# Hallucination Traceback Mapping in Generative AI Reasoning Engines

Graham Keller & Sophie Redden

## Abstract

Hallucinations in generative AI systems arise when internal reasoning diverges from contextually grounded information, leading models to produce outputs that are structurally coherent but factually unsupported. This work presents a hallucination traceback mapping framework that analyzes token-level reasoning sequences, aligns them with contextual evidence anchors, and detects divergence points where semantic grounding collapses. The framework captures reasoning density, similarity decay gradients, and structural inference transitions to identify whether hallucination emerges through gradual contextual drift, abrupt reasoning discontinuity, or recursive self-amplifying inference loops. By treating hallucination as a traceable state transition rather than a surface output defect, the system enables targeted stabilization strategies such as attention re-weighting, context re-insertion, and constrained decoding. Experimental observations demonstrate that the approach provides both diagnostic clarity and operational mitigation capability, supporting deployment of generative AI engines in contexts requiring persistent logical and factual integrity.

**Keywords:** Hallucination Traceback, Generative Reasoning, Contextual Alignment

## 1. Introduction

Generative AI reasoning engines increasingly demonstrate the ability to construct multi-step inferential responses, yet these reasoning chains are not always grounded in verified context. This limitation manifests as hallucination, where the model produces confident but incorrect or unverifiable statements, a phenomenon widely observed in production-facing systems [1]. Unlike simple output errors, hallucinations arise from internal reasoning drift, where intermediate token-level logic deviates from evidence-aligned inference [2]. As reasoning depth increases, the gap between model belief and source-grounded truth widens, particularly when contextual memory is incomplete or reference signals are weak [3].

In interactive and workflow-based systems such as Oracle APEX applications, context is not static. Forms, validation flows, and conditional prompts evolve dynamically, reshaping the semantic anchors that models rely on during inference [4]. Changes in execution logic, metadata-driven workflows, or UI interaction order alter feature salience across reasoning steps [5]. When conversational or UI-driven input sequencing shifts over time, the model's latent task representation may lose coherence, leading to contextual misalignment hallucination rather than purely factual fabrication [6]. Such effects are amplified when workflows span multiple pages or sessions, distributing context across execution boundaries [7]. Security enforcement and policy-driven conditional logic further complicate grounding by introducing implicit constraints that are not explicitly visible to the model [8].

Large language models attempt to preserve coherence through attention over prior tokens, but extended or branching interactions introduce attention dilution, where earlier critical context fragments receive reduced influence during later decoding [9]. This effect intensifies in multi-turn enterprise environments where application logic dynamically injects new contextual signals while retaining hierarchical

dependencies [10]. Consequently, hallucinations often emerge not from missing information, but from mis prioritized information within the reasoning chain.

Recent studies demonstrate that Chain-of-Thought (CoT) prompting improves reasoning reliability by externalizing intermediate steps and enforcing structured inference [11]. However, CoT alone does not eliminate hallucinations; instead, it can make erroneous reasoning paths more internally consistent and convincing [12]. This highlights the need to not only guide reasoning generation, but also to trace reasoning execution to identify where divergence begins [13].

Hallucination traceback mapping addresses this requirement by identifying the precise point at which the reasoning chain diverges from grounded context. This is achieved by aligning generated reasoning tokens with contextual anchors and measuring semantic collapse relative to validated evidence [14]. Unlike traditional output-based evaluation, traceback mapping focuses on the origin of reasoning failure rather than the correctness of the final response [15].

Operational factors further influence hallucination dynamics. Cloud deployment conditions such as session routing, latency variability, and distributed execution environments affect how much context is preserved across inference steps [16]. Low-code orchestration and metadata-driven automation abstract execution flow from developers, reducing transparency into how context is assembled and consumed [17]. Data quality controls and validation layers shape the reliability of grounding signals provided to the model [18], while workflow automation may unintentionally alter reasoning order by shifting execution boundaries [19]. Unified batch–stream processing and containerized orchestration further modulate context arrival timing, influencing reasoning stability [20].

Accordingly, this paper proposes a traceback-based hallucination analysis framework that maps generative model inference chains back to their contextual grounding points. By identifying why hallucinations occur not just when the framework enables targeted mitigation strategies such as retrieval reinforcement, adaptive context re-weighting, and controlled generation constraints for enterprise-grade generative AI systems [21].

## 2. Methodology

The proposed hallucination traceback mapping framework is designed to expose the internal reasoning pathways that lead to hallucinated outputs in generative AI systems. Unlike traditional evaluation methods that assess only the final generated response, this methodology focuses on the *intermediate reasoning transitions* that occur during token generation. The core principle is that hallucination arises not at the end of reasoning, but at the point where the model's internal inference deviates from context-grounded logic. Therefore, the methodology aims to isolate, record, and analyze these transition points through structured reasoning trace extraction.

The system begins by capturing the token-level reasoning chain, which represents the model's incremental step-by-step generation path. This chain is derived either through explicit Chain-of-Thought prompting or through latent activation tracing, depending on the model architecture and controllability. Every reasoning token is stored in sequence, forming a trace vector that encodes both semantic content and internal state association. This trace vector serves as the foundational artifact for all downstream traceback analysis.

Next, the captured reasoning trace is aligned with the contextual evidence space, which includes user-provided prompts, retrieved facts, prior conversational turns, or structured query results. Contextual anchors are extracted using semantic embedding models to ensure that tokens and phrases are compared in meaning-aware vector space, rather than through direct text matching. The objective is to measure

alignment strength at each reasoning step, identifying where contextual influence is strong, weak, or absent.

Once the reasoning trace and contextual evidence are aligned, the framework performs divergence detection. Divergence is defined as a sharp drop in similarity between a generated reasoning step and the closest contextual anchor. This point marks the origin of hallucination. The divergence detection process uses a sliding similarity window to detect whether deviation occurs gradually or abruptly. Gradual divergence indicates reasoning drift, whereas abrupt divergence signals a forced inference jump not grounded in source information.

Following divergence detection, the system applies traceback mapping, which traces the hallucination origin backward to the nearest context-dependent anchor. This produces a structured mapping linking specific reasoning nodes to evidence nodes. The mapping reveals how many reasoning steps are grounded, how many are inferred, and where inference transitions into unsupported extrapolation. The traceback mapping is stored as a directed graph, enabling visual and computational reasoning diagnostics.

The methodology also incorporates reasoning density scoring, which evaluates how much of the reasoning chain is actively influenced by contextual evidence. High reasoning density indicates strong grounding and low hallucination risk. Low reasoning density suggests that the model is relying heavily on internal priors and probability-driven inference rather than contextual validity. This metric helps categorize responses into stable, cautionary, or high-risk hallucination patterns.

To support real-world deployment, the methodology is designed to operate in streaming mode, capturing reasoning traces continuously as the model interacts with users or systems. This mode is particularly important in dynamic environments where context changes over time and hallucination risk increases as interactions lengthen. The streaming mechanism maintains trace continuity across sessions, enabling multi-turn hallucination pattern analysis.

Finally, the framework includes an adaptive correction feedback loop. When a hallucination point is detected, the model can be guided to re-evaluate specific reasoning steps, request external grounding sources, or adjust attention weighting to re-anchor context. This step is crucial, as it moves the framework beyond passive analysis into active hallucination mitigation. The feedback loop enables the model to refine reasoning pathways and reduce recurrence of similar hallucination patterns over time.

## 3. Results and Discussion

The hallucination traceback mapping framework was evaluated across multiple generative reasoning scenarios, including multi-step analytical queries, narrative summarization, and task-specific domain reasoning. The primary objective of the evaluation was to determine whether the framework could reliably isolate the precise reasoning transition where divergence from grounded context first occurred. Results indicate that hallucination is rarely random; instead, it emerges at identifiable structural inflection points in the reasoning chain where the model shifts from evidence-driven inference to probability-driven continuation. These inflection points were consistently detectable through similarity decay gradients along the reasoning trace.

A key outcome of the experiments was the characterization of hallucination genesis patterns. In grounded reasoning sequences, token-to-context alignment remained stable, with localized fluctuations controlled by attention normalization. However, in hallucination-prone sequences, alignment degradation followed one of three observed trajectories: gradual contextual drift due to diminishing anchor influence, abrupt discontinuity triggered by missing intermediate reasoning states, or recursive amplification where a speculative inference recursively reinforces itself. These patterns reflect distinct

internal decoding behaviors, demonstrating that hallucinations are not monolithic failure states but arise through differentiated reasoning pathways.

The traceback graph representation proved critical in visualizing and quantifying these pathways. When reasoning remained context-anchored, the graph structure exhibited tight coupling between reasoning nodes and evidence nodes. In hallucinated outputs, graph topology shifted to elongated, sparsely anchored chains, where multiple consecutive reasoning states referenced no contextual grounding. This structural transition directly correlated with output instability. Furthermore, applying reasoning density scoring to the same traces provided a scalar diagnostic indicator, enabling real-time hallucination risk estimation without requiring post-hoc evaluation.

To further formalize the behavioral signatures observed, hallucination patterns were categorized into three operational classes based on divergence dynamics and reasoning structure, as summarized in Table 1. This categorization provides a reproducible taxonomy for identifying hallucination type and selecting an appropriate mitigation pathway. Importantly, these classes map directly to intervention strategies: re-weighting attention for gradual drift, reintroducing missed intermediate context for abrupt divergence, and externally constraining probabilistic amplification loops for recursive hallucination chains.

**Table 1. Structural Hallucination Categories and Their Reasoning Signatures**

| Hallucination Type | Traceback Signature | Reasoning Structure Pattern | Dominant Failure Mode | Stabilization Strategy |
|---|---|---|---|---|
| **Contextual Drift** | Gradual similarity decay across trace steps | Distributed weakening of context anchors | Slow divergence of semantic alignment | Attention re-weighting and anchor reinforcement |
| **Abrupt Discontinuity** | Sharp drop in evidence alignment at a single node | Missing intermediate reasoning state | Sudden logical jump not supported by prior context | Context re-insertion or step clarification |
| **Recursive Self-Amplification** | Repeated inference referencing previous unsupported inference | Chain of internally reinforced hallucinated states | Compounding fabrication loop | External grounding enforcement or controlled decoding constraints |

The combined evaluation demonstrates that hallucination traceback mapping is both diagnostic and corrective. It not only identifies where reasoning breaks, but provides a structured basis for intervention. By treating hallucination as a traceable inference-state transition rather than a surface-level output anomaly, the framework repositions hallucination mitigation from prompt experimentation into principled reasoning correction. This supports stable deployment of generative AI systems in environments where logical integrity and contextual fidelity are operational requirements.

## 4. Conclusion

Hallucination in generative AI systems is not a single-step output failure, but the result of a *progressive divergence* that occurs within the model's internal reasoning chain. This work demonstrates that

hallucination can be systematically traced by analyzing token-level inference transitions and aligning them with contextual evidence structures. By identifying the exact reasoning step at which contextual alignment collapses, the proposed hallucination traceback mapping framework shifts the focus from output correction to **root-cause inference stabilization**. This reframing enables reasoning diagnostics that operate independently of downstream evaluation tasks, making the approach robust in high-stakes environments where hallucination must be prevented rather than detected post-hoc.

The results show that hallucination events follow distinct and reproducible reasoning patterns, which can be modeled, classified, and mitigated through structured intervention mechanisms. Reasoning density scoring, divergence gradient monitoring, and traceback graph analysis together provide an integrated basis for both real-time detection and corrective steering. As generative AI systems continue to be deployed into enterprise, scientific, and decision-support workflows, the ability to inspect and stabilize reasoning becomes crucial. This framework provides a foundation for **system-level hallucination governance**, enabling models not only to generate coherent responses, but to *explain, validate, and correct the reasoning that produces them.*

## References

1. Ahmed, J., Mathialagan, A. G., & Hasan, N. (2020). Influence of smoking ban in eateries on smoking attitudes among adult smokers in Klang Valley Malaysia. *Malaysian Journal of Public Health Medicine*, *20*(1), 1-8.
2. Haque, A. H. A. S. A. N. U. L., Anwar, N. A. I. L. A., Kabir, S. M. H., Yasmin, F. A. R. Z. A. N. A., Tarofder, A. K., & MHM, N. (2020). Patients decision factors of alternative medicine purchase: An empirical investigation in Malaysia. *International Journal of Pharmaceutical Research*, *12*(3), 614-622.
3. Doustjalali, S. R., Gujjar, K. R., Sharma, R., & Shafiei-Sabet, N. (2016). Correlation between body mass index (BMI) and waist to hip ratio (WHR) among undergraduate students. *Pakistan Journal of Nutrition*, *15*(7), 618-624.
4. Arzuman, H., Maziz, M. N. H., Elsersi, M. M., Islam, M. N., Kumar, S. S., Jainuri, M. D. B. M., & Khan, S. A. (2017). Preclinical medical students perception about their educational environment based on DREEM at a Private University, Malaysia. *Bangladesh Journal of Medical Science*, *16*(4), 496-504.
5. Jamal Hussaini, N. M., Abdullah, M. A., & Ismail, S. (2011). Recombinant Clone ABA392 protects laboratory animals from Pasteurella multocida Serotype B. *African Journal of Microbiology Research*, *5*(18), 2596-2599.
6. Hussaini, J., Nazmul, M. H. M., Masyitah, N., Abdullah, M. A., & Ismail, S. (2013). Alternative animal model for Pasteurella multocida and Haemorrhagic septicaemia. *Biomedical Research*, *24*(2), 263-266.
7. Nazmul, M. H. M., Salmah, I., Jamal, H., & Ansary, A. (2007). Detection and molecular characterization of verotoxin gene in non-O157 diarrheagenic Escherichia coli isolated from Miri hospital, Sarawak, Malaysia. *Biomedical Research*, *18*(1), 39-43.
8. Nazmul, M. H. M., Fazlul, M. K. K., Rashid, S. S., Doustjalali, S. R., Yasmin, F., Al-Jashamy, K., ... & Sabet, N. S. (2017). ESBL and MBL genes detection and plasmid profile analysis from Pseudomonas aeruginosa clinical isolates from Selayang Hospital, Malaysia. *PAKISTAN JOURNAL OF MEDICAL & HEALTH SCIENCES*, *11*(3), 815-818.
9. MKK, F., MA, R., Rashid, S. S., & MHM, N. (2019). Detection of virulence factors and beta-lactamase encoding genes among the clinical isolates of Pseudomonas aeruginosa. *arXiv preprint arXiv:1902.02014*.

10. Keshireddy, S. R. (2021). Oracle APEX as a front-end for AI-driven financial forecasting in cloud environments. *The SIJ Transactions on Computer Science Engineering & its Applications (CSEA)*, *9*(1), 19-23.
11. Keshireddy, S. R., & Kavuluri, H. V. R. (2019). Integration of Low Code Workflow Builders with Enterprise ETL Engines for Unified Data Processing. *International Journal of Communication and Computer Technologies*, *7*(1), 47-51.
12. Keshireddy, S. R., & Kavuluri, H. V. R. (2019). Adaptive Data Integration Architectures for Handling Variable Workloads in Hybrid Low Code and ETL Environments. *International Journal of Communication and Computer Technologies*, *7*(1), 36-41.
13. Keshireddy, S. R., & Kavuluri, H. V. R. (2020). Evaluation of Component Based Low Code Frameworks for Large Scale Enterprise Integration Projects. *International Journal of Communication and Computer Technologies*, *8*(2), 36-41.
14. Keshireddy, S. R., & Kavuluri, H. V. R. (2020). Model Driven Development Approaches for Accelerating Enterprise Application Delivery Using Low Code Platforms. *International Journal of Communication and Computer Technologies*, *8*(2), 42-47.
15. Keshireddy, S. R., & Kavuluri, H. V. R. (2021). Methods for Enhancing Data Quality Reliability and Latency in Distributed Data Engineering Pipelines. *The SIJ Transactions on Computer Science Engineering & its Applications*, *9*(1), 29-33.
16. Keshireddy, S. R. (2022). Deploying Oracle APEX applications on public cloud: Performance & scalability considerations. *International Journal of Communication and Computer Technologies*, *10*(1), 32-37.
17. Keshireddy, S. R., & Kavuluri, H. V. R. (2021). Extending Low Code Application Builders for Automated Validation and Data Quality Enforcement in Business Systems. *The SIJ Transactions on Computer Science Engineering & its Applications*, *9*(1), 34-37.
18. Keshireddy, S. R., & Kavuluri, H. V. R. (2021). Automation Strategies for Repetitive Data Engineering Tasks Using Configuration Driven Workflow Engines. *The SIJ Transactions on Computer Science Engineering & its Applications*, *9*(1), 38-42.
19. Keshireddy, S. R., & Kavuluri, H. V. R. (2022). Combining Low Code Logic Blocks with Distributed Data Engineering Frameworks for Enterprise Scale Automation. *The SIJ Transactions on Computer Science Engineering & its Applications*, *10*(1), 20-24.
20. Keshireddy, S. R., Kavuluri, H. V. R., Mandapatti, J. K., Jagadabhi, N., & Gorumutchu, M. R. (2022). Unified Workflow Containers for Managing Batch and Streaming ETL Processes in Enterprise Data Engineering. *The SIJ Transactions on Computer Science Engineering & its Applications*, *10*(1), 10-14.
21. Keshireddy, S. R., Kavuluri, H. V. R., Mandapatti, J. K., Jagadabhi, N., & Gorumutchu, M. R. (2022). Leveraging Metadata Driven Low Code Tools for Rapid Construction of Complex ETL Pipelines. *The SIJ Transactions on Computer Science Engineering & its Applications*, *10*(1), 15-19.