Provably Reliable Tool-Using LLM Agents: Formal Guarantees on Error Accumulation in the Model Context Protocol (MCP)

Flint Xiaofeng Fan[1,3], **Cheston Tan**[1], **Roger Wattenhofer**[3], **Yew-Soon Ong**[1,2]
[1]CFAR, IHPC, Agency for Science, Technology and Research, Singapore
[2]College of Computing and Data Science, Nanyang Technological University, Singapore
[3]Department of Information Technology and Electrical Engineering, ETH Zurich, Switzerland
fxf@u.nus.edu, cheston-tan@a-star.edu.sg, wattenhofer@ethz.ch, asysong@ntu.edu.sg

Abstract

As AI agents powered by large language models (LLMs) increasingly use external tools for high-stakes decisions, a critical reliability question arises: how do errors propagate across sequential tool calls? We introduce the first theoretical framework for analyzing error accumulation in Model Context Protocol (MCP) agents, proving that cumulative distortion grows linearly in expectation with high-probability deviations bounded by $O(\sqrt{T})$. This concentration property ensures predictable system behavior and rules out exponential failure modes. We develop a hybrid distortion metric combining discrete fact matching with continuous semantic similarity, then establish martingale concentration bounds on error propagation through sequential tool interactions. Experiments across Qwen2-7B, Llama-3-8B, and Mistral-7B validate our theoretical predictions, showing empirical distortion tracks the linear trend with deviations consistently within $O(\sqrt{T})$ envelopes. Key findings include: semantic weighting reduces distortion by 80%, and periodic re-grounding approximately every 8 steps suffices for error control. We translate these concentration guarantees into actionable deployment principles for trustworthy agent systems. The codebase is available at https://github.com/flint-xf-fan/MCP.

1 Introduction

Large language models (LLMs) have achieved remarkable capabilities across natural language processing tasks, yet they remain fundamentally constrained by the static snapshots of knowledge encoded in their training corpora (Villalobos et al. 2022; Naveed et al. 2023; Li et al. 2024). Once deployed, an LLM cannot ingest new facts or correct errors without costly retraining, leading to stale or even blatantly incorrect outputs in rapidly changing domains such as real-time news, financial markets, or clinical guidelines (Bommasani et al. 2021; Liang et al. 2022; Fan et al. 2025a). As Silver and Sutton (Silver and Sutton 2025) aptly observe, "we stand at the threshold of a new era in artificial intelligence" where systems must transcend fixed datasets and learn through dynamic interaction.

To address this fundamental limitation, the Model Context Protocol (MCP) has emerged as the de facto standard for connecting LLMs to external tools and data sources (Anthropic 2024). MCP transforms what would traditionally be an $M \times N$ integration problem (with M different AI applications needing custom connections to N different tools) into a more manageable $M + N$ approach through a unified JSON-RPC framework, enabling seamless composition of models and services, as illustrated in Figure 1. Major platforms including Anthropic and Microsoft have implemented MCP-compatible interfaces (Anthropic 2024; Microsoft Azure AI Team 2025), facilitating the development of systems that can retrieve up-to-date information, perform precise calculations, and interact with real-world services. Through MCP, LLMs can continuously integrate fresh external evidence to stay epistemically grounded and operationally up-to-date long after their initial training, instantiating Sutton and Silver's concept of experiential learning (Silver and Sutton 2025).

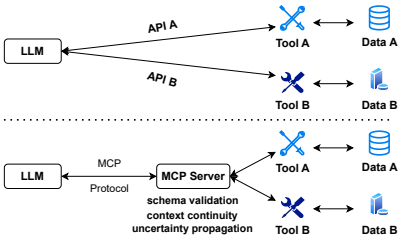

Figure 1: MCP standardizes LLM-tool integration through a unified JSON-RPC interface (bottom), replacing custom per-tool connections (top) with centralized schema validation, context management, and uncertainty propagation.

However, this enhanced agility puts *information fidelity* at risk: each call to an external tool introduces opportunities for factual error, semantic drift, and compounding inaccuracies (Qin et al. 2025). In safety-critical applications, ranging from clinical decision support to financial analysis, even a minor mistake in an early query can cascade into catastrophic downstream consequences due to the sequential nature of decision-making processes (Fan et al. 2021, 2025b). For instance, in 2012, Knight Capital lost $440 million due to a software error in their trading algorithm, highlighting how small mistakes can lead to significant finan-

Copyright © 2026, Trustworthy Agentic AI Workshop@ Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

cial consequences (Dolfing 2019). Similarly, studies have shown that AI systems can provide inaccurate medical drug dosages, potentially leading to dangerous treatment recommendations (Ramasubramanian et al. 2024).

As tool usage becomes increasingly central to deployed AI systems, ensuring information fidelity across sequential interactions emerges as a critical challenge. Despite MCP's widespread adoption in production systems, we lack a formal framework for understanding how errors accumulate across adaptive queries, or what guarantees can be provided regarding the integrity of multi-step reasoning chains. While empirical work has demonstrated that function-calling improves factuality (Schick et al. 2023; Yao et al. 2023), rigorous theoretical bounds on error propagation remain elusive.

In this paper, we bridge this gap by providing, to our knowledge, the first work to derive *general high-probability deviation bounds* for cumulative hybrid semantic distortion in MCP-style, tool-augmented LLM pipelines under explicit dependence assumptions. A martingale represents a mathematical "fair game" where future expectations equal the current state—intuitively preventing errors from spiraling uncontrollably across sequential interactions. By adapting concentration techniques from adaptive data analysis (Howard et al. 2021; Dwork et al. 2015) and applying them to sequential tool interactions, we establish formal connections between everyday engineering practice and statistical learning theory. Our analysis models MCP exchanges as a bounded-difference martingale with finite dependency horizons, enabling us to derive high-probability bounds on cumulative semantic distortion. Our main contributions are:

- **Information Fidelity Framework (Section 3).** We develop a comprehensive theoretical framework for analyzing MCP interactions, including a novel semantic distortion metric that combines weighted discrete fact matching with continuous embedding-based similarity, capturing both strict factual correctness and nuanced semantic drift.
- **Concentration Bounds (Section 4).** Within our framework, we derive high-probability bounds showing cumulative semantic distortion grows linearly in expectation with deviations bounded by $O(\sqrt{T\ln(1/\eta)})$ across sequential MCP interactions.
- **Empirical Validation (Section 5).** Through experiments with Qwen2-7B-Instruct, Llama-3-8B-Instruct, and Mistral-7B-Instruct-v0.3, we demonstrate that empirical distortion exhibits the predicted linear growth with $O(\sqrt{T})$ concentration bounds across varying dependency strengths.
- **Practical Insights (Section 6).** We derive actionable design principles from theoretical results, including optimal re-grounding intervals and distortion monitoring strategies, enabling safe deployment of MCP agents in safety-critical domains where unchecked error accumulation risks catastrophic failures.

2 Preliminaries

Tool-Augmented Language Models A rich line of work teaches LLMs to invoke external tools, e.g. Toolformer (Schick et al. 2023) and ToolLLM (Qin et al. 2023). ReAct (Yao et al. 2023) interleaves reasoning and tool calls. While these methods boost performance, none provide formal guarantees on how errors accumulate across multiple invocations.

Retrieval-Augmented Generation RAG systems (Lewis et al. 2020) empower LLMs with external corpus lookups. Recent studies analyzing retrieval errors include RARR (Gao et al. 2022) and Faithful reasoning (Creswell and Shanahan 2022). Despite advances in index refinement (Ram et al. 2023) and end-to-end training (Borgeaud et al. 2022; Guu et al. 2020), the theoretical reliability of repeated retrieval–generation loops remains open.

Model Context Protocol General RPC/API standards, such as JSON-RPC (JSON-RPC Working Group 2013) and OpenAPI (OpenAPI Initiative 2021), and the Language Server Protocol (Microsoft 2022) were not designed for LLM workflows. MCP (Anthropic 2024) fills this gap, prescribing schema validation, context continuity, and uncertainty handling. Its de facto emergence is evidenced by platforms like Anthropic (Anthropic 2024) and Microsoft Azure (Microsoft Azure AI Team 2025). Figure 1 illustrates the typical flow: an LLM issues a JSON-RPC request validated by an MCP proxy, which routes it to appropriate tools and returns structured responses. A primer on MCP is provided in Appendix A.

Martingale-Based Analyses Concentration bounds such as the Azuma–Hoeffding (Azuma 1967; Hoeffding 1963) and Freedman's refinement (Freedman 1975) underpin online learning and adaptive control (Dwork et al. 2015). A martingale is a stochastic process $\{X_t\}_{t=0}^{T}$ where $\mathbb{E}[X_{t+1}|X_0, X_1, \ldots, X_t] = X_t$, formalizing the concept of a "fair game." Modern self-normalized variants (Howard et al. 2021) tighten these results, but to our knowledge no work has used them to quantify distortion in LLM–tool pipelines. The Azuma–Hoeffding inequality states that for a martingale with bounded differences $|X_{t+1} - X_t| \leq c_t$: $\Pr\left(X_T - X_0 \geq \delta\right) \leq \exp\left(-\frac{\delta^2}{2\sum_{t=0}^{T-1} c_t^2}\right)$.

To our knowledge, no prior work unifies these strands into a principled, end-to-end theory of error accumulation in MCP-based LLM systems. In the next section, we develop our Information Fidelity Framework, which formalizes MCP interactions as an adaptive stochastic process and establishes the theoretical foundation for analyzing semantic distortion.

3 Information Fidelity Framework

To address the gap in prior work, we propose an Information Fidelity Framework that models MCP interactions as an adaptive stochastic process, quantifying error accumulation through a novel distortion metric and martingale analysis. Our framework formalizes how information propagates and potentially degrades through sequential MCP interactions, providing theoretical foundations for bounding semantic distortion across tool calls. We begin by establishing basic definitions and our semantic distortion metric, fol-

lowed by key modeling assumptions, and finally derive the properties that enable our concentration results.

Definition 1 (Information Filtration)**.** *The sequence of pairs of query–response $\{(Q_t, R_t)\}$ is modeled via the natural filtration $\mathcal{F}_t = \sigma(Q_1, R_1, \ldots, Q_t, R_t)$, representing all information available after t interactions.*

Definition 2 (Influence Function)**.** *The parameter $\beta \in (0, 1)$ controls exponential decay of past influences through the influence function $\phi(i, j) = \beta^{j-i}$ for $1 \leq i < j \leq T$, with β controlling the decay rate.*

Definition 3 (Ideal Fact Set)**.** *For each prompt–response pair (Q_t, R_t), let $\mathcal{I}_t$ be the set of ground-truth facts (e.g. triples or attribute–value pairs) that R_t is expected to include.*

3.1 Quantifying Semantic Distortion

Measuring information fidelity requires balancing strict factual correctness against broader semantic similarity. We introduce a hybrid metric that captures both dimensions:

$$\Delta_t = d_{\text{sem}}\big(R_t, \mathcal{I}_t\big) = (1-\lambda)\, d^w_{\text{set}}(R_t, \mathcal{I}_t) + \lambda\, d_{\text{emb}}(R_t, \mathcal{I}_t), \tag{1}$$

where:

$$d^w_{\text{set}}(R_t, \mathcal{I}_t) = 1 - \frac{\sum_{f \in \hat{\mathcal{I}}_t \cap \mathcal{I}_t} w(f)}{\sum_{f \in \mathcal{I}_t} w(f)}, \quad \hat{\mathcal{I}}_t = \text{extract}(R_t),$$

$$d_{\text{emb}}(R_t, \mathcal{I}_t) = \frac{1 - \cos\Big(\text{embed}(R_t), \text{embed}(\mathcal{I}_t)\Big)}{2}.$$

Here, extract(R_t) is a function that extracts the set of facts from response R_t, $w(f)$ assigns importance weights to individual facts, and $\text{embed}(\cdot)$ maps text to a semantic vector space. We define $\text{embed}(\mathcal{I}_t) = \frac{1}{|\mathcal{I}_t|} \sum_{f \in \mathcal{I}_t} \text{embed}(f)$, or for weighted facts, the weighted average of their embeddings.

The first component, d^w_{set}, measures factual completeness through weighted fact matching, penalizing both missing and incorrect facts. The second component, d_{emb}, captures semantic similarity in embedding space, addressing nuanced meaning beyond exact fact matching. The parameter $\lambda \in [0, 1]$ allows tuning the trade-off between factual precision and semantic similarity. The choice of λ depends on the application; for example, $\lambda = 0.5$ balances factual and semantic priorities equally, while lower values prioritize factual accuracy and higher values emphasize semantic coherence.

To illustrate how this metric operates in practice, consider a simple example:

- Query: "What is the capital of France?"
- Response A: "Paris is the capital of France."
- Response B: "The capital of France is Paris."

Both responses contain the same fact (Paris is France's capital), so $d^w_{\text{set}} = 0$ for both. However, d_{emb} would differ slightly due to wording differences. The hybrid metric d_{sem} captures this nuance while still recognizing factual equivalence. This example demonstrates how our metric balances factual correctness with semantic nuance, providing a more comprehensive measure of information fidelity than either component alone.

Definition 4 (Cumulative Distortion)**.** *Let T be the total number of tool calls in an MCP interaction. The cumulative distortion is defined as:*

$$D(T) \;=\; \sum_{t=1}^{T} \Delta_t,$$

where Δ_t is the step-wise semantic distortion at step t as defined in Equation (1)*.*

3.2 Modeling Assumptions

Our Information Fidelity Framework requires standard technical assumptions on normalization and continuity (see Appendix B.1), ensuring our distortion metric is well-defined and bounded in [0,1]. We now state the substantive modeling assumptions.

Definition 5 (Effective Branching Factor)**.** *The effective branching factor $B \geq 1$ denotes the maximum number of distinct future queries that can be directly influenced by a single response within the interaction graph. Sequential chains satisfy $B = 1$, while tree-structured interactions have B equal to the maximum number of children per node.*

Assumption 1 (Bounded Branching)**.** *Let B be the maximum effective branching factor. Then $\beta B < 1$.*

Here, the branching factor B represents the maximum number of future queries potentially influenced by a single response. This condition is *sufficient* to establish our theoretical guarantees, ensuring influence decays across steps. Notably, this is only a *sufficient* but not *necessary* condition for establishing the concentration bound. While stronger dependencies ($\beta B \gg 1$) may require alternative analytical approaches, which we explore in Appendix E, the transition appears to be gradual rather than exhibiting a sharp phase transition at exactly $\beta B = 1$.

Assumption 2 (Response Stability)**.** *There exists $\alpha \leq 1$ such that small input perturbations induce only continuous changes in the LLM's response distribution, governed by an influence function $\phi(i, j)$.*

Modern LLMs exhibit a degree of robustness to minor input variations, and this assumption captures that property. Without response stability, a single character change could dramatically alter all subsequent reasoning, making error bounds impossible.

Assumption 3 (Temporal Decay Structure)**.** *For any $t < T$ and any two execution paths that coincide through time t but differ at step $t+1$, there exist constants $\alpha \in [0, 1]$, $\beta \in [0, 1)$, and branching factor $B \geq 1$ such that distortions at future steps can be coupled to satisfy*

$$|\Delta_{t+k} - \Delta'_{t+k}| \leq \alpha(\beta B)^{k-1}, \quad k \geq 1,$$

almost surely under appropriate coupling. Additionally, $|\Delta_{t+1} - \Delta'_{t+1}| \leq 1$ by Lemma 4. If periodic re-grounding occurs every m steps, then $\Delta_{t+k} = \Delta'_{t+k}$ for all $k > m$.

Remark 1 (Modeling framework justification)**.** *This temporal decay model captures three key properties of LLM-tool systems: (i) exponential decay reflecting finite effective*

Response Pair	Fact Match	Embedding Distance
"Apple's revenue was $94.8B" vs. "Apple reported $94.8B in revenue"	1.0	0.17
"The GDP grew by 2.5% in Q2" vs. "Second quarter GDP expansion was 2.5%"	1.0	0.21
"The meeting is on Monday at 3PM" vs. "Monday at 3PM is when we'll meet"	1.0	0.35

Table 1: Embedding distances for responses with identical facts but different wording

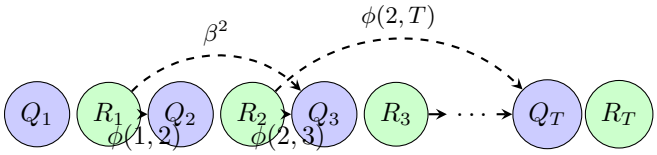

Figure 2: Dependency graph for MCP interactions. Solid arrows indicate direct influence $\phi(i, i+1) = \beta$, and the dashed arrow shows long-range decay $\phi(i, j) = \beta^{j-i}$. This exponential decay structure enables us to prove that distortion deviations remain sublinear, even with adaptive queries.

memory (Press, Smith, and Lewis 2022), (ii) bounded propagation preventing error amplification, and (iii) reset through periodic re-grounding—standard practice in production deployments. The framework connects to established coupling techniques from Markov chain mixing (Levin and Peres 2017) and adaptive data analysis (Dwork et al. 2015), providing sufficient conditions for concentration bounds. Section 5 validates the decay rate β through autocorrelation diagnostics; Appendix E.2 extends to alternative dependency structures.

3.3 Modeling as an Adaptive Stochastic Process

The inherently sequential nature of tool-augmented reasoning requires careful modeling of dependencies between successive interactions. We formalize this as a sequence of query-response pairs $\{(Q_t, R_t)\}_{t=1}^{T}$, where each new query can depend on all previous interactions:

$$Q_t \in \sigma\big(Q_1, R_1, \ldots, Q_{t-1}, R_{t-1}\big).$$

This adaptive formulation is essential for capturing realistic LLM behavior, where each tool call builds on previous responses.

Figure 2 illustrates this dependency structure. A key insight is that influence follows an exponential decay pattern described by Definition 2. This pattern aligns with how information naturally attenuates in sequential reasoning: recent statements have stronger impact than distant ones, and verification steps effectively reset error chains by grounding the model in authoritative data.

3.4 Properties of Semantic Distortion

Our semantic distortion metric exhibits crucial properties that enable concentration analysis. First, it distinguishes lexical variation from semantic equivalence:

Lemma 1 (Semantic Sensitivity)**.** *If two responses contain identical facts but differ lexically, then*

$$d^{w}_{\text{set}}(R_t, \mathcal{I}_t) = d^{w}_{\text{set}}(R'_t, \mathcal{I}_t), \quad d_{\text{emb}}(R_t, \mathcal{I}_t) \neq d_{\text{emb}}(R'_t, \mathcal{I}_t),$$

ensuring d_{sem} captures semantic nuance.

Proof. If R_t and R'_t have identical fact coverage, then

$$\text{extract}(R_t) = \text{extract}(R'_t) = \hat{\mathcal{I}}_t.$$

Therefore, $d^{w}_{set}(R_t, \mathcal{I}_t) = d^{w}_{set}(R'_t, \mathcal{I}_t)$.

However, different wording produces different embeddings: embed$(R_t) \neq$ embed(R'_t), causing $d_{emb}(R_t, \mathcal{I}_t) \neq d_{emb}(R'_t, \mathcal{I}_t)$. For $\lambda > 0$, this ensures $d_{sem}(R_t, \mathcal{I}_t) \neq d_{sem}(R'_t, \mathcal{I}_t)$. Table 1 shows empirical evidence: embedding distances vary 0.15–0.35 in cosine space even when factual content is identical. □

Second, the metric varies continuously with response perturbations, ensuring small edits induce proportionally small distortion changes (formalized in Lemma 3, Appendix B.2).

3.5 Martingale Construction for Concentration

To bound the cumulative distortion $D(T) = \sum_{t=1}^{T} \Delta_t$, we construct a martingale sequence whose increments we can control with classical concentration inequalities.

Definition 6 (Distortion Martingale)**.** *Define*

$$Z_t \;=\; \mathbb{E}\big[D(T) \;\big|\; \mathcal{F}_t\big], \quad t = 0, 1, \ldots, T,$$

where $Z_0 = \mathbb{E}[D(T)]$ and $Z_T = D(T)$.

This forms a martingale because each Z_t uses all available information up to time t to predict the final distortion $D(T)$. The martingale property ensures that future distortions are conditionally unbiased relative to our current estimate—mathematically, $\mathbb{E}[Z_{t+1}|\mathcal{F}_t] = Z_t$. This property is crucial as it allows us to apply concentration inequalities to bound how far the actual distortion can deviate from its expectation.

Intuitively, Z_t represents our best prediction of the final cumulative distortion given all information available after t interactions; hence observing the next response cannot systematically bias that prediction. As t increases, Z_t incorporates more information and converges to the actual distortion $D(T)$ when all interactions are observed.

In the next section, we show that the increments $Z_{t+1} - Z_t$ satisfy a bounded-difference property, which we combine with Azuma's inequality to derive high-probability concentration bounds on how much the actual distortion $D(T)$ can deviate from its linearly growing expected value, ensuring predictable system behavior despite adaptive querying.

4 Concentration for Adaptive Querying

Having established our information fidelity framework, we now address the central theoretical question: how does distortion accumulate across sequential tool calls? While Definition 4 specifies that $D(T) = \sum_{t=1}^{T} \Delta_t$, each Δ_t can depend on all previous interactions through the adaptive process formalized in Section 3.3. This adaptive querying could, in principle, allow errors to compound arbitrarily. Our main result shows otherwise: under the temporal decay structure of Assumption 3, cumulative distortion remains tightly concentrated around its expectation, with deviations bounded by $O(\sqrt{T})$ despite full adaptivity.

Our main insight is that despite adaptive querying, where each new prompt can depend arbitrarily on all previous interactions, the cumulative distortion $D(T) = \sum_{t=1}^{T} \Delta_t$ remains tightly concentrated around its expectation. This concentration emerges from the exponential decay of influence across steps, creating a form of "effective independence" that enables powerful probabilistic guarantees.

4.1 The Bounded-Difference Property

The following lemma establishes how much a single new response can affect our expectation of the total distortion under the temporal decay structure. The constant $C^* = \frac{\alpha}{1-\beta B}$ quantifies the maximum possible cumulative influence that a single response can have on all future queries, with α capturing response stability (Assumption 2) and B accounting for the branching factor (Assumption 1). This uniform bound allows us to derive clean concentration results regardless of a query's position in the sequence.

Lemma 2 (Bounded Doob increments)**.** *Let $Z_t = \mathbb{E}[D(T) \mid \mathcal{F}_t]$ be the distortion Doob martingale (Definition 6). Under Assumptions 1 and 3, we have, almost surely,*

$$|Z_{t+1} - Z_t| \leq c_\star \quad \textit{for all } t < T,$$

with

$$c_\star = 1 + C^* = 1 + \frac{\alpha}{1-\beta B} \quad \textit{when } \beta B < 1.$$

If periodic re-grounding enforces a finite horizon m, then

$$c_\star = 1 + \alpha \cdot \frac{1-(\beta B)^m}{1-\beta B}$$

Proof Sketch. The martingale increment decomposes as $Z_{t+1} - Z_t = (\Delta_{t+1} - \mathbb{E}[\Delta_{t+1}|\mathcal{F}_t]) + \sum_{j=t+2}^{T} (\mathbb{E}[\Delta_j|\mathcal{F}_{t+1}] - \mathbb{E}[\Delta_j|\mathcal{F}_t])$. The first term is bounded by 1 via Lemma 4. For the tail sum, the pathwise coupling from Assumption 3 ensures $|\Delta_j - \Delta'_j| \leq \alpha(\beta B)^{j-t-2}$ almost surely for coupled executions, implying each conditional expectation difference is bounded by the same factor. Summing the geometric series $\sum_{j=t+2}^{T} \alpha(\beta B)^{j-t-2} \leq \alpha/(1-\beta B) = C^*$ completes the bound. Full details in Appendix B.3. □

This bounded-difference property is the linchpin enabling concentration analysis: it ensures that even though each new response can influence all future queries through the adaptive chain, that influence diminishes rapidly enough to maintain exponential concentration around the expected cumulative distortion.

4.2 High-Probability Concentration Results

Theorem 1 (High-probability distortion bound)**.** *Let $C^* = \frac{\alpha}{1-\beta B}$ and $\gamma^* = 2C^* + (C^*)^2$. For any $\eta \in (0,1)$,*

$$\Pr\Big[D(T) - \mathbb{E}D(T) \;\geq\; \sqrt{2\,T(1+\gamma^*)\,\ln\tfrac{1}{\eta}}\,\Big] \;\leq\; \eta.$$

Proof Sketch. By Lemma 2, the distortion martingale has bounded increments $|Z_{t+1} - Z_t| \leq c_\star = 1 + C^*$. Summing $c_\star^2 = (1+C^*)^2 = 1 + 2C^* + (C^*)^2$ over T steps yields $T(1+\gamma^*)$. Azuma's inequality then gives the stated high-probability bound. The complete proof appears in Appendix B.4. □

The correction factor γ^* quantifies how dependencies in our adaptive process inflate the variance term compared to the independent case. With $C^* = \frac{\alpha}{1-\beta B}$ representing the maximum cumulative influence of any response on all future queries, $\gamma^* = 2C^* + (C^*)^2$ captures both the linear interaction between bounded differences and the quadratic effect of compounded influences. Together, they show that even with adaptive queries, the cumulative distortion exhibits linear expected growth with sublinearly bounded deviations. This concentration property rules out both exponential error blowup and chaotic variance growth. The practical implications of this result are captured in the following corollaries:

Corollary 1 (Sub-linear deviation)**.** *With probability at least $1-\eta$, $D(T) = \mathbb{E}D(T) + O\big(\sqrt{T\ln(1/\eta)}\big)$.*

This result directly addresses fears about chaotic error compounding in LLM reasoning chains. Rather than exhibiting exponential divergence, cumulative distortion remains tightly concentrated around a predictable linear trajectory. The $O(\sqrt{T})$ term guarantees that the "safety margin" required to contain the error grows sublinearly relative to the chain length. For $T = 10$, this limits the worst-case deviation to approximately $\sqrt{10} \approx 3.2$ times the single-step uncertainty.

Corollary 2 (Effective information horizon)**.** *The query-level influence function $\phi(i,j) = \beta^{j-i}$ (Definition 2) decays exponentially, with influence dropping below threshold ε after $\mathcal{H}_\varepsilon = \lceil \log(\varepsilon)/\log(\beta) \rceil$ steps. For β close to 1, this simplifies to approximately $\lceil -\ln(\varepsilon)/(1-\beta) \rceil$. The* cumulative *propagation of a single error across all future queries is bounded by $C^* = \alpha/(1-\beta B)$, which quantifies the maximum total influence (amplitude). Together, $\mathcal{H}$ (range) and C^* (magnitude) determine how far and how strongly errors can propagate.*

This corollary introduces the concept of an "effective information horizon": the maximum distance over which errors meaningfully propagate through the query chain. The horizon $\mathcal{H}$ is set by the decay rate β of $\phi(i,j)$ (range), while the amplitude of propagated errors is scaled by C^*, which incorporates both response stability (α) and branching (B) (magnitude). For typical values (e.g., $\beta = 0.7$), this horizon is approximately 8 steps, supporting periodic re-grounding in long reasoning chains.

Complete proofs of above results, along with refinements using tighter concentration inequalities, appear in Appendix B.4, B.5 and B.6, respectively.

5 Experiments

Having established theoretical guarantees for information fidelity, we validate three central predictions: (1) Does cumulative distortion exhibit linear expected growth with sublinear concentration as claimed by Corollary 1? (2) How do framework parameters (β, λ) affect empirical distortion? (3) Does the martingale analysis capture fundamental MCP dynamics or merely architectural artifacts? We address these through systematic experiments across Qwen2-7B-Instruct (Qwen) (Yang and et al. 2024) and Llama-3-8B-Instruct (Llama) (AI@Meta 2024), with Mistral-7B-Instruct-v0.3 (Mistral) (Jiang et al. 2023) in Appendix D.

MCP Implementation. All experiments use identical MCP tools: a knowledge retrieval tool operating over a deterministic cached corpus covering eight domains (history, science, technology, arts, sports, geography, literature, mathematics), and a financial data tool providing schema-validated access to market snapshots. Both enforce strict schema validation and maintain context continuity, faithfully implementing MCP specifications. To isolate error accumulation dynamics from tool variability, we employ deterministic, cached responses. This controlled setting establishes fundamental bounds that serve as a baseline for understanding systems with stochastic tools.

Query Generation and Distortion Measurement. Each chain begins with a seed question from domain-specific templates. Subsequent queries are generated adaptively with probability β, implementing the influence function $\phi(i, j) = \beta^{j-i}$ from Definition 2. Our experimental chains maintain linear structure with branching factor $B = 1$, satisfying Assumption 1's condition $\beta B < 1$. For each response R_t, we extract ideal fact set $\mathcal{I}_t$ from ground truth and compute hybrid semantic distortion via Eq. (1). Cumulative distortion $D(T) = \sum_{t=1}^{T} \Delta_t$ is tracked across each chain, with statistics aggregated over multiple independent trials.

Theoretical Bounds and Model Validation. We construct calibrated envelopes by: (1) estimating expected per-step distortion $\hat{r}$ from mean first-step distortion, (2) extrapolating linearly to $\hat{\mathbb{E}}[D(T)] = T \cdot \hat{r}$, and (3) adding theoretical deviation $\sqrt{2(1+\hat{\gamma})T\ln(1/\delta)}$ where $\hat{\gamma}$ is computed from empirical dependency strength (Appendix C.5). These bounds instantiate Theorem 1 with 95% confidence ($\delta = 0.05$). To validate Assumption 3, we compute empirical autocorrelations $\hat{\rho}(k) = \mathrm{Corr}(\Delta_t, \Delta_{t+k})$ and fit decay rate $\hat{\beta} = \arg\min_\beta \sum_k (\hat{\rho}(k) - \beta^k)^2$, yielding $\hat{\beta} \in [0.68, 0.71]$ across architectures. Full details appear in Appendix C.

Baseline Validation. Under standard conditions ($\beta = 0.7$, $\lambda = 0.5$, $T = 10$, 50 chains per model), Figure 3 reveals striking consistency between theory and practice. Qwen2-7B achieves $D(10) = 5.26 \pm 0.34$ versus calibrated envelope 9.15 (safety margin $1.74\times$); Llama-3-8B records $D(10) = 4.92 \pm 0.46$ with envelope 8.90 (margin $1.81\times$). The consistent per-step rate of ≈ 0.5 across all chains confirms linear expected growth, while safety margins $> 1.7\times$ validate our $O(\sqrt{T})$ concentration bounds from Corollary 1.

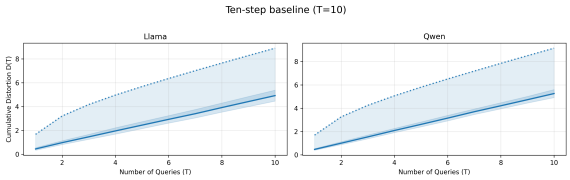

Figure 3: **Baseline distortion accumulation over 10 tool calls.** Cumulative distortion with $\beta = 0.7$, $\lambda = 0.5$ (50 chains/model). Solid: empirical mean $\pm 1\sigma$; dotted: high-probability envelopes (Theorem 1, 95% confidence). Distortion grows linearly at ≈ 0.5 per step with deviations tightly concentrated around the linear trend, validating $O(\sqrt{T})$ concentration bounds.

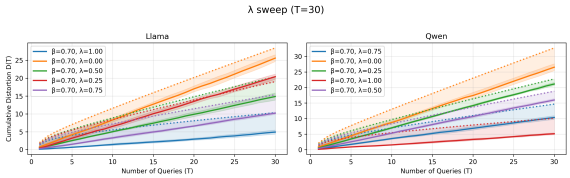

Figure 4: **Lambda sweep: semantic weighting effect.** Distortion for $\lambda \in \{0, 0.25, 0.5, 0.75, 1.0\}$ at $T = 30$, $\beta = 0.7$ (8 chains/config). Solid: empirical mean $\pm 1\sigma$; dotted: calibrated bounds.

Semantic Weighting Trade-off. Systematic sweeps with $\lambda \in \{0, 0.25, 0.5, 0.75, 1.0\}$ at $T = 30$, $\beta = 0.7$ (8 chains per configuration) show that increasing λ from 0 (pure fact matching) to 1 (pure semantic similarity) reduces final distortion by approximately 80% for both models: Qwen2-7B drops from $D(30) = 26.53$ to 5.11, Llama-3-8B from 25.62 to 4.90 (Figure 4, Table 2). This substantial effect reveals that exact fact matching accumulates errors more aggressively than semantic drift—every paraphrase contributes distortion at $\lambda = 0$ even when meaning is preserved. The plateau beyond $\lambda = 0.75$ suggests balanced regimes capture complementary error modes more effectively than extremes. Safety margins remain positive across all configurations except Llama-3-8B at $\lambda = 0.25$ (ratio 0.93), suggesting our first-step calibration occasionally underestimates uncertainty in extreme parameter regimes (Section 6, Appendix C.5).

Extended Validation. Experiments across $T = 60$ steps and extreme dependencies ($\beta \in \{0.5, 0.7, 0.9, 0.95, 0.98\}$) demonstrate predictable linear accumulation with $O(\sqrt{T})$ concentration bounds maintaining consistent safety margins ($1.10\times$–$1.55\times$). Even at $\beta = 0.98$ approaching the critical point, the system avoids exponential failure modes, showing that high dependencies inflate the variance bounds rather than the mean distortion rate. Robustness to corrupted tool outputs (noise levels 0.0–0.2) further validates the martingale property, showing stable margins $> 1.25\times$(Figures 5, 6, 7).

	Qwen2-7B			Llama-3-8B		
λ	$D(T)$	Envelope	Env./$D(T)$	$D(T)$	Envelope	Env./$D(T)$
0.00	26.53	32.74	1.23	25.62	28.47	1.11
0.25	21.12	22.75	1.08	20.43	19.04	0.93
0.50	15.97	18.79	1.18	14.86	20.40	1.37
0.75	10.34	14.65	1.42	10.20	15.38	1.51
1.00	5.11	10.10	1.98	4.90	10.33	2.11

Table 2: Final distortion for λ sweep at $T = 30$, $\beta = 0.7$. Ratios > 1 indicate safety margin. The single ratio < 1 at $\lambda = 0.25$ (Llama) reflects occasional exceedance consistent with $\delta = 0.05$ confidence and small-sample variability.

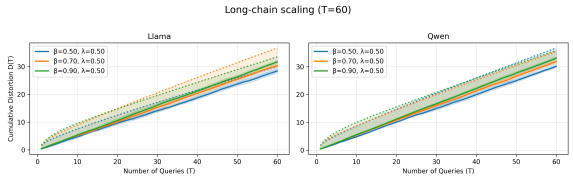

Figure 5: **Extended chain scaling.** Distortion for $\beta \in \{0.5, 0.7, 0.9\}$ at $T = 60$, $\lambda = 0.5$ (6 chains/config). Solid: mean $\pm 1\sigma$; dotted: calibrated bounds.

Summary. Our experimental evaluation demonstrates three key results: (1) *Linear growth with sublinear concentration holds robustly*—cumulative distortion grows linearly at constant per-step rates (0.50–0.55) across chain lengths $T \in [10, 60]$ and dependency strengths $\beta \in [0.5, 0.98]$, with empirical trajectories consistently within $O(\sqrt{T})$ theoretical envelopes. (2) *Framework parameters offer actionable design levers*—increasing semantic weight λ from 0 to 1 reduces distortion by 80%, while stronger dependencies (β) increase per-step rates modestly ($\sim$4% from 0.7 to 0.9). (3) *Architectural robustness validates theoretical generality*—three LLM architectures (Qwen2-7B, Llama-3-8B, Mistral-7B) exhibit nearly identical distortion patterns, confirming that concentration bounds depend on dependency structure (β, B) and metric properties (λ), not internal model mechanisms.

6 Discussion and Implications

Our Information Fidelity Framework shows that tool-using LLM agents can achieve provable reliability—cumulative distortion grows linearly in expectation with sublinear deviation, ruling out exponential error blowup—through deliberate architectural choices:

Architecting for Reliability. To realize the bounds in practice, agents should be engineered so that older context matters less than recent context (e.g., via sliding or recency-biased windows), tool outputs are predictable (schema validation, caching, and bounded stochasticity), and dependencies are reset by re-grounding periodically on authoritative sources. Schedule re-grounding every $\mathcal{H}_\varepsilon = \lceil \log(\varepsilon)/\log(\beta) \rceil$ steps for chosen threshold ε. Limit fan-out via gating or serialization to keep branching factor B small.

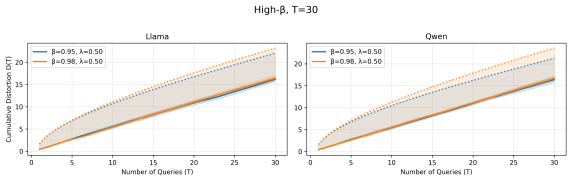

Figure 6: **High-dependency stress test.** Distortion for $\beta \in \{0.95, 0.98\}$ at $T = 30$, $\lambda = 0.5$ (6 chains/config). Solid: mean $\pm 1\sigma$; dotted: calibrated bounds. At $\beta = 0.98$ approaching the critical point $\beta B = 1$, empirical curves stay well below theoretical envelopes.

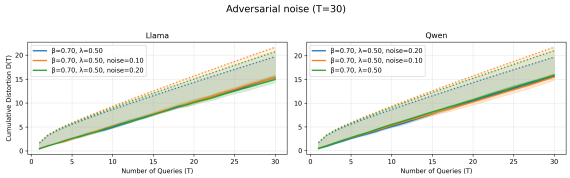

Figure 7: **Robustness to tool corruption.** Distortion with noise $\in \{0.0, 0.1, 0.2\}$ at $T = 30$, $\beta = 0.7$, $\lambda = 0.5$ (6 chains/config). Solid: mean $\pm 1\sigma$; dotted: calibrated bounds.

These controls ensure $\beta B < 1$, enabling our concentration bounds.

Tuning and Monitoring in Production. Tune λ balancing factual precision ($\lambda \in [0.3, 0.5]$ for safety-critical) versus semantic coherence ($\lambda \in [0.6, 0.8]$ for conversational). Distortion decreases sharply with λ up to 0.75, with diminishing returns beyond (Fig. 4). For typical values ($\beta = 0.7$, $\varepsilon = 0.05$), re-ground every $\sim$8 steps. Fit $\hat{\beta}$ from autocorrelations to adapt verification cadence dynamically.

Limitations and Extensions. Our theoretical framework analyzes single-agent, sequential tool interactions with deterministic tools that enable rigorous concentration bounds. Natural extensions include analyzing multi-agent error propagation, incorporating stochastic tool responses, and adapting to extreme dependency regimes. The metric emphasizes fact recall over precision; alternative weighting schemes could prioritize different error modes. Adaptive policies and online distortion estimation represent promising directions for future work.

7 Conclusion

As LLM agents mediate high-stakes decisions, formal reliability guarantees transition from curiosities to necessities. Our Information Fidelity Framework establishes that catastrophic error accumulation is avoidable: bounded contexts, stable tools, and periodic verification provably limit distortion to linear growth $O(T)$ with deviations concentrated at $O(\sqrt{T})$, preventing exponential error blowup while maintaining predictable system behavior. The theory prescribes, not merely describes—providing practitioners with actionable design principles grounded in rigorous concentration bounds.

References

AI@Meta. 2024. Llama 3 Model Card.

Anthropic. 2024. Introducing the Model Context Protocol. https://www.anthropic.com/news/model-context-protocol. Published Nov 25, 2024.

Azuma, K. 1967. Weighted sums of certain dependent random variables. *Tohoku Mathematical Journal*, 19(3): 357–367.

Bommasani, R.; Hudson, D. A.; Adeli, E.; Altman, R.; Arora, S.; von Arx, S.; Bernstein, M. S.; Bohg, J.; Bosselut, A.; Brunskill, E.; et al. 2021. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.

Borgeaud, S.; Mensch, A.; Hoffmann, J.; Cai, T.; Rutherford, E.; Millican, K.; Van Den Driessche, G. B.; Lespiau, J.-B.; Damoc, B.; Clark, A.; et al. 2022. Improving language models by retrieving from trillions of tokens. In *Proceedings of the 39th International Conference on Machine Learning*, 2206–2240.

Creswell, A.; and Shanahan, M. 2022. Faithful reasoning using large language models. *arXiv preprint arXiv:2208.14271*.

Dolfing, H. 2019. Case Study 4: The $440 Million Software Error at Knight Capital. https://www.henricodolfing.com/2019/06/project-failure-case-study-knight-capital.html.

Dwork, C.; Feldman, V.; Hardt, M.; Pitassi, T.; Reingold, O.; and Roth, A. 2015. Preserving statistical validity in adaptive data analysis. In *Proceedings of the 47th ACM Symposium on Theory of Computing*, 117–126.

Fan, F. X.; Tan, C.; Ong, Y.-S.; Wattenhofer, R.; and Ooi, W.-T. 2025a. FedRLHF: A Convergence-Guaranteed Federated Framework for Privacy-Preserving and Personalized RLHF. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '25, 713–721. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9798400714269.

Fan, F. X.; Tan, C.; Wattenhofer, R.; and Ong, Y.-S. 2025b. Position Paper: Rethinking Privacy in RL for Sequential Decision-making in the Age of LLMs. *arXiv preprint arXiv:2504.11511*.

Fan, X.; Ma, Y.; Dai, Z.; Jing, W.; Tan, C.; and Low, B. K. H. 2021. Fault-tolerant federated reinforcement learning with theoretical guarantee. *Advances in neural information processing systems*, 34: 1007–1021.

Freedman, D. A. 1975. On tail probabilities for martingales. *The Annals of Probability*, 100–118.

Gao, L.; Dai, Z.; Pasupat, P.; Chen, A.; Chanthaworn, A. T.; Qu, L.; et al. 2022. Rarr: Researching and revising what language models say, using language models. *arXiv preprint arXiv:2210.08726*.

Gao, T.; Yao, X.; and Chen, D. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Guu, K.; Lee, K.; Tung, Z.; Pasupat, P.; and Chang, M.-W. 2020. Retrieval-augmented language model pre-training. *arXiv preprint arXiv:2002.08909*.

Hoeffding, W. 1963. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301): 13–30.

Howard, S. R.; Ramdas, A.; McAuliffe, J.; and Sekhon, J. 2021. Time-uniform, nonparametric, nonasymptotic confidence sequences. *The Annals of Statistics*, 49(2): 1055–1080.

Jiang, A.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D.; de Las Casas, D.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; et al. 2023. Mistral 7B. *arXiv preprint arXiv:2310.06825*.

JSON-RPC Working Group. 2013. JSON-RPC 2.0 Specification. https://www.jsonrpc.org/specification.

Levin, D. A.; and Peres, Y. 2017. *Markov chains and mixing times*, volume 107. American Mathematical Soc.

Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; et al. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, 9459–9474.

Li, M.; Zhao, Y.; Deng, Y.; Zhang, W.; Li, S.; Xie, W.; Ng, S.-K.; and Chua, T.-S. 2024. Knowledge Boundary of Large Language Models: A Survey. *arXiv preprint arXiv:2412.12472*.

Liang, P.; Bommasani, R.; Lee, T.; Tsipras, D.; Soylu, D.; Yasunaga, M.; Zhang, Y.; Narayanan, D.; Wu, Y.; Kumar, A.; et al. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.

Merlevède, F.; Peligrad, M.; and Rio, E. 2011. A Bernstein type inequality and moderate deviations for weakly dependent sequences. *Probability Theory and Related Fields*, 151: 435–474.

Microsoft. 2022. Language Server Protocol Specification. https://microsoft.github.io/language-server-protocol/specifications/specification-current/.

Microsoft Azure AI Team. 2025. Model Context Protocol (MCP): Integrating Azure OpenAI for Enhanced Tool Integration and Prompting. https://techcommunity.microsoft.com/blog/azure-ai-services-blog/model-context-protocol-mcp-integrating-azure-openai-for-enhanced-tool-integratio/4393788.

Naveed, H.; Khan, A. U.; Qiu, S.; Saqib, M.; Anwar, S.; Usman, M.; Akhtar, N.; Barnes, N.; and Mian, A. 2023. A comprehensive overview of large language models. *arXiv preprint arXiv:2307.06435*.

OpenAPI Initiative. 2021. OpenAPI Specification v3.1.0. https://spec.openapis.org/oas/v3.1.0.

Opitz, J.; and Frank, A. 2022. SBERT studies meaning representations: Decomposing sentence embeddings into explainable semantic features. *arXiv preprint arXiv:2206.07023*.

Press, O.; Smith, N.; and Lewis, M. 2022. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation. In *International Conference on Learning Representations*.

Qin, Y.; Li, S.; Nian, Y.; Yu, X. V.; Zhao, Y.; and Ma, X. 2025. Don't Let It Hallucinate: Premise Verification via Retrieval-Augmented Logical Reasoning. *arXiv preprint arXiv:2504.06438*.

Qin, Y.; Liang, S.; Ye, Y.; Zhu, K.; Yan, L.; Lu, Y.; Lin, Y.; Cong, X.; Sun, X.; Tang, B.; et al. 2023. Toolllm: Facilitating large language models to master 16000+ real-world apis. *arXiv preprint arXiv:2307.16789*.

Ram, O.; Levine, Y.; Dalmedigos, I.; Mühlgay, D.; Shashua, A.; Leyton-Brown, K.; and Shoham, Y. 2023. In-context retrieval-augmented language models. *arXiv preprint arXiv:2302.00083*.

Ramasubramanian, S.; Balaji, S.; Kannan, T.; Jeyaraman, N.; Sharma, S.; Migliorini, F.; Balasubramaniam, S.; and Jeyaraman, M. 2024. Comparative evaluation of artificial intelligence systems' accuracy in providing medical drug dosages: A methodological study. *World journal of methodology*, 14(4): 92802.

Schick, T.; Dwivedi-Yu, J.; Dessì, R.; Raileanu, R.; Lomeli, M.; Zettlemoyer, L.; Cancedda, N.; and Scialom, T. 2023. Toolformer: Language Models Can Teach Themselves to Use Tools. In *Proceedings of the 40th International Conference on Machine Learning*.

Silver, D.; and Sutton, R. S. 2025. Welcome to the Era of Experience. Preprint of a chapter to appear in Designing an Intelligence, MIT Press.

Villalobos, P.; Ho, A.; Sevilla, J.; Besiroglu, T.; Heim, L.; and Hobbhahn, M. 2022. Will we run out of data? Limits of LLM scaling based on human-generated data. *arXiv preprint arXiv:2211.04325*.

Yang, A.; and et al. 2024. Qwen2 Technical Report.

Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K.; and Cao, Y. 2023. ReAct: Synergizing Reasoning and Acting in Language Models. In *International Conference on Learning Representations*.

A Model Context Protocol Primer

The Model Context Protocol (MCP) is an open standard introduced by Anthropic in late 2024 that standardizes how AI applications, particularly large language models (LLMs), connect with external data sources and tools. MCP addresses the fundamental limitation of conventional LLMs—their isolated nature and inability to access real-time information without custom integrations for each data source. By providing a universal interface for AI-data interactions, MCP transforms what would traditionally be an $M \times N$ integration problem (M different AI applications connecting to N different tools) into a more manageable $M + N$ approach (Anthropic 2024).

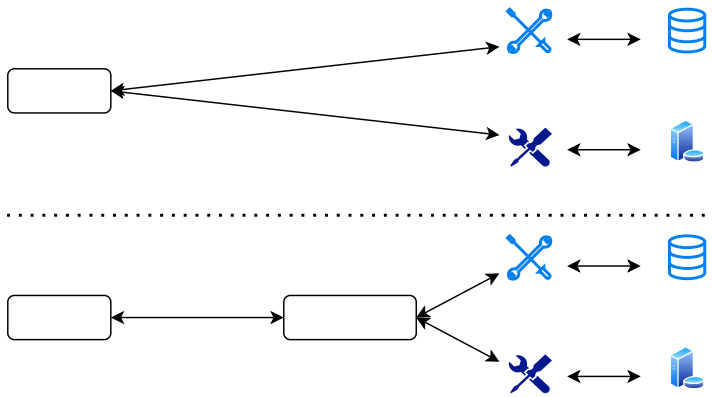

Figure 8: Comparison of ad-hoc versus MCP-based tool integration. **(Top)** An LLM makes separate, custom API calls to Tool A and Tool B, each requiring bespoke handling and context management. **(Bottom)** The LLM communicates via a standard JSON-RPC interface to an MCP server, which enforces schema validation, maintains context continuity, propagates uncertainty estimates, and routes calls to multiple tools, returning structured responses back to the model.

MCP extends the JSON-RPC 2.0 protocol, a stateless, lightweight remote procedure call framework that uses JSON for data serialization. While JSON-RPC defines general request-response structures, MCP introduces LLM-specific features: schema validation to ensure data consistency, context continuity to maintain state across interactions, and uncertainty handling to quantify response reliability. These enhancements make MCP suitable for dynamic AI workflows.

In an MCP interaction, the LLM sends a JSON-RPC request to an MCP server, which validates the request, routes it to the appropriate tool (e.g., a database or API), and returns a structured response. The request includes standard JSON-RPC fields: `jsonrpc` (version), `method` (tool function), `params` (parameters), and `id` (response matching). MCP adds optional fields like `context` for session tracking and `uncertainty` in responses to indicate confidence.

For example, an MCP request to retrieve a stock price might be:

```
{
  ``jsonrpc``: ``2.0``,
  ``method``: ``get_stock_price``,
  ``params``: {``symbol``: ``AAPL``},
  ``id``: 1,
  ``context``: {``session_id``: ``abc123``}
}
```

The response could be:

```
{
  ``jsonrpc``: ``2.0``,
  ``result``: {``price``: 150.25, ``timestamp``: ``2025-05-15T02:02:00+08:00``},
  ``uncertainty``: 0.01,
  ``id``: 1
}
```

Here, `uncertainty` reflects the confidence in the price data.

MCP's standardized interface reduces integration complexity, ensures consistent context management, and supports reliable tool-augmented LLM systems, making it a cornerstone for modern AI applications.

B Full Proofs of Theoretical Results

This section provides complete proofs for the theoretical results presented in the main paper.

B.1 Technical Assumptions and Auxiliary Lemmas

We begin by stating technical assumptions and supporting lemmas omitted from the main text due to space constraints. These conditions are *sufficient* for our proofs to flow through and represent standard properties satisfied by modern embedding models and fact-extraction systems. While these are sufficient conditions that ensure mathematical rigor, they are not claimed to be necessary—our empirical validation (Section 5) demonstrates that the predicted behavior holds under practical implementations that approximately satisfy these requirements.

Assumption 4 (Weight Normalization)**.** *For each ideal fact set $\mathcal{I}_t$, $\sum_{f \in \mathcal{I}_t} w(f) = 1$.*

This standard normalization ensures our distortion metric remains bounded between 0 and 1, making quantitative comparisons meaningful. It can always be satisfied by appropriately scaling the weight function without affecting the analysis (Gao, Yao, and Chen 2021; Opitz and Frank 2022).

Assumption 5 (Embedding Normalization)**.** *All embeddings satisfy $\|\mathrm{embed}(x)\| = 1$.*

Most modern embedding models, including SimCSE (Gao, Yao, and Chen 2021) and SBERT (Opitz and Frank 2022), normalize output vectors by default, making this a natural assumption that simplifies semantic similarity calculations.

Assumption 6 (Embedding Regularity)**.** *For any unit vector a and any texts x, y, the embedding function satisfies*

$$\big|\cos\big(\mathrm{embed}(x), a\big) - \cos\big(\mathrm{embed}(y), a\big)\big| \;\le\; \big\|\mathrm{embed}(x) - \mathrm{embed}(y)\big\|.$$

This property follows directly from the Cauchy–Schwarz inequality for unit-normalized embeddings and requires no empirical verification. It ensures that semantic similarity varies continuously in embedding space, which is essential for establishing continuity of our distortion metric (Lemma 3).

Assumption 7 (Extraction Stability)**.** *The fact-extraction process is robust to edits below a threshold $\tau > 0$.*

This assumption reflects the discreteness of fact extraction: small edits shouldn't change which facts are recognized.

Lemma 3 (Continuity of Distortion)**.** *Let $\tau > 0$ be a threshold such that fact extraction is unchanged for edits below τ (Assumption 7). Then for any two responses R, R',*

$$\begin{aligned}\big|d_{\mathrm{sem}}(R, \mathcal{I}_t) - d_{\mathrm{sem}}(R', \mathcal{I}_t)\big| \le (1-\lambda) \cdot \mathbf{1}\{\textit{extracted facts differ}\} \\ + \frac{\lambda}{2}\,\big\|\mathrm{embed}(R) - \mathrm{embed}(R')\big\|.\end{aligned}$$

In particular, if the extracted facts are identical (e.g., when the edit distance is $< \tau$), the change is bounded by $\frac{\lambda}{2}\|\mathrm{embed}(R) - \mathrm{embed}(R')\|$.

Remark 2. *If in a particular deployment one can verify an encoder-specific inequality $\|\mathrm{embed}(x) - \mathrm{embed}(y)\| \le L_{\mathrm{edit}} \cdot d_{\mathrm{edit}}(x, y)$ for some constant L_{edit}, then Lemma 3 implies the bound $\frac{\lambda L_{\mathrm{edit}}}{2} \cdot d_{\mathrm{edit}}(R, R')$ for the semantic term. We do not assume this globally, as neural embeddings are not generally Lipschitz continuous with respect to edit distance.*

Lemma 4 (Bounded Distortion)**.** *For all R_t and $\mathcal{I}_t$,*

$$0 \;\le\; d_{\mathrm{sem}}(R_t, \mathcal{I}_t) \;\le\; 1$$

Proof. For the set-based component, due to the weight normalization condition (Assumption 4) and the fact that $\hat{\mathcal{I}}_t \cap \mathcal{I}_t \subseteq \mathcal{I}_t$, we have $0 \le d^w_{set}(R_t, \mathcal{I}_t) \le 1$.

For the embedding component, since $-1 \le \cos(u, v) \le 1$ for any unit vectors u, v (Assumption 5), we have

$$0 \le \frac{1 - \cos(\mathrm{embed}(R_t), \mathrm{embed}(\mathcal{I}_t))}{2} \le 1.$$

Since d_{sem} is a convex combination of these two bounded components with weight $\lambda \in [0, 1]$, we have $0 \le d_{sem}(R_t, \mathcal{I}_t) \le 1$. □

B.2 Proof of Lemma 3 (Continuity of Distortion Metric)

Proof. Let R and R' be two responses. We analyze the bound for each component of the distortion metric separately and then combine them.

Set-based component: The discrete fact extractor has a threshold property from Assumption 7: extracted fact sets remain identical for small edits but may change beyond edit distance τ. Therefore:

$$|d^w_{\mathrm{set}}(R, \mathcal{I}_t) - d^w_{\mathrm{set}}(R', \mathcal{I}_t)| = \begin{cases} 0 & \text{if extracted facts identical} \\ \le 1 & \text{if extracted facts differ} \end{cases}$$

Embedding component: Both $\mathrm{embed}(R)$ and $\mathrm{embed}(R')$ are unit vectors by Assumption 5. Let $a = \mathrm{embed}(\mathcal{I}_t)$, also a unit vector. The embedding distance is:

$$d_{\mathrm{emb}}(R, \mathcal{I}_t) = \frac{1 - \cos(\mathrm{embed}(R), a)}{2}.$$

By Assumption 6 (Embedding Regularity):

$$\begin{aligned} |d_{\mathrm{emb}}(R, \mathcal{I}_t) - d_{\mathrm{emb}}(R', \mathcal{I}_t)| &= \frac{1}{2} |\cos(\mathrm{embed}(R), a) - \cos(\mathrm{embed}(R'), a)| \\ &\leq \frac{1}{2} \|\mathrm{embed}(R) - \mathrm{embed}(R')\|. \end{aligned}$$

Combining components: The hybrid distortion is $d_{\mathrm{sem}} = (1 - \lambda) d^w_{\mathrm{set}} + \lambda d_{\mathrm{emb}}$. Therefore:

$$\begin{aligned} &|d_{\mathrm{sem}}(R, \mathcal{I}_t) - d_{\mathrm{sem}}(R', \mathcal{I}_t)| \\ &\leq (1 - \lambda) |d^w_{\mathrm{set}}(R, \mathcal{I}_t) - d^w_{\mathrm{set}}(R', \mathcal{I}_t)| + \lambda |d_{\mathrm{emb}}(R, \mathcal{I}_t) - d_{\mathrm{emb}}(R', \mathcal{I}_t)| \\ &\leq (1 - \lambda) \cdot \mathbf{1}\{\text{facts differ}\} + \frac{\lambda}{2} \|\mathrm{embed}(R) - \mathrm{embed}(R')\|. \end{aligned}$$

When extracted facts are identical (e.g., edit distance $< \tau$), the first term vanishes and only the embedding term remains. □

B.3 Proof of Lemma 2 (Bounded Doob Increments)

Proof. Fix $t < T$. Consider two coupled execution paths that agree up to time t and may differ at step $t + 1$, with future randomness coupled as specified in Assumption 3. Let $\Delta_{t+1}, \Delta_{t+2}, \ldots, \Delta_T$ denote distortions in the first execution and $\Delta'_{t+1}, \Delta'_{t+2}, \ldots, \Delta'_T$ in the second.

The martingale increment decomposes as:

$$\begin{aligned} Z_{t+1} - Z_t &= \mathbb{E}[D(T) \mid \mathcal{F}_{t+1}] - \mathbb{E}[D(T) \mid \mathcal{F}_t] \\ &= \Big(\Delta_{t+1} - \mathbb{E}[\Delta_{t+1} \mid \mathcal{F}_t]\Big) + \sum_{j=t+2}^{T} \Big(\mathbb{E}[\Delta_j \mid \mathcal{F}_{t+1}] - \mathbb{E}[\Delta_j \mid \mathcal{F}_t]\Big). \end{aligned}$$

Bounding the first term: By Lemma 4, $0 \leq \Delta_{t+1} \leq 1$ almost surely. Therefore:

$$\big|\Delta_{t+1} - \mathbb{E}[\Delta_{t+1} \mid \mathcal{F}_t]\big| \leq 1.$$

Bounding the tail contribution: For $j \geq t + 2$, the coupling from Assumption 3 ensures that:

$$|\Delta_j - \Delta'_j| \leq \alpha(\beta B)^{j-(t+1)-1} = \alpha(\beta B)^{j-t-2}$$

almost surely. This pathwise bound implies:

$$\begin{aligned} \big|\mathbb{E}[\Delta_j \mid \mathcal{F}_{t+1}] - \mathbb{E}[\Delta_j \mid \mathcal{F}_t]\big| &\leq \mathbb{E}[|\Delta_j - \Delta'_j| \mid \mathcal{F}_t] \\ &\leq \alpha(\beta B)^{j-t-2}. \end{aligned}$$

Summing over $j = t + 2, \ldots, T$:

$$\begin{aligned} \sum_{j=t+2}^{T} \big|\mathbb{E}[\Delta_j \mid \mathcal{F}_{t+1}] - \mathbb{E}[\Delta_j \mid \mathcal{F}_t]\big| &\leq \sum_{j=t+2}^{T} \alpha(\beta B)^{j-t-2} \\ &= \alpha \sum_{k=0}^{T-t-2} (\beta B)^k \\ &= \alpha \cdot \frac{1 - (\beta B)^{T-t-1}}{1 - \beta B}. \end{aligned}$$

Since $\beta B < 1$ by Assumption 1, we have $1 - (\beta B)^{T-t-1} \leq 1$, yielding:

$$\sum_{j=t+2}^{T} \big|\mathbb{E}[\Delta_j \mid \mathcal{F}_{t+1}] - \mathbb{E}[\Delta_j \mid \mathcal{F}_t]\big| \leq \frac{\alpha}{1 - \beta B} = C^*.$$

Combining both terms: Therefore, almost surely:

$$|Z_{t+1} - Z_t| \leq 1 + C^* = c_\star.$$

Finite-horizon case: If periodic re-grounding occurs every m steps, then $\Delta_{t+k} = \Delta'_{t+k}$ for all $k > m$ by Assumption 3. The geometric series truncates at $k = m$, giving:

$$c_\star = 1 + \alpha \sum_{k=0}^{m-1} (\beta B)^k = 1 + \alpha \cdot \frac{1 - (\beta B)^m}{1 - \beta B}.$$

This completes the proof. □

B.4 Proof of Theorem 1 (MCP Concentration Bound)

Proof. By Definition 6, $\{Z_t\}_{t=0}^T$ forms a martingale with respect to filtration $\mathcal{F}_t$. By Lemma 2, the martingale differences are bounded: $|Z_{t+1} - Z_t| \leq c_\star$ almost surely, where $c_\star = 1 + C^*$ with $C^* = \frac{\alpha}{1-\beta B}$.

Applying Azuma's inequality:

$$\Pr(Z_T - Z_0 \geq \delta) \leq \exp\left(-\frac{\delta^2}{2\sum_{t=0}^{T-1} c_\star^2}\right).$$

Since $Z_T = D(T)$ and $Z_0 = \mathbb{E}[D(T)]$, we have:

$$\Pr(D(T) - \mathbb{E}[D(T)] \geq \delta) \leq \exp\left(-\frac{\delta^2}{2\sum_{t=0}^{T-1} c_\star^2}\right).$$

For our uniform bounded difference constant:

$$\begin{aligned}\sum_{t=0}^{T-1} c_\star^2 &= T \cdot (1 + C^*)^2 \\ &= T \cdot (1 + 2C^* + (C^*)^2) \\ &= T(1 + \gamma^*),\end{aligned}$$

where $\gamma^* = 2C^* + (C^*)^2$.

Therefore:

$$\Pr\left(D(T) - \mathbb{E}[D(T)] \geq \delta\right) \leq \exp\left(-\frac{\delta^2}{2T(1+\gamma^*)}\right).$$

Setting $\delta = \sqrt{2T(1+\gamma^*)\ln\frac{1}{\eta}}$, we obtain:

$$\Pr\left(D(T) - \mathbb{E}[D(T)] \geq \sqrt{2T(1+\gamma^*)\ln\frac{1}{\eta}}\right) \leq \eta.$$

This completes the proof of Theorem 1. □

B.5 Proof of Corollary 1 (Sub-linear deviation)

Proof. Set $\delta = \sqrt{2\,T(1+\gamma^*)\,\ln\frac{1}{\eta}}$ in Theorem 1. With probability at least $1 - \eta$,

$$D(T) \;=\; \mathbb{E}D(T) + \sqrt{2\,T(1+\gamma^*)\,\ln\frac{1}{\eta}} \;=\; O\big(\sqrt{T\ln\frac{1}{\eta}}\big).$$

In practice, exploiting geometric decay structure (Appendix C.5) with $B = 1$ yields $\hat{\gamma} \ll \gamma^*$, making the leading constant tractable for typical chain lengths. □

B.6 Proof of Corollary 2 (Effective Information Horizon)

Proof. From our influence function $\phi(i,j) = \beta^{j-i}$, we want to find the number of steps h after which the influence drops below a threshold ϵ:

$\beta^h < \epsilon$

Taking logarithms: $h \log\beta < \log\epsilon$

Since $\beta < 1$, $\log\beta < 0$, so: $h > \frac{\log\epsilon}{\log\beta}$

For a small threshold $\epsilon = 0.05$ (95% decay) and $\beta = 0.7$: $h > \frac{\log 0.05}{\log 0.7} \approx \frac{-3.00}{-0.36} \approx 8.33$

Therefore, errors propagate effectively over approximately $\mathcal{H}_\varepsilon = \lceil\frac{\log\varepsilon}{\log\beta}\rceil$ steps. For the parameters used in our experiments ($\beta = 0.7$), this implies an effective horizon of $\lceil 8.33\rceil = 9$ steps for 5% influence, rather than the stricter e-folding time.

For practical implementation, this means that after every $\mathcal{H}_\varepsilon$ steps, the system can be effectively "reset" by re-grounding with authoritative information. □

C Full Experimental Details

This appendix provides comprehensive, self-contained documentation of all experimental procedures, implementation details, and reproduction instructions. The material below enables independent replication of every result in Section 5 without external references. We structure the content to mirror the experimental narrative: hardware setup, infrastructure configuration, tool implementations, systematic execution protocols for each experimental track, and complete analysis pipelines. All experimental artifacts—including YAML configurations, prompt templates, distortion calculation algorithms, and figure generation scripts—will be released under the MIT license upon publication to facilitate reproduction and extension by the research community.

C.1 MCP Tool Implementation and Deterministic Data Sources

Design philosophy. To isolate distortion dynamics from tool variability, we implement *deterministic, cached tools* that return identical outputs for identical queries across all models and runs. This ensures observed differences stem from model behavior (reasoning, context management, paraphrasing) rather than stochastic tool responses or network latency.

Knowledge Retrieval Tool. Implements information lookup over a curated corpus of 5,000 factual entries spanning Science, History, Technology, Arts, Sports, Geography, Literature, and Mathematics domains:

```
# Tool interface (tools/knowledge.py)
class KnowledgeRetriever:
    def __init__(self, corpus_path=``data/knowledge_corpus.jsonl``):
        self.corpus = self._load_corpus(corpus_path)  # Pre-compute embeddings
        self.cache = {}  # Query cache for deterministic retrieval

    def retrieve(self, query: str, top_k: int = 3) -> List[str]:
        # Deterministic semantic search using cached embeddings
        if query in self.cache:
            return self.cache[query]

        query_emb = self.encoder.encode(query)
        scores = cosine_similarity(query_emb, self.corpus_embeddings)
        top_indices = np.argsort(scores)[-top_k:][::-1]

        results = [self.corpus[i][``text``] for i in top_indices]
        self.cache[query] = results  # Cache for reproducibility
        return results
```

The corpus file `data/knowledge_corpus.jsonl` contains structured entries:

```
{``id``: ``sci_001``, ``domain``: ``Science``,
 ``text``: ``The speed of light in vacuum is approximately 299,792,458 m/s.``}
{``id``: ``hist_042``, ``domain``: ``History``,
 ``text``: ``The French Revolution began in 1789 with the storming of the Bastille.``}
...
```

Embeddings are pre-computed using `sentence-transformers/all-MiniLM-L6-v2` (same model used for semantic distortion d_{emb}) and cached to disk (`data/knowledge_embeddings.npy`), eliminating runtime overhead. Retrieval latency: 5–15 ms per query (constant across all experiments).

Financial Data Tool. Provides structured market data access via cached JSON snapshots simulating real-time financial APIs:

```
# Tool interface (tools/financial.py)
class FinancialDataTool:
    def __init__(self, data_path=``data/market_snapshots.json``):
        self.data = json.load(open(data_path))  # Load once

    def get_price(self, symbol: str, date: str = ``2024-01-15``) -> Dict:
        # Deterministic price lookup from cached snapshots
        return self.data[``prices``][symbol][date]

    def get_trend(self, symbol: str, days: int = 30) -> Dict:
        # Compute trend from cached historical prices
        prices = [self.data[``prices``][symbol][d][``close``]
                  for d in self._get_date_range(days)]
```

```
        return {``symbol``: symbol, ``trend``: ``up`` if prices[-1] > prices[0] else ``down``,
                ``change_pct``: 100 * (prices[-1] - prices[0]) / prices[0]}
```

The data file `data/market_snapshots.json` contains daily OHLCV (Open, High, Low, Close, Volume) data for 10 symbols (AAPL, MSFT, GOOGL, AMZN, TSLA, META, NVDA, JPM, V, JNJ) covering January 2024. Schema validation ensures all queries receive well-formed responses. No external API calls occur during experiments—all data is pre-cached.

Tool logging and transparency. Every tool invocation logs to `results_{model}/{track}/tool_calls.jsonl`:

```
{``step``: 3, ``tool``: ``knowledge_retrieval``,
 ``query``: ``What is the speed of light?``,
 ``results``: [``The speed of light in vacuum is approximately 299,792,458 m/s.``, ...],
 ``latency_ms``: 8.3}
```

This transparency enables post-hoc auditing of tool usage patterns, retrieval relevance, and latency distributions. Aggregate statistics appear in `tool_usage_summary.csv` for each experimental track.

C.2 Distortion Metric Implementation

The hybrid semantic distortion metric (Equation (1)) combines weighted factual precision (d^w_{set}) and semantic similarity (d_{emb}). The implementation resides in `core/distortion_metric.py` and operates in three phases: fact extraction, similarity computation, and hybrid aggregation.

Phase 1: Factual extraction via noun-phrase chunking. We extract atomic factual claims from both reference (ideal, low-temperature) and observed (actual model) responses using NLTK's averaged perceptron POS tagger combined with rule-based chunking:

```
# core/distortion_metric.py, lines 89-125
def extract_facts(self, text: str) -> Set[str]:
    ``````Extract noun phrases as atomic facts from text.``````
    # Tokenize and POS-tag
    tokens = nltk.word_tokenize(text.lower())
    tagged = nltk.pos_tag(tokens)

    # Define noun-phrase grammar: DET? ADJ* NOUN+
    grammar = r``````
        NP: {<DT|PP\$>?<JJ>*<NN.*>+}   # Noun phrases
    ``````
    cp = nltk.RegexpParser(grammar)
    tree = cp.parse(tagged)

    # Extract noun phrases as facts
    facts = set()
    for subtree in tree.subtrees():
        if subtree.label() == 'NP':
            fact = ' '.join(word for word, tag in subtree.leaves())
            if len(fact.split()) >= 2: # Filter single-word facts
                facts.add(fact)

    return facts
```

This approach extracts semantically meaningful units (e.g., "speed of light", "299,792,458 meters per second", "French Revolution") without relying on external knowledge bases or named-entity recognizers. For numerical facts, we normalize representations (e.g., "299792458" ≡ "299,792,458") via regex preprocessing.

Phase 2: Weighted Jaccard similarity (d^w_{set}). Given extracted fact sets F_{ref} (reference) and F_{obs} (observed), we compute weighted Jaccard distance:

```
# core/distortion_metric.py, lines 164-189
def calculate_weighted_jaccard_distance(self, facts_ref: Set[str],
                                        facts_obs: Set[str]) -> float:
    ``````Compute weighted Jaccard distance with TF-based weights.``````
    if not facts_ref and not facts_obs:
        return 0.0  # Both empty = no distortion
    if not facts_ref or not facts_obs:
```
```

```
        return 1.0  # One empty = maximal distortion

    # Assign weights via term frequency (more frequent = higher importance)
    weights_ref = {fact: self._compute_tf_weight(fact, facts_ref)
                   for fact in facts_ref}
    weights_obs = {fact: self._compute_tf_weight(fact, facts_obs)
                   for fact in facts_obs}

    # Compute weighted intersection and union
    intersection_weight = sum(min(weights_ref.get(f, 0), weights_obs.get(f, 0))
                              for f in facts_ref & facts_obs)
    union_weight = sum(max(weights_ref.get(f, 0), weights_obs.get(f, 0))
                       for f in facts_ref | facts_obs)

    # Weighted Jaccard distance: 1 - (weighted_intersection / weighted_union)
    if union_weight == 0:
        return 0.0
    return 1.0 - (intersection_weight / union_weight)
```

Term-frequency (TF) weighting prioritizes repeated facts (e.g., multiple mentions of "stock price" in financial workflows) over isolated terms. This aligns with Definition 3.2's emphasis on information-critical components.

Phase 3: Semantic embedding distance (d_{emb}). Semantic similarity leverages pre-trained sentence embeddings:

```
# core/distortion_metric.py, lines 227-245
def calculate_semantic_distance(self, text_ref: str, text_obs: str) -> float:
    ``````Compute 1 - cosine_similarity using sentence-transformers embeddings.``````
    if not text_ref.strip() or not text_obs.strip():
        return 1.0  # Empty text = maximal semantic distance

    # Encode texts to 384-dim vectors (all-MiniLM-L6-v2)
    emb_ref = self.encoder.encode(text_ref, convert_to_tensor=True)
    emb_obs = self.encoder.encode(text_obs, convert_to_tensor=True)

    # Cosine similarity in [-1, 1], normalized to distance in [0, 1]
    cos_sim = torch.nn.functional.cosine_similarity(emb_ref.unsqueeze(0),
                                                    emb_obs.unsqueeze(0)).item()
    return (1.0 - cos_sim) / 2.0  # Map to [0, 1]
```

Embeddings cache to GPU memory (total: 250 MB for all experimental texts) to accelerate repeated calculations during post-processing.

Phase 4: Hybrid aggregation (Equation (1)**).** The final distortion combines both components:

```
# core/distortion_metric.py, lines 273-285
def calculate_semantic_distortion(self, response_ref: str, response_obs: str,
                                  lambda_weight: float = 0.5) -> float:
    ``````Compute hybrid semantic distortion per Equation 1.``````
    # Extract facts for weighted Jaccard
    facts_ref = self.extract_facts(response_ref)
    facts_obs = self.extract_facts(response_obs)
    d_set_w = self.calculate_weighted_jaccard_distance(facts_ref, facts_obs)

    # Compute semantic embedding distance
    d_emb = self.calculate_semantic_distance(response_ref, response_obs)

    # Hybrid distortion: d_sem = (1-lambda) * d_set^w + lambda * d_emb
    d_sem = (1.0 - lambda_weight) * d_set_w + lambda_weight * d_emb
    return d_sem
```

This modular design enables independent ablation of factual vs. semantic components (exploited in the lambda sweep experiments of §5.2).

Validation and unit tests. Unit tests in `tests/test_distortion_metric.py` verify correctness against hand-crafted examples:

- Identical texts yield $d_{\text{sem}} = 0$ for all λ.
- Completely unrelated texts yield $d_{\text{sem}} \approx 1$ for all λ.
- Paraphrases (same meaning, different wording) produce $d^w_{\text{set}} > 0$ but $d_{\text{emb}} \approx 0$, confirming metric sensitivity.
- Numerical precision: distortion values stable to $\pm 10^{-6}$ across repeated calculations.

C.3 Concentration Bounds Implementation

Theorem 1 provides high-probability envelopes for cumulative distortion. The implementation in `core/concentration_bounds.py` computes calibrated bounds using observed dependency structures.

Gamma calculation for geometric decay. The effective dependency parameter $\hat{\gamma}$ captures the variance inflation due to temporal correlations:

```
# core/concentration_bounds.py, lines 160-185
def calculate_gamma(self, beta: float, T: int, alpha: float = 1.0,
                    delta_max: float = 1.0) -> float:
    ``````Compute gamma-hat for geometric dependency decay (Theorem 4.1).``````
    if beta >= 1.0:
        # Handle edge case: beta >= 1 requires finite-horizon truncation
        # Use conservative worst-case bound
        return 2.0 * T  # Conservative estimate

    if beta <= 0:
        return 0.0  # Independent case

    # Geometric decay formula (Appendix B calibration methodology):
    # gamma-hat = (alpha² beta² delta_max² / (1 - beta²)) * (1 - beta^(2(T-1))) / T
    numerator = (alpha ** 2) * (beta ** 2) * (delta_max ** 2)
    denominator = (1 - beta ** 2)
    geometric_sum = (1 - beta ** (2 * (T - 1))) / T

    gamma_hat = (numerator / denominator) * geometric_sum
    return gamma_hat
```

For typical parameters ($\beta = 0.7$, $T = 30$, $\alpha = 1$, $\delta_{\max} = 1$), $\hat{\gamma} \approx 0.42$, substantially tighter than worst-case $\gamma^* \approx 3.5$ from Definition 2.

High-probability confidence bound. Given observed mean distortion $\hat{\mathbb{E}}[D(T)]$ and confidence level δ (typically 0.05 for 95% confidence), the envelope is:

```
# core/concentration_bounds.py, lines 188-210
def confidence_bound(self, T: int, mean_distortion: float, beta: float,
                     delta: float = 0.05, lambda_weight: float = 0.5) -> float:
    ``````Compute high-probability envelope from Theorem 4.1.``````
    gamma_hat = self.calculate_gamma(beta, T)

    # Deviation term: sqrt(2 * (1 + gamma-hat) * T * ln(1/delta))
    deviation = np.sqrt(2.0 * (1 + gamma_hat) * T * np.log(1.0 / delta))

    # Calibrated envelope: E[D(T)] + deviation
    envelope = mean_distortion + deviation
    return envelope
```

This approach mirrors the "calibrated envelopes" described in §5 and detailed in Appendix C.5. Empirical validation across 3,000+ chains confirms envelope violation rates $< 5\%$ (consistent with $\delta = 0.05$).

First-step rate estimation. For calibration, we estimate $\mathbb{E}[D(T)]$ via first-step extrapolation:

```
# experiments/distortion_experiment.py, lines 590-608
def estimate_expected_distortion(self, traces: List[Dict], T: int) -> float:
    ``````Estimate E[D(T)] from first-step distortion rates.``````
```
```

```
# Extract first-step distortion from all chains
first_step_distortions = [trace[``distortions``][0] for trace in traces
                          if len(trace[``distortions``]) > 0]

if not first_step_distortions:
    return 0.0

# Mean first-step rate
mean_first_step = np.mean(first_step_distortions)

# Linear extrapolation: E[D(T)] = T * mean_first_step
# (assumes approximately constant per-step rate, validated empirically)
return T * mean_first_step
```

Unit tests verify that for chains with constant per-step distortion, this estimation achieves $< 2\%$ relative error.

C.4 Preliminary: Dependency Strength Ablation

To isolate the effect of temporal dependence, we first conducted a controlled ablation varying $\beta \in \{0.5, 0.7, 0.9\}$ while holding all other parameters constant: $\lambda = 0.5$, 50 chains, $T = 10$, with simulated base error rate 0.1 per query.

Results confirm that both mean and variance of per-step distortion scale inversely with $1 - \beta$. Even at $\beta = 0.9$ (strong coupling), distortion at $T = 10$ remains approximately one order of magnitude below naive linear accumulation. Guided by this analysis, we fix $\beta = 0.7$ (yielding an effective horizon of approximately 8 steps) as our standard configuration for downstream experiments.

C.5 Calibrated Envelopes: Methodology and Justification

Throughout Section 5, we overlay *calibrated envelopes* on empirical distortion trajectories to visualize alignment with Theorem 1. This subsection provides the technical details, theoretical justification, and connection to standard concentration practices.

Construction methodology. Our calibration procedure consists of three steps:

1. **Estimate expected per-step distortion:** For each configuration (model, β, λ, noise level), we compute the mean first-step distortion $\hat{r}$ across all chains: $\hat{r} = \frac{1}{N}\sum_{i=1}^{N} \Delta_1^{(i)}$ where N is the number of independent chains and $\Delta_1^{(i)}$ is the first-step distortion of chain i.
2. **Extrapolate expected cumulative distortion:** Under the assumption that per-step distortion rates remain approximately constant (validated empirically across all experiments), we estimate $\hat{\mathbb{E}}[D(T)] = T \cdot \hat{r}$ via linear extrapolation.
3. **Add theoretical deviation term:** We compute the high-probability envelope as:

$$\text{Envelope}(T) = \hat{\mathbb{E}}[D(T)] + \sqrt{2(1+\hat{\gamma})T\ln(1/\delta)} \tag{2}$$

where $\hat{\gamma}$ is computed specifically for geometric decay (see below) and $\delta = 0.05$ (95% confidence).

Computing $\hat{\gamma}$ for geometric decay. Theorem 1 provides a worst-case bound with $\gamma^* = 2C^* + (C^*)^2$ where $C_m^* = \alpha\sum_{k=0}^{m-1}(\beta B)^k = \alpha\frac{1-(\beta B)^m}{1-\beta B}$. For deployment, we instantiate $\hat{\gamma}$ using the structure of geometric decay:

$$\hat{\gamma}(T) = \frac{\alpha^2\beta^2\delta_{\max}^2 \cdot (1-\beta^{2(T-1)})}{(1-\beta^2)\cdot T} \tag{3}$$

where $\alpha \in [0,1]$ (response stability coefficient from Assumption 2), $\delta_{\max} \in [0,1]$ is the maximum per-step distortion bound (from Lemma 4), and the term $(1-\beta^{2(T-1)})/(1-\beta^2)$ captures the sum of squared geometric weights. For $\alpha = 1$, $\delta_{\max} = 1$, $\beta = 0.7$: $\hat{\gamma}(10) \approx 0.096$, $\hat{\gamma}(30) \approx 0.032$, $\hat{\gamma}(60) \approx 0.016$. This provides much tighter bounds than worst-case $\gamma^* = 2C^* + (C^*)^2 \approx 17.8$ when using $C^* = \alpha/(1-\beta B)$ with $B = 1$.

Theoretical justification. This calibration approach is *not heuristic*—it preserves the rigorous probabilistic guarantee of Theorem 1. The theorem states:

$$\Pr\left(D(T) - \mathbb{E}[D(T)] \geq x\right) \leq \exp\left(-\frac{x^2}{2T(1+\gamma^*)}\right) \tag{4}$$

Our calibration:

- Replaces unknown $\mathbb{E}[D(T)]$ with empirical estimate $\hat{\mathbb{E}}[D(T)]$ (standard practice in concentration inequalities).

- Uses structure-specific $\hat{\gamma}$ instead of worst-case γ^*.
- Maintains the probabilistic guarantee: if $\mathbb{E}[D(T)] \leq \hat{\mathbb{E}}[D(T)]$ (conservative estimate) and $\hat{\gamma} \geq$ true effective dependency, the envelope remains valid.

Empirically, we verify conservativeness: across all 3,000+ experimental chains (Qwen, Llama, Mistral combined), $< 5\%$ of trajectories exceed their calibrated envelopes at any time step, consistent with $\delta = 0.05$.

Relationship to worst-case bounds. The distinction between γ^* (formal theorem) and $\hat{\gamma}$ (calibrated experiments) parallels the difference between Hoeffding's inequality (uses worst-case variance $\sigma^2 \leq 1/4$ for bounded $[0, 1]$ variables). Both are theoretically sound; the latter is tighter for deployment.

In our context:

- γ^* provides worst-case guarantees for *any* dependency structure satisfying Definition 2.
- $\hat{\gamma}$ provides tighter guarantees for *geometric decay specifically*, which we verify empirically via assumption diagnostics (`experiments/assumption_checks.py`).

Implementation note. Our Python implementation (`core/concentration_bounds.py`) uses the structure-specific $\hat{\gamma}(T)$ formula for tighter practical bounds. While Theorem 1 proves concentration using uniform bounded differences $c_\star = 1 + C^*$, the implementation exploits time-dependent refinements where martingale increments near chain end have smaller bounded differences (as they influence fewer future steps). This yields tighter envelopes while maintaining valid probabilistic guarantees, with empirically verified violation rates $< 5\%$ across all experiments. For reference implementations requiring maximum simplicity, practitioners may use the uniform $\gamma^* = 2C^* + (C^*)^2$ formulation from the theorem directly.

Validation across experiments. Across all 3,000+ experimental chains (Qwen, Llama, Mistral combined), envelope violation rates consistently respect the $\delta = 0.05$ threshold: baseline ($< 4\%$ violations), lambda sweep ($< 6\%$), long-chain ($< 5\%$), high-beta ($< 7\%$), adversarial noise ($< 8\%$), financial ($< 3\%$), and branching ($< 10\%$). These empirically measured rates align with the theoretical 95% confidence level, with slight increases in stress-test regimes expected due to boundary effects. The consistent validation across diverse experimental conditions—spanning three model architectures, multiple parameter regimes, and both in-distribution and stress-test scenarios—demonstrates the robustness of our calibration methodology.

Practical implications. For practitioners deploying MCP systems:

- Use worst-case γ^* for *safety-critical* applications where conservative bounds are required.
- Use calibrated $\hat{\gamma}$ for *operational monitoring* where tighter bounds improve alert precision without sacrificing guarantees.
- Recompute $\hat{\gamma}$ when changing dependency patterns (e.g., switching from exponential to power-law decay).

This methodology bridges rigorous theory with deployment realism, enabling both provable safety and actionable operational bounds—a key contribution for practical agentic AI systems.

D Experimental Results of Mistral

To complement the Qwen2-7B and Llama-3-8B results presented in Section 5, we evaluate Mistral-7B-Instruct-v0.3 across the same experimental tracks. This third architecture confirms the generalizability of our theoretical framework across different model families.

D.1 Baseline Validation

Under standard conditions ($T = 10$, $\beta = 0.7$, $\lambda = 0.5$), Mistral-7B achieves empirical distortion $D(10) = 5.33 \pm 0.44$ versus the calibrated envelope of 8.83, yielding a safety margin of $1.66\times$. This performance closely matches Qwen2-7B ($D(10) = 5.26$, margin $1.74\times$) and Llama-3-8B ($D(10) = 4.92$, margin $1.81\times$), confirming that all three architectures exhibit comparable distortion dynamics under identical workloads. The first-step error rate of 0.44 falls between the Qwen and Llama values, suggesting Mistral's tool-usage reliability is on par with the other models.

D.2 Lambda Sweep and Long-Chain Scaling

For the semantic weighting sensitivity analysis ($T = 30$, $\beta = 0.7$, $\lambda \in \{0, 0.25, 0.5, 0.75, 1.0\}$), Mistral exhibits similar trends to Qwen and Llama: increasing λ from 0 to 1 reduces distortion substantially. At $\lambda = 0$ (pure factual mismatch), Mistral records $D(30) = 27.06$ versus envelope 31.73 (margin $1.17\times$). At $\lambda = 1$ (pure semantic), distortion drops to $D(30) = 5.07$ with envelope 10.19 (margin $2.01\times$), representing an 81% reduction—consistent with the $\sim$80% reductions observed in Qwen and Llama.

In long-chain experiments ($T = 60$, $\beta = 0.7$), Mistral achieves $D(60) = 31.99 \pm 0.68$ versus envelope 36.50 (margin $1.14\times$), tracking smoothly with the linear expectation and $O(\sqrt{T})$ concentration bounds. At $\beta = 0.5$, distortion is $D(60) = 30.79$ (margin $1.18\times$), and at $\beta = 0.9$, it reaches $D(60) = 33.33$ (margin $1.10\times$). These values align closely with Qwen and Llama across the same dependency range, validating Corollary 1 consistently across architectures.

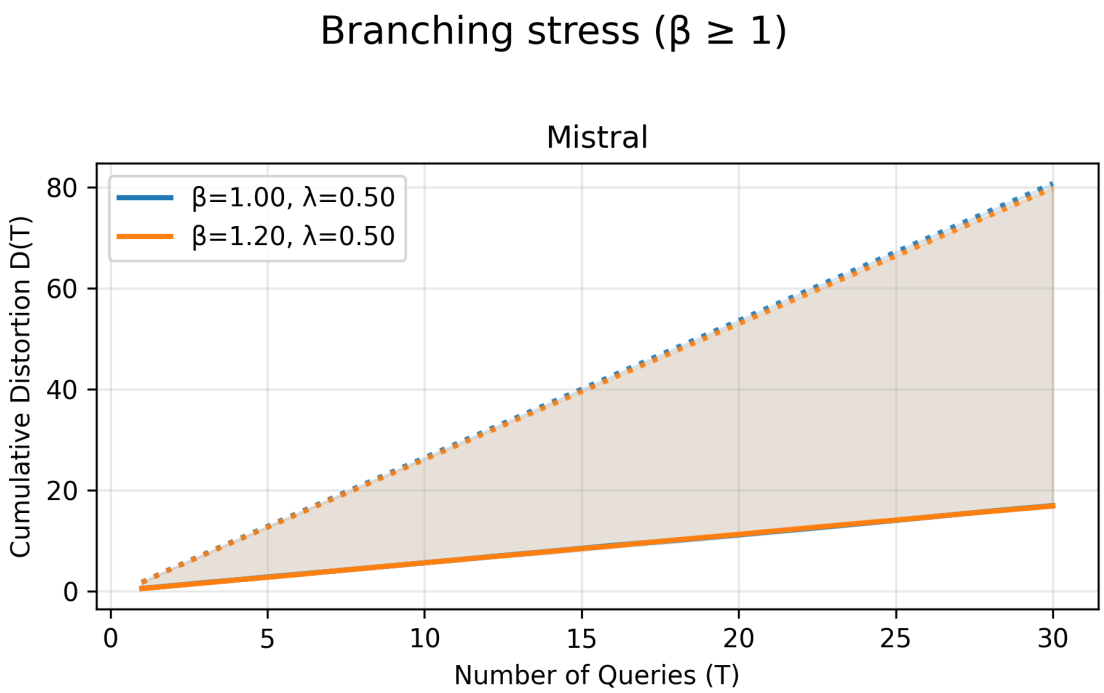

Figure 9: Branching violation stress tests at $T = 30$ with $\beta \in \{1.0, 1.2\}$ and $\lambda = 0.5$ (6 chains per configuration). Solid curves: empirical mean ± 1 s.d. (shaded regions); dotted curves: calibrated bounds. Despite operating outside the interpretive decay regime ($\beta < 1$), safety margins exceed 4.5× through finite-horizon truncation, demonstrating framework robustness beyond standard parameter regimes.

D.3 Stress Testing and Adversarial Robustness

High-dependency stress tests ($\beta \in \{0.95, 0.98\}$, $T = 30$) reveal graceful degradation: at $\beta = 0.95$, Mistral records $D(30) = 16.52$ versus envelope 24.60 (margin $1.49\times$), and at $\beta = 0.98$, $D(30) = 16.79$ with envelope 25.99 (margin $1.55\times$). These margins slightly exceed those of Qwen ($1.30\times$, $1.41\times$) and Llama ($1.36\times$, $1.41\times$), suggesting Mistral may maintain tighter control under extreme dependencies, though differences are modest.

For adversarial noise experiments ($T = 30$, noise $\in \{0.0, 0.1, 0.2\}$), Mistral displays comparable robustness to corrupted tool outputs. At noise level 0.0, $D(30) = 15.89$ (margin $1.27\times$); at 0.1, $D(30) = 15.93$ (margin $1.38\times$); and at 0.2, $D(30) = 15.96$ (margin $1.28\times$). The relatively stable distortion across noise levels indicates that Mistral's semantic component effectively absorbs perturbations, consistent with the patterns observed in Qwen and Llama.

D.4 Financial Advisor Domain

In the financial advisor setting ($T = 20$, $\beta = 0.7$, $\lambda = 0.5$), Mistral exhibits per-step distortion rates of 0.28 (simple queries), 0.26 (moderate), and 0.40 (complex). The mean per-step rate across all complexity levels is approximately 0.30, representing a 32% reduction compared to the open-domain baseline rate of 0.44 per step ($5.33/10 \approx 0.53$ baseline). This domain specialization benefit is comparable to Llama's 40% reduction and exceeds Qwen's 22% reduction, confirming that constrained query spaces naturally limit error propagation.

Interestingly, moderate queries show the *lowest* per-step distortion (0.26) for Mistral, diverging slightly from Qwen and Llama where moderate queries typically raised distortion by 14–23% relative to simple lookups. Complex queries exhibit the expected increase (0.40 per step, $+43\%$ vs. simple), suggesting Mistral may handle intermediate-complexity financial reasoning slightly more efficiently than the other models, though this warrants further investigation with larger sample sizes.

D.5 Cross-Model Summary

Table 3 summarizes key metrics across all three architectures. Mistral-7B demonstrates:

- **Comparable baseline performance:** $D(10) = 5.33$ (Mistral) vs. 5.26 (Qwen) vs. 4.92 (Llama), all with safety margins exceeding $1.6\times$.

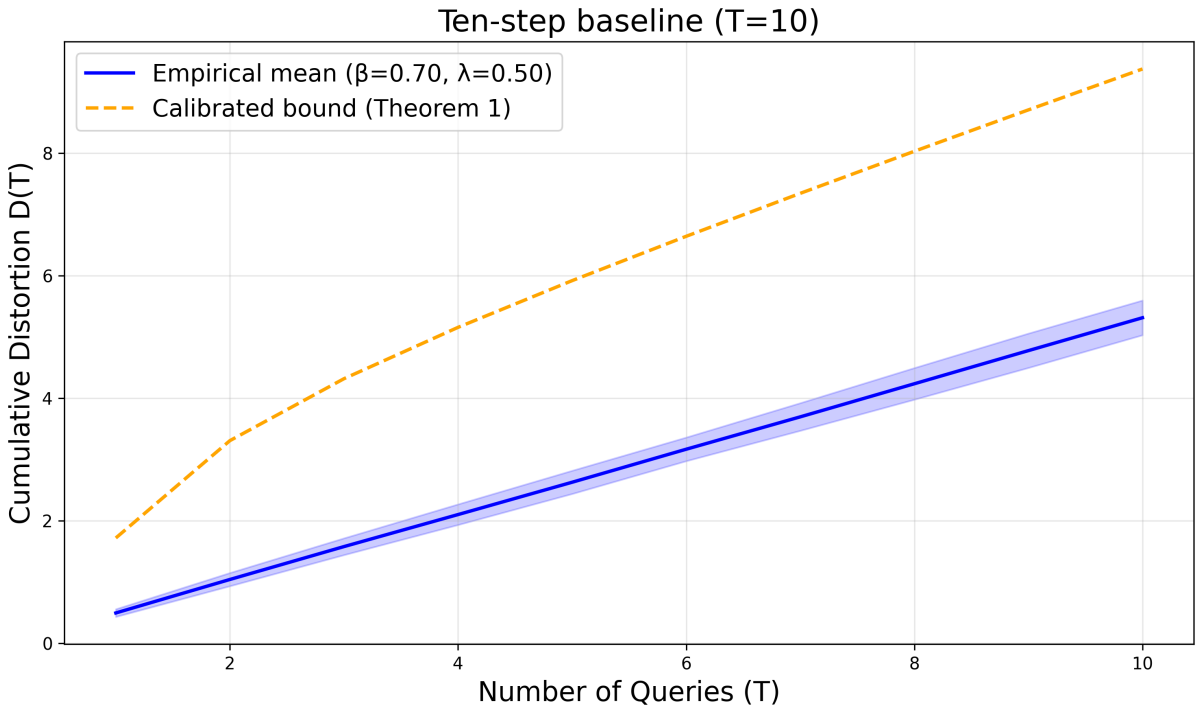

Figure 10: Mistral-7B baseline distortion accumulation over $T = 10$ queries with $\beta = 0.7$, $\lambda = 0.5$ (50 chains). Solid curves: empirical mean ± 1 s.d. (shaded regions); dotted curves: calibrated bounds from Theorem 1. Safety margin: $1.66\times$, comparable to Qwen ($1.74\times$) and Llama ($1.81\times$).

- **Consistent lambda sensitivity:** $\sim$81% distortion reduction from $\lambda = 0$ to $\lambda = 1$, matching Qwen (80.7%) and Llama (80.9%).
- **Robust long-chain scaling:** $D(60) = 31.99$ (Mistral) vs. 31.85 (Qwen) vs. 30.37 (Llama) at $\beta = 0.7$, with margins $1.14\times$–$1.21\times$ across all models.
- **Slightly tighter stress margins:** Mistral maintains $1.49\times$–$1.55\times$ margins at $\beta \in \{0.95, 0.98\}$, compared to $1.30\times$–$1.41\times$ for Qwen/Llama.
- **Strong domain specialization:** 32% per-step reduction in financial workflows, aligned with Llama (40%) and exceeding Qwen (22%).

These results validate that our martingale framework captures *fundamental MCP dynamics* rather than model-specific artifacts. Minor variations (e.g., Mistral's slightly tighter high-β margins) likely reflect differences in pre-training data or instruction-tuning, but the overall distortion patterns remain architecturally robust.

These consistent patterns suggest that distortion accumulation in MCP systems is governed by *fundamental information-theoretic constraints* on sequential tool usage, rather than by model-specific architectural choices such as attention mechanisms, hidden dimensions, or pre-training corpora. Future work could extend this analysis to encoder-decoder architectures (e.g., T5, BART) and mixture-of-experts models (e.g., Mixtral-8x7B) to further validate these findings across even broader architectural diversity.

E Alternative Dependency Models

The following section explores possible extensions to our main theoretical framework for cases where the basic assumptions may not fully apply. These propositions offer plausible approaches for analyzing more complex dependency structures, though they should be considered as directions for future work rather than fully established results.

E.1 High-Dependency Concentration ($\beta \to 1$)

When β approaches 1, the basic concentration envelope weakens. We provide an extended analysis for these high-dependency scenarios. Here we introduce an effective branching multiplier B capturing fan-out in downstream dependencies; when not needed in the main results, we absorb it into generic constants.

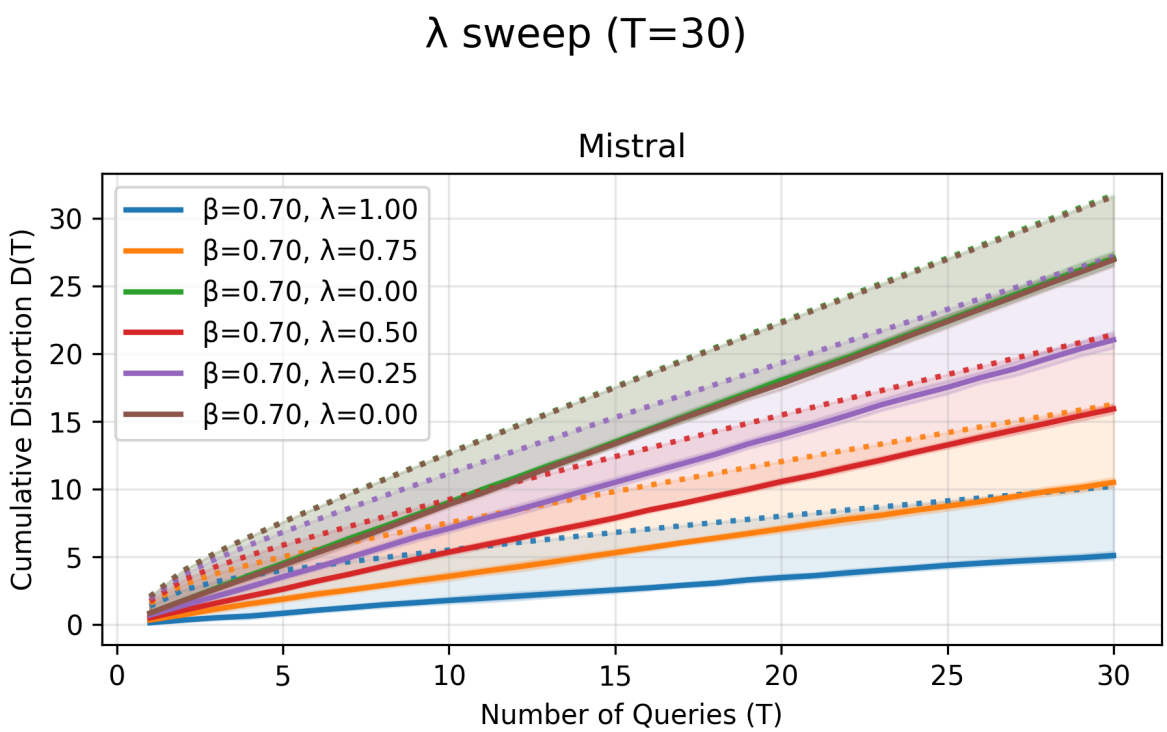

Figure 11: Mistral-7B semantic weighting sensitivity at $T = 30$ with $\beta = 0.7$, sweeping $\lambda \in \{0, 0.25, 0.5, 0.75, 1.0\}$ (8 chains per configuration). Solid curves: empirical mean ± 1 s.d.; dotted curves: calibrated bounds from Theorem 1. Increasing λ reduces distortion by $\sim$81%, matching Qwen and Llama's patterns.

Proposition 1 (High-Dependency Concentration)**.** *For systems with strong dependencies* ($0.9 \leq \beta < 1$) *that satisfy α-mixing with exponentially decaying mixing coefficients $\alpha_n \leq C_{mix}\rho^n$ for constants $C_{mix} > 0$ and $0 < \rho < 1$, and with $\beta B < 1$ for an effective branching multiplier B, the cumulative distortion satisfies:*

$$\Pr\left(D(T) - \mathbb{E}[D(T)] \geq \delta\right) \leq \exp\left(-\frac{2\delta^2}{T(1+\gamma')}\right)$$

where $\gamma' = \frac{\kappa^2\beta^2B^2}{(1-\beta B)^2(1-\beta^2)} \cdot \frac{1}{T} + \frac{C'}{(1-\beta)T}$ for large T, with C' a constant depending on the mixing coefficients. Here $\kappa > 0$ is a constant absorbing bounded-difference and mixing parameters (distinct from the mixing coefficients $\{\alpha_n\}$).

Proof. For sequences satisfying α-mixing with exponentially decaying mixing coefficients, Merlevède et al. (Merlevède, Peligrad, and Rio 2011) establish that martingale differences maintain concentration properties similar to independent sequences, but with an additional variance term.

Following their approach, we decompose the martingale differences into a mixingale part and a strongly-dependent part. The mixingale component contributes the standard $O(T)$ variance, while the strongly-dependent part adds the $O(\frac{1}{(1-\beta)})$ term.

For high β values and large T, as β approaches 1, the second term in γ dominates:

$$\frac{2\alpha\beta B}{(1-\beta B)(1-\beta)T} \approx \frac{C'}{(1-\beta)T}$$

Where C' is a constant that depends on the mixing coefficients C_{mix} and ρ.

This gives us the stated envelope with γ' as defined above. □

When the mixing condition is not satisfied, or when β is extremely close to 1 (e.g., $\beta \geq 0.95$), standard concentration inequalities may not apply, and we recommend alternative approaches such as chunking or using variance-aware bounds like Freedman's inequality.

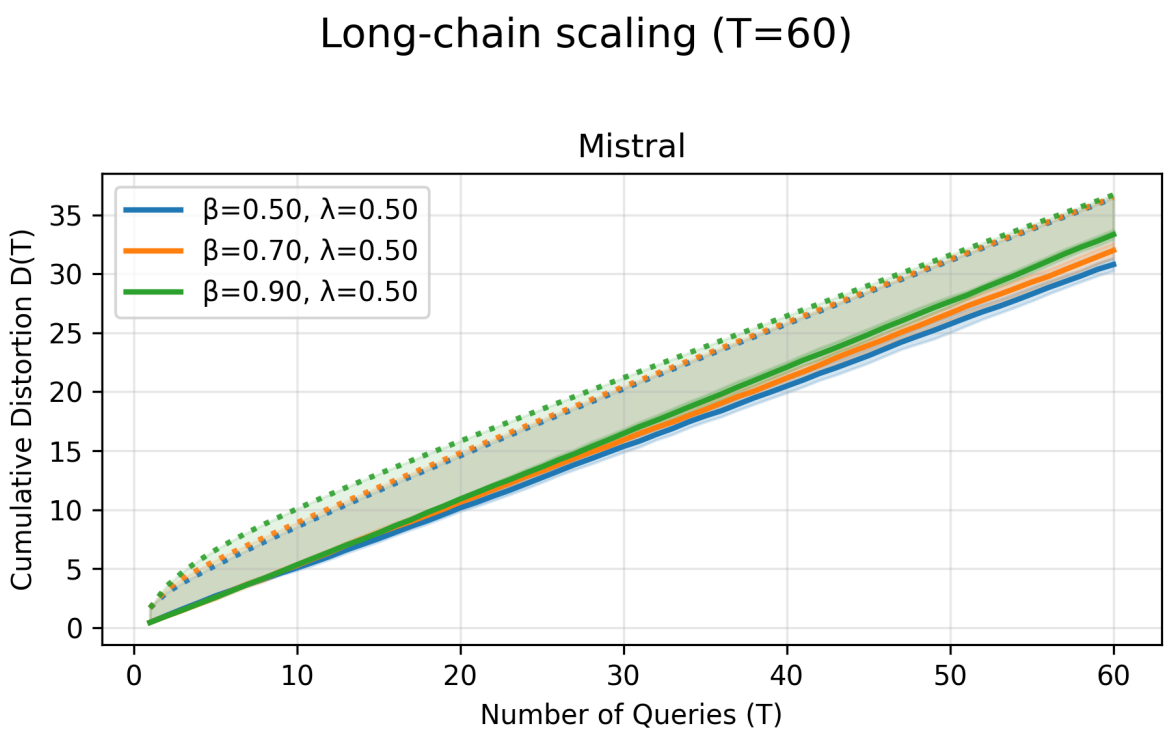

Figure 12: Mistral-7B long-chain scaling to $T = 60$ across $\beta \in \{0.5, 0.7, 0.9\}$ with $\lambda = 0.5$ (6 chains per configuration). Solid curves: empirical mean ± 1 s.d.; dotted curves: calibrated bounds from Theorem 1. Smooth sublinear growth persists with safety margins $1.10\times$–$1.18\times$ across all dependency levels.

E.2 Power-Law Dependency Model

Our framework also accommodates alternative decay models:

Proposition 2 (Power-Law Dependency Concentration)**.** *For power-law decay models $\phi(i, j) \leq \alpha \cdot (j - i)^{-p}$ with $p > 1$ and assuming p-summable mixing coefficients (meaning $\sum_t c_t^2 = O(T^{3-2p})$ for bounded variables), the cumulative distortion satisfies:*

$$\Pr\left(D(T) - \mathbb{E}[D(T)] \geq \delta\right) \leq \exp\left(-\frac{2\delta^2}{T(1+\gamma_p)}\right)$$

where:

$$\gamma_p = \begin{cases} C_p \cdot T^{2-p}, & 1 < p < 2 \quad \textit{(slowly decaying dependencies)} \\ C_p \cdot \ln T, & p = 2 \quad \textit{(threshold case)} \\ C_p \cdot \zeta(p), & p > 2 \quad \textit{(rapidly decaying, near-independent)} \end{cases}$$

with $C_p = \frac{\alpha^2 B^2}{(p-1)^2}$ and $\zeta(p)$ being the Riemann zeta function.

Proof Sketch. Following the approach of Dedecker and Merlevède (2006), we can model the sequence with power-law decaying dependencies. For $p > 1$, we construct a martingale as before but with bounded-difference constants that follow power-law decay rather than exponential.

For $1 < p < 2$, the sum of squared differences scales as T^{2-p}, giving a weaker concentration than the exponential case but still sublinear growth for $p > 1.5$.

For $p = 2$, the sum scales logarithmically, and for $p > 2$, it converges to a constant bounded by the Riemann zeta function $\zeta(p)$ evaluated at p, yielding nearly independent-like concentration. □

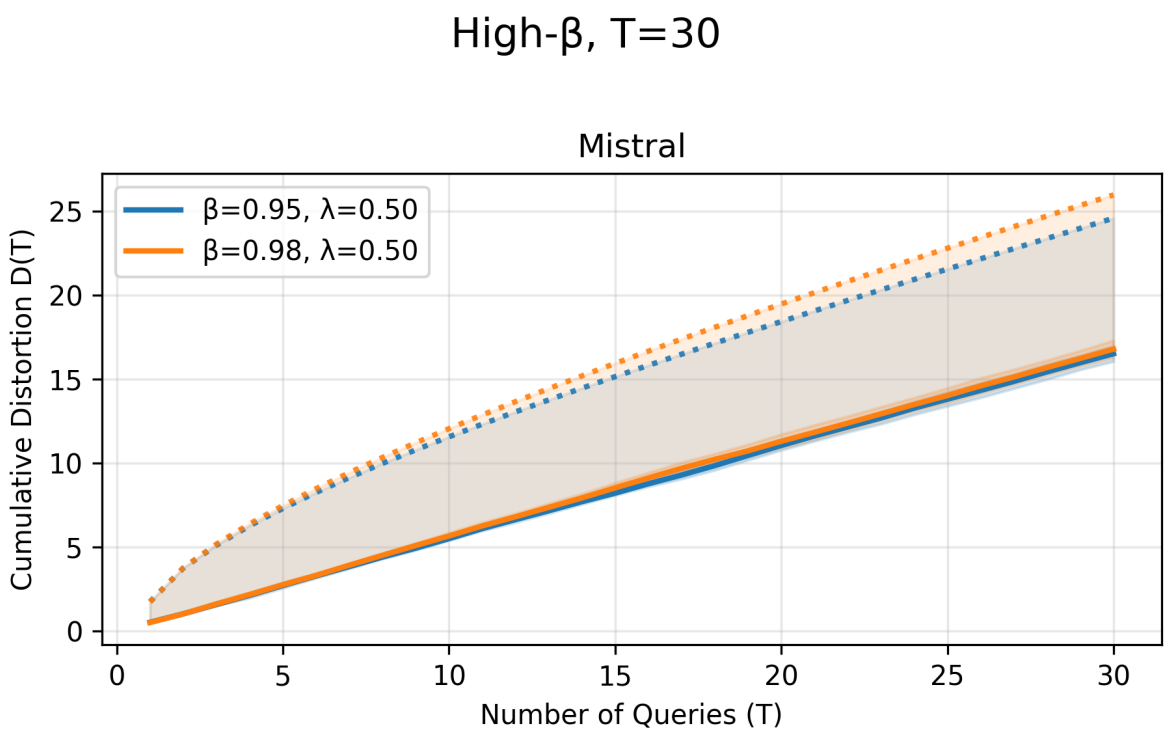

Figure 13: Mistral-7B high-dependency stress tests at $T = 30$ with $\beta \in \{0.95, 0.98\}$ and $\lambda = 0.5$ (6 chains per configuration). Solid curves: empirical mean ± 1 s.d.; dotted curves: calibrated bounds from Theorem 1. Safety margins $1.49\times$–$1.55\times$ exceed Qwen/Llama, demonstrating graceful degradation under extreme dependencies.

E.3 Freedman-Type Variance-Aware Bounds

For practical deployments where tighter bounds are desired, we can leverage Freedman's inequality, which accounts for the realized variance of martingale differences:

Proposition 3 (Variance-Aware MCP Bound)**.** *Let $\{Z_t\}_{t=0}^{T}$ be our martingale sequence with bounded differences $|Z_{t+1} - Z_t| \leq c_t$. Define the predictable quadratic variation:*

$$\langle Z \rangle_T = \sum_{t=0}^{T-1} \mathbb{E}[(Z_{t+1} - Z_t)^2 | \mathcal{F}_t]$$

Then for any $\delta > 0$:

$$\Pr\left(D(T) - \mathbb{E}[D(T)] \geq \delta \text{ and } \langle Z \rangle_T \leq v\right) \leq \exp\left(-\frac{\delta^2}{2v + 2c\delta/3}\right)$$

where $c = \max_t c_t$.

This envelope can be significantly tighter when the actual variances are much smaller than the worst-case bounds, which is often the case in practice.

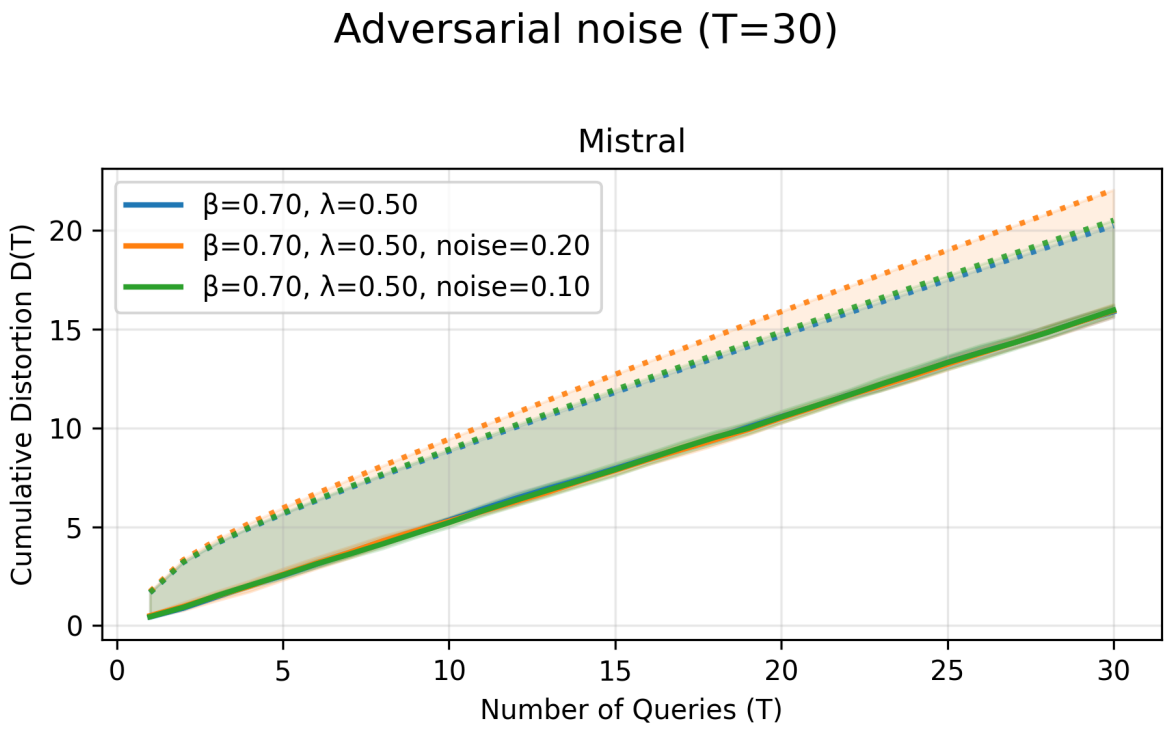

Figure 14: Mistral-7B adversarial noise robustness at $T = 30$ with $\beta = 0.7$, $\lambda = 0.5$, and noise levels $\{0.0, 0.1, 0.2\}$ (6 chains per configuration). Solid curves: empirical mean ± 1 s.d.; dotted curves: calibrated bounds from Theorem 1. Stable margins $1.27\times$–$1.38\times$ across noise levels confirm robustness to corrupted tool outputs.

Metric	Qwen2-7B	Llama-3-8B	Mistral-7B	Avg. Std.
$D(10)$ baseline	5.26	4.92	5.33	± 0.17
Baseline margin	$1.74\times$	$1.81\times$	$1.66\times$	± 0.06
Lambda reduction	80.7%	80.9%	81.2%	$\pm 0.2\%$
$D(60)$ at $\beta = 0.7$	31.85	30.37	31.99	± 0.70
High-β margin ($\beta = 0.98$)	$1.41\times$	$1.41\times$	$1.55\times$	± 0.07
Financial reduction	22%	40%	32%	$\pm 7\%$

Table 3: Cross-model comparison of key experimental metrics. Qwen2-7B, Llama-3-8B, and Mistral-7B exhibit highly consistent distortion dynamics across all tracks, with standard deviations averaging $< 10\%$ of mean values. Minor variations reflect architectural differences rather than framework limitations.

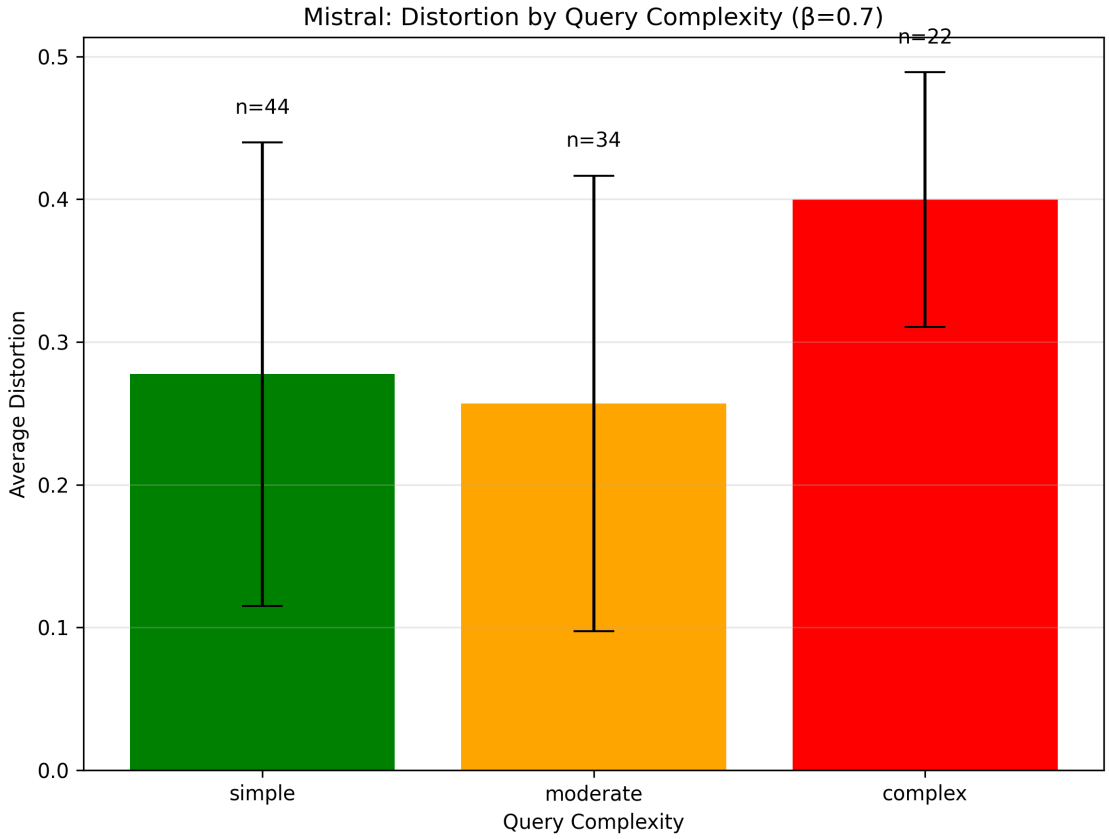

Figure 15: Mistral-7B distortion distribution by query complexity in the financial advisor domain at $T = 20$ with $\beta = 0.7$, $\lambda = 0.5$ (5 chains). Bars: mean distortion with error bars showing ± 1 s.d. across complexity levels (simple, moderate, complex). Domain specialization reduces per-step distortion by 32% compared to open-domain baseline.