.7.2 Qualitative Case Study of Deep Research (Chinese Query)

We present a real-world example from the Deep Research Bench to illustrate how our Dual-Process framework corrects epistemic errors through self-diagnosis.

User Query: Collect and organize the current actual income and financial status of China's 9 social strata in 2025, specifically researching the characteristics, population size, and financial power of the middle class.

Phase 1: System 1 (Forward Pass)

The agent generates an initial decomposition plan using the standard 'general_search'.

- **Draft Plan:** Generate 4 queries covering income distribution and middle-class definitions.
- **Verbalized Confidence:**

  `<confidence>0.9</confidence>`
- **Verbalized Explanation (The Diagnosis):**

  "I am confident the queries cover the core elements. However, I have specific uncertainties: (1) The specific classification standards for '9 social strata' may vary across institutions; (2) Middle class definitions vary in academia; (3) Official 2025 statistical data might not be fully released yet; (4) I missed specific subdivisions like regional or industry differences."

Switching Logic: With $\tau = 0.95$, the system detects $\hat{c}_{init}(0.9) < \tau$. **System 2 is triggered.**

Phase 2: System 2 (Inverse Reflection)

The agent enters the Reflection Loop (Best-of-N). It uses the explanation above to refine its strategy.

- **Refining Logic:**
  1. *Addressing Concern #1 & #2 (Definitions):* The agent explicitly adds "Official Definition"and "Academic Standards"to keywords.
  2. ***Tool Upgrade:*** Crucially, the agent switches from `general_search` to `academic_search` for the definition-related queries to ensure authority.
  3. *Addressing Concern #4 (Missing Subdivisions):* It adds a new sub-query specifically for "Regional Differences and Industry Distribution".
- **Final Plan (Selected from Best-of-N):**
  - Query 1: "China 9 social strata classification standards **official definition**..."(`Tool: academic_search`)
  - Query 2: "China middle class definition **quantitative standards**..."(`Tool: academic_search`)
  - Query 3: "China middle class **regional differences and industry distribution** 2025"(`Tool: general_search`)
  - ... (other queries)
- **Final Confidence:**

  `<confidence>0.95</confidence>`

	ReAct (Baseline)		AUQ (Ours)			
Step	**Action**	**Confidence**	**Initial Action**	**Initial Confidence**	**Action after UAR**	**Confidence After UAR**
0	go to desk 1	0.9	go to desk 1	0.9	go to desk 1	0.9
1	examine desk 1	0.8	examine desk 1	0.8	look	**0.9**
2	look	0.8	look	0.8	examine desk 1	**0.9**
3	examine desk 1	0.8	look	0.8	go to shelf 1	**0.85**
4	take bowl 1 from desk 1	0.85	go to desk 1	0.7	examine shelf 2	**0.8**
5	examine bowl 1	0.95	go to shelf 3	0.8	go to dresser 1	**0.85**
6	move bowl 1 to desk 1	0.85	go to desk 1	0.9	go to desk 1	0.9
7	examine desk 1	0.85	take bowl 1 from desk 1	0.9	take bowl 1 from desk 1	0.9
8	look	0.75	go to dresser 1	0.9	go to dresser 1	0.9
9	examine desk 1	0.85	use desklamp 1	0.9	use desklamp 1	0.9
10	look	0.8	**Success (stop)**	–	**Success (stop)**	–
11	examine desk 1	0.7				
12	take bowl 1 from desk 1	0.75				
13	examine bowl 1	0.85				
14	move bowl 1 to desk 1	0.8				
15	examine desk 1	0.85				
16	examine desk 1	0.75				
17	examine desk 1	0.75				
18	take bowl 1 from desk 1	0.7				
19	examine bowl 1	0.85				
20	examine desk 1	0.75				
21	move bowl 1 to desk 1	0.75				
22	examine desk 1	0.85				
23	examine desk 1	0.65				
24	look	0.8				
25	examine desk 1	0.75				
26	take bowl 1 from desk 1	0.85				
27	examine bowl 1	0.85				
28	look	0.75				
29	examine bowl 1	0.85				
30	examine desk 1	0.85				
31	move bowl 1 to desk 1	0.75				
32	examine desk 1	0.8				
33	examine desk 1	0.8				
34	look	0.7				
35	examine desk 1	0.75				
36	look	0.7				
37	examine desk 1	0.75				
38	take bowl 1 from desk 1	0.65				
39	examine bowl 1	0.85				
40	look	0.75				
41	move bowl 1 to desk 1	0.8				
42	examine desk 1	0.75				
43	look	0.85				
44	look	0.85				
45	examine desk 1	0.85				
46	take bowl 1 from desk 1	0.85				
47	examine bowl 1	0.95				
48	move bowl 1 to desk 1	0.85				
49	examine desk 1	0.85				

Table 8: A full qualitative example comparing baseline and our method (ALFWorld).

- **Final Explanation:** *"I am more confident now because I have explicitly targeted official sources for definitions to resolve ambiguity and added granular queries for regional/industry breakdowns to improve comprehensiveness."*

Conclusion: This case demonstrates that **UAR** is not a blind retry mechanism. By semantically parsing its own uncertainty (e.g., "missing subdivisions"), the agent was able to structurally upgrade its plan (adding specific queries) and operationally refine

its execution (switching tools), resulting in a more robust research trajectory.

A.7.3 Detailed Qualitative Case Study of Deep Research (English Query)

This example provides a full transcript of the reasoning process for a real-world example from the Deep Research Bench. The case demonstrates the **Dual-Process (AUQ)** mechanism in action: detecting uncertainty in the initial decomposition (System 1) and triggering the iterative reasoning process (System 2) to produce a superior research plan.

User Query: Japan Elderly Population Market Analysis
Scenario: The agent is tasked with a complex market analysis topic: *"From 2020 to 2050, how many elderly people will there be in Japan? What is their consumption potential across various aspects such as clothing, food, housing, and transportation? ... produce a market size analysis report."*

Phase 1: Initial Decomposition (System 1)

Confidence Score: 0.88
Threshold: 0.95
Status: **Reflection Triggered** (0.88 < 0.95)
The initial plan generated by the model was structurally sound but lacked the specificity required for professional market sizing.

Phase 2: The Reflection Process (System 2)

The system triggered a **UAR** strategy with 3 parallel branches. The agent iteratively refined the plan. Below is the trajectory of the successful branch.

Phase 3: Final Decomposition (Refined)

Final Confidence: 0.96
Outcome: The final plan significantly improved granularity (9 subtopics vs 4), source authority (IPSS/UN), and logic (separated spending vs. population).

Summary of Trajectory Refinement

In this example, the "Initial Decomposition"represents the standard output of the EDR planner. The **AUQ** module intercepted this output due to the low confidence score (0.88), preventing the EDR system from executing a suboptimal search plan. The "Refined Decomposition" was then injected back into the EDR pipeline, guiding the subsequent multi-step browsing agents. To quantify the impact of our Dual-Process mechanism on this specific trajectory, we analyze the step-by-step evolution of the decomposition plan. Table 9 summarizes the optimization process across different System 2 branches.

Evolution of the Decomposition Plan. The System 2 reflection process transformed the initial generic plan into a rigorously structured research strategy. Key qualitative improvements include:

- **Source Authority & Tool Selection:** The initial plan relied on generic queries. The final plan explicitly targets authoritative bodies (IPSS, UN projections) and strategically switches to `academic_search` for demographic data while retaining `general_search` for consumer trends.
- **Analytical Structure:** The system recognized a "calculation gap"in the original plan. The final output structurally decouples variables, requesting separate data for *per-capita spending* and *population projections*, to ensure the downstream Analyst agent can perform accurate market sizing.
- **Temporal & Granularity Expansion:**
  - *Temporal:* Added specific time-horizon queries (2020-2030, 2030-2040, 2040-2050) to capture non-linear growth trends over the 30-year period.
  - *Categorical:* Expanded the taxonomy from 4 to 9 subtopics, adding granular queries for specific consumption categories.
- **Methodological Grounding:** Crucially, the final plan includes a meta-query for "market sizing calculation frameworks," ensuring the research is grounded in established economic methodologies.

Key Insights. This trajectory highlights the **autonomous self-correction** capabilities of the **AUQ** framework:

1. **Iterative Refinement:** The improvement was not instantaneous but cumulative. Each reflection step (Iter 1 → Iter 2) built upon the identified gaps, progressively closing the knowledge/reasoning gap.
2. **Parallel Exploration:** By maintaining three active branches, the system avoided getting stuck in local optima (e.g., Branch 2 plateaued

```
{
  ``topic_complexity``: ``complex``,
  ``complex_topic``: {
    ``main_query``: ``Japan elderly population 2020-2050 market size consumption potential``,
    ``subtopics``: [
      {
        ``query``: ``Japan elderly population projections 2020-2050 statistics``,
        ``aspect``: ``Population projections``,
        ``rationale``: ``Foundation for understanding the demographic base``,
        ``suggested_tool``: ``general_search``
      },
      {
        ``query``: ``Japan elderly consumption patterns clothing food housing transportation``,
        ``aspect``: ``Consumption patterns by category``,
        ``rationale``: ``Core requirement to analyze consumption potential across specified categories``,
        ``suggested_tool``: ``general_search``
      },
      {
        ``query``: ``Japan elderly consumer behavior spending habits 2020-2025``,
        ``aspect``: ``Consumer behavior and willingness``,
        ``rationale``: ``Understanding current behavior to project future changes``,
        ``suggested_tool``: ``general_search``
      },
      {
        ``query``: ``Japan silver economy market size elderly demographic analysis``,
        ``aspect``: ``Market size analysis``,
        ``rationale``: ``Directly addresses the market size report requirement``,
        ``suggested_tool``: ``general_search``
      }
    ],
    ``analysis``: ``This is a complex multi-dimensional research topic requiring demographic projections,
     consumption pattern analysis, and market sizing across multiple categories (clothing, food, housing,
     transportation). The topic spans a 30-year period and requires both quantitative projections and
     qualitative analysis of behavioral changes.``
  }
}
```

Listing 1: Initial Decomposition Output

Agent Uncertainty Explanation

Uncertainty Explanation for Low Confidence (0.88):
"I'm confident about the overall structure of the decomposition, as it covers the key dimensions: population projections, consumption patterns by category, consumer behavior, and market sizing. The queries are well-structured and should capture relevant information. However, I have some uncertainties:

1. The specific data sources for Japan's official population projections (IPSS, UN) may require more targeted queries.
2. Consumption potential calculations need both per-capita spending data and population numbers - the queries may need to be more specific about combining these elements.
3. The time period spans 30 years (2020-2050), and consumption habits may change significantly - queries should account for temporal variations.
4. Market size analysis typically requires quantitative modeling - the current queries may not capture the analytical framework needed.
5. Category-specific queries (clothing, food, housing, transportation) could be more granular to capture sub-categories and trends

"

Figure 10: Example low-confidence explanation produced by System 1 before UAR reflection.

at 0.93), allowing it to discover the superior reasoning path in Branch 3.

3. **Transparent Decision Making:** Unlike black-box optimizations, every improvement is logged with a specific *rationale* (e.g., "Address data source uncertainty"), providing full interpretability of the agent's metacognitive process.

A.8 LLM Usage

We used LLM assistance for grammar, clarity, and style polishing of the manuscript. We also used an AI-assisted image generation/editing tool to create

```
{
  ``complex_topic``: {
    ``main_query``: ``Japan elderly population 2020-2050 market size consumption potential analysis``,
    ``subtopics``: [
      {
        ``query``: ``Japan IPSS UN population projections elderly 65+ 2020-2050 official statistics``,
        ``aspect``: ``Official population projections``,
        ``rationale``: ``Authoritative demographic data from IPSS and UN for accurate base calculations``,
        ``suggested_tool``: ``academic_search``
      },
      {
        ``query``: ``Japan elderly per capita consumption spending 2020-2025 by category clothing food housing transportation``,
        ``aspect``: ``Per-capita consumption baseline``,
        ``rationale``: ``Establish baseline spending patterns to project forward``,
        ``suggested_tool``: ``general_search``
      },
      {
        ``query``: ``Japan elderly population projections 2020-2030 2030-2040 2040-2050 age cohorts``,
        ``aspect``: ``Temporal population breakdown``,
        ``rationale``: ``Capture population changes across different time periods for accurate market sizing``,
        ``suggested_tool``: ``general_search``
      },
      {
        ``query``: ``Japan elderly consumption behavior changes trends 2020-2025 future projections``,
        ``aspect``: ``Behavioral change trends``,
        ``rationale``: ``Understand how consumption habits may evolve over the 30-year period``,
        ``suggested_tool``: ``general_search``
      },
      {
        ``query``: ``Japan silver economy market size calculation methodology elderly demographic``,
        ``aspect``: ``Market sizing methodology``,
        ``rationale``: ``Identify analytical frameworks for calculating market size from demographic and spending data``,
        ``suggested_tool``: ``academic_search``
      },
      {
        ``query``: ``Japan elderly clothing spending trends functional apparel adaptive fashion``,
        ``aspect``: ``Clothing category specifics``,
        ``rationale``: ``Detailed sub-category analysis for clothing consumption potential``,
        ``suggested_tool``: ``general_search``
      },
      {
        ``query``: ``Japan elderly food consumption patterns health foods delivery services 2020-2025``,
        ``aspect``: ``Food category specifics``,
        ``rationale``: ``Detailed analysis of food consumption including health trends and delivery services``,
        ``suggested_tool``: ``general_search``
      },
      {
        ``query``: ``Japan elderly housing costs home modifications aging in place services``,
        ``aspect``: ``Housing category specifics``,
        ``rationale``: ``Comprehensive housing analysis including modifications and services``,
        ``suggested_tool``: ``general_search``
      },
      {
        ``query``: ``Japan elderly transportation mobility services community transport accessible travel``,
        ``aspect``: ``Transportation category specifics``,
        ``rationale``: ``Detailed transportation analysis including mobility services and accessibility``,
        ``suggested_tool``: ``general_search``
      }
    ],
    ``analysis``: ``This enhanced decomposition addresses the key uncertainties: (1) Uses authoritative sources (IPSS, UN) for population data, (2) Separates per-capita spending from population projections for accurate calculations, (3) Includes temporal breakdowns for the 30-year period, (4) Adds market sizing methodology queries, and (5) Provides granular category-specific queries. The decomposition now supports a comprehensive market size analysis report.``
  }
}
```

Listing 2: Final Refined Decomposition (Branch 3, Iteration 2)

Agent Uncertainty Explanation

Uncertainty Explanation for High Confidence (0.96):
"After reflection, I'm more confident in this decomposition. Key improvements include:

1. **Addressed data source uncertainty**: Added specific queries for IPSS and UN official projections, using academic_search for authoritative data
2. **Separated calculation components**: Created distinct queries for per-capita spending and population projections, enabling proper market size calculations
3. **Added temporal dimension**: Included queries for different time periods (2020-2030, 2030-2040, 2040-2050) to capture changes over the 30-year span
4. **Included methodology**:Added a query for market sizing calculation frameworks
5. **Enhanced category granularity**: Expanded from 4 to 9 subtopics with more specific queries for each consumption category

However, I'm still somewhat uncertain about:

1. Whether the queries will capture sufficient data for quantitative modeling
2. The balance between breadth and depth in category-specific queries

My confidence has improved from 0.88 to 0.96, reflecting these enhancements. "

Figure 11: Example high-confidence explanation after UAR reflection.

Attempt	Reflection Stage	Conf. ($\hat{c}$)	Imp.	Status
1	Initial Decomposition	0.88	-	*Triggered System 2*
2	Branch 1 / Iteration 1	0.92	+0.04	Below Threshold
3	Branch 1 / Iteration 2	0.94	+0.02	Below Threshold
4	Branch 2 / Iteration 1	0.90	+0.02	Below Threshold
5	Branch 2 / Iteration 2	0.93	+0.03	Below Threshold
6	Branch 3 / Iteration 1	0.94	+0.06	Below Threshold
7	**Branch 3 / Iteration 2**	**0.96**	**+0.08**	**Selected ($\geq \tau$)**

Table 9: **Trajectory of Confidence Optimization.** The system explored three parallel reasoning branches using Best-of-N sampling. Branch 3 ultimately yielded the highest confidence solution ($\hat{c} = 0.96$), surpassing the acceptance threshold.

and refine the conceptual illustration in Figure 1, which is used only as a schematic visualization of the proposed framework.