

Black-box Reasoning Path Planning Protocol: A Methodology for Reasoning Structure Integrity Auditing

Yiwei Liu

Independent Researcher

04092@qztc.edu.cn

Abstract

Aiming at the problems of insufficient interpretability and difficult verification of reasoning structures in black-box reasoning systems, this paper proposes a Zetetic Abductive Audit Engine (ZAAE)—a black-box reasoning path planning protocol for auditing reasoning structure integrity. By defining standardized forward reasoning and backward completion interfaces, the protocol constructs a diverse and redundant set of reasoning paths, and combines stripping tests to realize quantitative auditing of reasoning structure integrity. A relevance-diversity balance mechanism is introduced to ensure the correlation between paths and conclusions while improving the diversity of path combinations, providing a path-level auditing basis for subsequent cross-validation. Through core indicators such as Premise Compliance Rate (PCR) and breakpoint sets, multi-dimensional diagnosis of black-box reasoning behavior from macro to micro is realized. Case analysis shows that the protocol can successfully locate structural breakpoints in reasoning paths, verifying its logical feasibility and effectiveness as a reasoning structure auditing tool. The diversity penalty mechanism helps improve path coverage, providing reliable support for the credibility evaluation of black-box reasoning results.

1 Introduction

Black-box reasoning systems such as large language models and deep neural networks are widely used in fields such as natural language processing and intelligent decision-making. However, the “black-box” nature of their internal mechanisms makes the interpretability and structural integrity of the reasoning process difficult to verify, which has become a key bottleneck for their application in high-reliability scenarios.

Current research on black-box reasoning interpretability mostly focuses on the extraction and interpretation of a single reasoning path, which is difficult to fully reflect the reasoning behavior of the system and has poor generality. To solve the above problems, this paper proposes a black-box reasoning path planning protocol that does not rely on the internal mechanism of the model and is based on standardized interfaces. It constructs reasoning paths through the forward reasoning interface, realizes path integrity auditing through the backward completion interface, generates diverse paths combined with the relevance-diversity balance mechanism, and finally realizes a comprehensive evaluation of the reasoning structure through quantitative indicators.

The main contributions of this paper are as follows: (1) Propose a general black-box reasoning path planning framework, which is applicable to black-box reasoning systems with forward reasoning and backward completion interfaces; (2) Design a path construction algorithm with relevance-diversity balance to improve the comprehensiveness of audit results; (3) Define core indicators such as path penetration rate and breakpoint sets to realize quantitative diagnosis of reasoning structure integrity; (4) Verify the effectiveness of the protocol through experiments, providing guidance for the optimization of black-box reasoning systems.

The subsequent content of this paper is arranged as follows: Section 2 introduces related work; Section 3 elaborates on the core content of the protocol; Section 3.4 analyzes the applicability and limitations of the protocol; Section 4 verifies the effectiveness of the protocol through experiments; Section 5 summarizes the full text and prospects for future work.

2 Related work

2.1 Research on black-box reasoning interpretability

Research on black-box reasoning interpretability is divided into two categories: model-internal and model-external methods. Internal methods extract reasoning paths by analyzing model parameters, intermediate features or attention distributions, which can obtain fine-grained information, but have poor generality and are difficult to apply to commercial systems that do not open internal parameters. External methods do not rely on the internal mechanism of the model, and construct reasoning explanations through input-output analysis, which have strong generality, but most of them can only generate a single explanation, making it difficult to evaluate the stability and redundancy of the reasoning structure (Rudin, 2019; Chang et al., 2026).

2.2 Research on reasoning path planning

Reasoning path planning mainly targets structured data such as knowledge graphs, and is divided into search-based and embedding-based methods. Search-based methods find effective paths through knowledge graph search, which can generate complete paths, but have low efficiency in large-scale fact sets and are difficult to process unstructured data. Embedding-based methods find paths through low-dimensional embedding space, which have high efficiency, but the generated paths lack interpretability, making it difficult to verify the integrity of the reasoning structure (Wu et al., 2025).

In summary, existing research has problems such as poor generality and difficulty in evaluating the integrity of reasoning structures. The protocol proposed in this paper effectively solves the above problems through standardized interfaces and diverse path construction, providing a general and efficient method for black-box reasoning auditing.

3 Black-box reasoning path planning protocol

3.1 Problem formalization and protocol interfaces

3.1.1 Core definitions

Let $\mathcal{B}$ be a black-box reasoning system that provides two types of standardized query interfaces; let $\mathcal{F} = \{f_1, f_2, \dots, f_n\}$ be a set of objective facts; let C be the conclusion space, and $c \in C$ be the target conclusion to be evaluated. The goal of the protocol is to construct and execute a set of reasoning paths $P = \{\mathcal{F}_1, \dots, \mathcal{F}_m\}$, where each path in the final output set $\mathcal{P}_{\text{succ}}$ can independently derive c . Quantitative evaluation of the reasoning structure integrity is realized through the auditing of this set.

3.1.2 Protocol interface definition

The black-box reasoning system $\mathcal{B}$ needs to provide the following two types of standardized interfaces: (1) Forward reasoning interface $\mathcal{B}_{\text{fwd}} : 2^{\mathcal{F}} \rightarrow C \cup \{\perp\}$, which receives a premise subset $\mathcal{F}' \subseteq \mathcal{F}$ and outputs the corresponding reasoning conclusion $c' \in C \cup \{\perp\}$; (2) Backward completion interface $\mathcal{B}_{\text{bwd}} : C \times 2^{\mathcal{F}} \rightarrow 2^{\mathcal{F}}$, which receives a target conclusion c and a partial premise subset $\mathcal{F}_{\text{partial}} \subseteq \mathcal{F}$ and outputs a completed premise set $\mathcal{F}_{\text{completed}} \subseteq \mathcal{F}$. C is the conclusion space, containing all valid conclusion instances that the system can output. $\perp$ is a special state, indicating that the system cannot give a valid conclusion or the

input is beyond the reasoning range. This protocol requires the output of $\mathcal{B}_{\text{fwd}}$ to strictly fall into $C \cup \{\perp\}$.

The call constraint of $\mathcal{B}_{\text{bwd}}$ in this protocol is the single-premise stripping test (i.e., $\mathcal{F}_{\text{partial}} = \mathcal{F}_i \setminus \{f_j\}$). Under this specialized call, the expected output semantics are: if completion is successful, output exactly one fact $\mathcal{F}_{\text{completed}} = \{f'\}$ with $\sigma(f', f_j) \geq \tau$; if completion fails, output $\emptyset$. If multiple facts are output, the protocol takes the one with the highest semantic matching degree.

The type signature $2^{\mathcal{F}}$ is retained for compatibility, but correct operation only depends on single-premise completion semantics. If $f' \notin \mathcal{F}$ is returned, a normalized mapping π maps it to the closest fact in $\mathcal{F}$; if $\max \sigma(f', f) < \tau$, it is treated as failure. This ensures consistent statistics.

An effective output is $\mathcal{B}_{\text{fwd}}(\mathcal{F}') \in C$; $\perp$ is invalid and counted separately.

3.1.3 Interface call constraints and input/output

The protocol only uses single-premise stripping: $\mathcal{F}_{\text{partial}} = \mathcal{F}_i \setminus f_j$. Successful completion requires $|\mathcal{F}_{\text{completed}}| = 1$ and $\sigma \geq \tau$.

Input: $\mathcal{F}, c$, parameters $(M, \tau, \theta, \lambda, K, T)$. Output: six-tuple $\langle c, \text{PCR}, \text{FPR}, \mathcal{W}, \mathcal{P}_{\text{succ}}, \Theta \rangle$. $\Theta = (M, \tau, \theta, \lambda, K, T)$. If no valid path, PCR, FPR, and $\mathcal{W}$ are undefined.

θ and τ are calibrated on a set of $N = 50$ manually labeled triples using all-MiniLM-L6-v2, yielding $\theta = 0.95$, $\tau = 0.75$.

3.2 Path construction protocol

3.2.1 Relevance measurement and diversity measurement

Define inter-fact relevance $\sigma : \mathcal{F} \times \mathcal{F} \rightarrow [0, 1]$ via semantic embedding cosine similarity; fact-conclusion relevance $\sigma_c(f, c) \in [0, 1]$; inter-path diversity:

$$\delta(\mathcal{F}_i, \mathcal{F}_j) = 1 - \left(\frac{1}{|\mathcal{F}_i \cup \mathcal{F}_j|} \right) \sum_{f \in \mathcal{F}_i \cap \mathcal{F}_j} \max_{f' \in \mathcal{F}_i \cup \mathcal{F}_j} \sigma(f, f')$$

Higher δ means greater diversity. δ is valid for any two fact sets, including intermediate states. $T = 3$ is the default consecutive failure limit.

3.2.2 Path construction algorithm

The algorithm generates diverse reasoning paths that each independently derive c : 1. Initialize $P = \emptyset$, $\mathcal{F}_{\text{pool}} = \mathcal{F}$, $\text{failed_attempts} = 0$. 2. Main loop until M paths or no valid seeds: (a) Select top- K seeds by σ_c , pick the one yielding the longest valid path via simulated expansion (paths must have length ≥ 3); (b) Initialize $\mathcal{F}_i$ with the seed; (c) Greedy expansion: score candidate f_j as $\text{score}(f_j) = \lambda \cdot \max_{f \in \mathcal{F}_i} \sigma(f_j, f) - (1 - \lambda) \cdot \max_{\mathcal{F}_h \in P} \delta(\mathcal{F}_i \cup \{f_j\}, \mathcal{F}_h)$; add if $\mathcal{B}_{\text{fwd}}(\mathcal{F}_i \cup \{f_j\})$ is semantically equivalent to c ($\sigma_c \geq \theta = 0.95$); otherwise increment failure counter; stop if T consecutive failures; (d) Verify final path; if passes, add to P ; (e) Restore fact pool. 3. Output P . Default $\lambda = 0.7$ maximizes harmonic mean of PCR and FPR on development set. Query complexity: $O(M \times K \times |\mathcal{F}|^2 \times L)$; with $K = 5, L \leq 10$, approximately linear.

3.3 Path execution, auditing and aggregation

3.3.1 Path execution and integrity detection

1. Forward filtering: For each $\mathcal{F}_i \in P$, call $\mathcal{B}_{\text{fwd}}$. If $\perp$, discard. If $c' \in C$ with $\sigma_c(c', c) \geq \theta$, retain in P_{passed} . If $m_{\text{passed}} = 0$, terminate. 2. Integrity detection: For each f_j in each valid

path, perform stripping test: $\mathcal{F}_{\text{partial}} = \mathcal{F}_i \setminus f_j$, call $\mathcal{B}_{\text{bwd}}(c, \mathcal{F}_{\text{partial}})$. If returned $f' \notin \mathcal{F}$, apply π first. If $\mathcal{F}_{\text{completed}} = \emptyset$, fail. Otherwise, compute $\text{match} = \max_{f' \in \mathcal{F}_{\text{completed}}} \sigma(f_j, f')$; pass if $\geq \tau$, else mark breakpoint. Redundant facts ($\sigma(f_a, f_b) \geq \tau$) may cause passing via substitution.

3.3.2 Core evaluation indicators

1. Premise Compliance Rate (PCR):

$$\text{PCR} = \begin{cases} \text{undefined}, & m_{\text{passed}} = 0 \\ \frac{\sum_{i=1}^{m_{\text{passed}}} \sum_{f_j \in \mathcal{F}_i} I[\text{Passed integrity detection}]}{\sum_{i=1}^{m_{\text{passed}}} |\mathcal{F}_i|}, & m_{\text{passed}} > 0 \end{cases}$$

Note: PCR quantifies consistency between forward and backward reasoning, not logical validity. High PCR indicates mutual recoverability of premises given the conclusion.

2. Forward Pass Rate (FPR): $\text{FPR} = |P_{\text{passed}}| / |P|$. If $\text{FPR} < 0.3$, diversity penalty is excessive; suggest reducing λ .

3. Breakpoint Set $\mathcal{W}$: $\mathcal{W}_{\text{strong}} = \{f | \exists \mathcal{F}_i \in P_{\text{passed}}, f \in \mathcal{F}_i \text{ failed}\}$; $\mathcal{W}_{\text{weak}} = \{f | \text{failure rate} > 0.5\}$. Default output $\mathcal{W}_{\text{weak}}$; if $|P_{\text{passed}}| \leq 3$, fallback to $\mathcal{W}_{\text{strong}}$ with warning.

4. Fully Successful Path Set $\mathcal{P}_{\text{succ}}$: paths where all premises pass integrity detection.

3.3.3 Protocol output specification

Output six-tuple $\langle c, \text{PCR}, \text{FPR}, \mathcal{W}, \mathcal{P}_{\text{succ}}, \Theta \rangle$.

3.4 Protocol applicability and limitations

Applicable to black-box systems with forward and backward interfaces (e.g., LLMs via prompting). Limitations: (1) PCR and $\mathcal{W}$ are sample-based statistics; (2) Does not verify factual truth; (3) Homologous verification risk—agreement may reflect internal biases rather than objective soundness; diversity and cross-validation partially mitigate; (3b) For high-stakes use, deploy with two independent systems; (4) Decisions should consider PCR, FPR, $\mathcal{W}$, and Θ jointly; (5) Does not cover single- or double-premise reasoning (requires length ≥ 3); (6) P_{passed} size depends on λ and system consistency; if FPR too low, statistical validity is limited (Bechler-Speicher et al., 2024).

4 Case analysis: demonstration of protocol operation logic

This section demonstrates ZAAE’s complete logical chain via PoC-level case analysis (Zhao et al., 2025). All cases use artificially constructed facts and behaviors; no real model calls.

4.1 Case design

Four scenarios are selected:

- Case 1: Simple causal reasoning — tests basic construction, filtering, stripping.
- Case 2: Multi-hop with redundant facts — tests interference filtering and diversity.
- Case 3: Multi-path reasoning — tests diversity balance and multi-path cross-validation.
- Case 4: Structurally complete reasoning — tests ability to certify complete structures.

Default parameters: $M = 5, \tau = 0.75, \theta = 0.95, \lambda = 0.7, K = 3, T = 3$.

4.2 Case analysis

4.2.1 Case 1: simple causal reasoning

- $\mathcal{F}$: F1 (water boiling point 100°C under standard pressure), F2 (heating raises temperature above 100°C), F3 (standard pressure is default).
- c : Under standard pressure, water boils when heated.
- Path construction: F1 selected as seed; greedy expansion adds F2 then F3; final path $\{F1, F2, F3\}$.
- Stripping test: F1 fails, F2 fails, F3 passes. $\text{PCR} = 1/3 \approx 0.33$. $\mathcal{W} = \{F1, F2\}$. $\mathcal{P}_{\text{succ}} = \emptyset$.
- Interpretation: Protocol identifies core premises F1, F2 as breakpoints. The forward-backward asymmetry is captured by stripping test.

4.2.2 Case 2: multi-hop with redundant facts

- $\mathcal{F}$: F1 (rain forecast), F2 (umbrella blocks rain), F3 (need to walk), F4 (temperature 25°C), F5 (spare umbrella at home), F6 (wind force 3) (redundant).
- c : Need to bring umbrella today.
- Path construction: F1 selected as seed; adds F2, F3, F5; filters F4, F6. Final path $\{F1, F2, F3, F5\}$.
- Stripping test: F1 fails, F2 fails, F3 passes, F5 passes. $\text{PCR} = 2/4 = 0.50$. $\mathcal{W} = \{F1, F2\}$. $\mathcal{P}_{\text{succ}} = \emptyset$.
- Interpretation: Protocol filters irrelevant facts and identifies core dependencies.

4.2.3 Case 3: multi-path reasoning

- $\mathcal{F}$: F1 (leave 8:00), F2 (commute 40 min), F3 (clock-in 8:30), F4 (subway 99% on-time), F5 (drive 60 min in congestion), F6 (5-min grace).
- c : Xiao Ming will be late.
- Path construction: Path 1: $\{F3, F2, F1\}$; Path 2: $\{F5, F1, F3\}$.
- Stripping test: Path 1: F1 passes, F2 fails, F3 fails. Path 2: F1 passes, F3 fails, F5 fails. $\text{PCR} = 2/6 \approx 0.33$. $\mathcal{W} = \{F2, F3, F5\}$. $\mathcal{P}_{\text{succ}} = \emptyset$.
- Interpretation: Multi-path cross-validation comprehensively identifies breakpoints across different reasoning chains.

4.2.4 Case 4: structurally complete reasoning

- $\mathcal{F}$: F1 (All humans mortal), F2 (Socrates human), F3 (Socrates mortal).
- c : Socrates is mortal.
- Path construction: Single path $\{F1, F2, F3\}$.
- Stripping test: F1 passes, F2 passes, F3 passes. $\text{PCR} = 1.0$. $\mathcal{W} = \emptyset$. $\mathcal{P}_{\text{succ}} = \{\{F1, F2, F3\}\}$.
- Interpretation: Protocol correctly certifies complete reasoning structures, proving it is a bidirectional diagnostic tool.

4.3 Case discussion

The results are summarized in Table 1.

The four cases demonstrate: (1) basic feasibility in simple causal reasoning; (2) effective interference filtering in complex fact pools; (3) multi-path cross-validation for comprehensive

Case	Paths Built	PCR	Breakpoints	$\mathcal{P}_{\text{succ}}$
1	1	0.33	F1, F2	$\emptyset$
2	1	0.50	F1, F2	$\emptyset$
3	2	0.33	F2, F3, F5	$\emptyset$
4	1	1.00	$\emptyset$	$\{F1, F2, F3\}$

Table 1: Summary of case analysis results

breakpoint identification; (4) correct certification of complete structures, confirming ZAAE as a true bidirectional diagnostic tool. Cases 1–3 reflect forward-backward asymmetry captured by stripping tests, while Case 4 demonstrates the symmetric complete case. FPR=1 for all cases, indicating $\lambda = 0.7$ is reasonable in PoC settings.

5 Summary and outlook

This paper proposes ZAAE for auditing black-box reasoning structure integrity via standardized interfaces, relevance-diversity balanced path construction, and quantitative indicators. Case analysis verifies its logical feasibility.

Future work will deploy ZAAE on real LLMs (e.g., Llama 3, Qwen2.5) across reasoning benchmarks with quantitative comparisons against forward-only leave-one-out ablation; explore external fact verification and multi-source cross-validation to address homologous verification; and extend to more black-box systems.

Acknowledgments

The author thanks a former classmate. The story between us was complex enough, unusual enough, and asymmetric enough in its forward and backward unfolding. This led the author to realize that a conclusion can be treated as a preset target, from which the required premises are reconstructed through backward completion, and the stripping test is then used to check whether each premise can be recovered, thereby auditing the integrity of the reasoning structure.

Ethics Statement

All experiments use artificially constructed scenarios and public embedding models, with no real user data or sensitive information.

References

- Maya Bechler-Speicher, Amir Globerson, and Ran Gilad-Bachrach. The interpretable and effective graph neural additive networks. In *International Conference on Machine Learning (ICML)*, 2024. arXiv:2406.01317.
- Shuaichen Chang, Tian Bai, Xin Zhang, Qiang Ma, Qian Liu, Zhou Liao, Yuxuan Miao, and Lili Niu. Unlocking the black box of latent reasoning: An interpretability-guided approach to intervention. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 34019–34032, San Diego, California, United States, 2026. Association for Computational Linguistics.
- Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, 2019.
- Rui Wu, Peng Cai, Jian Mei, Lin Wen, Tao Hu, Xue Yang, Dong Fu, and Botian Shi. KG-TRACES: Enhancing large language models with knowledge graph-constrained trajectory reasoning. *arXiv preprint arXiv:2506.00783*, 2025.

Ruoxi Zhao, Fei Zhao, and Hao Zhang. Correcting on graph: Faithful semantic parsing over knowledge graphs with large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 5364–5376, Vienna, Austria, 2025. Association for Computational Linguistics.