ANCHORWEAVE: PRESERVING CAUSAL CONTINUITY IN BUDGETED LLM TRACE ARCHIVES

Anonymous authors
Paper under double-blind review

ABSTRACT

An archived reasoning trace is useful only if a later system can continue the computation that the trace records. A byte cap makes this difficult: exact retention can protect identifiers, literals, and tool arguments while discarding the intervening state that explains how those facts should be used. We formulate recover-and-continue archival as a representation problem with two distinct obligations: preserve brittle local facts and preserve the ordered transitions that make those facts actionable. We describe AnchorWeave, a record design with exact anchors, representations for intervening intervals, explicit boundary metadata, and an optional exact fallback. A simple indistinguishability result shows that, whenever a continuation target depends on omitted transition state, an exact-only archive that retains the same anchors cannot be sufficient for all traces consistent with those anchors. We then give a reporting contract for evaluating concrete archive implementations: it fixes the success denominator, serialized-byte accounting, fallback treatment, continuation protocol, provenance, comparator specification, and uncertainty calculation. The contract specifies the records needed to test a concrete archive implementation.

1 INTRODUCTION

Reasoning models and tool-using agents generate intermediate reasoning and action traces (Wei et al., 2022; Yao et al., 2023; Shinn et al., 2023; Yang et al., 2024; DeepSeek-AI, 2025; Snell et al., 2024). When such traces are retained as durable operational state, their value is not exhausted by byte identity. A later request must recover enough of the earlier trajectory to choose the next action correctly.

This requirement creates a rate–utility problem specific to cold trace storage. Lossless compressors, content-defined deduplication, and delta coding exploit byte redundancy while preserving replayable records (Collet & Kucherawy, 2021; Xia et al., 2016; Burns & Long, 1998). Under a strict cap, however, an archive can be forced to keep only selected exact spans. This protects local facts but can remove the state transitions that connect a fact to its role in a later continuation.

We call this missing property *causal continuity*: a recovered trace must retain both local invariants and the ordered, task-relevant transitions between them. AnchorWeave is a representation design for this setting. It stores selected anchors exactly, associates each omitted interval with a recoverable representation, records the boundaries and order that bind an interval to its anchors, and permits an exact fallback when a deployment policy judges recovery unsafe. The design is deliberately separated from any particular encoder, corpus, or measured operating point.

This paper makes three contributions:

- a formal recover-and-continue archive contract and a structural condition under which exact-only retention cannot determine the correct continuation;
- an anchor-and-transition record design that makes boundary, ordering, and fallback state explicit; and
- an empirical reporting contract that specifies the evidence required to compare concrete archive implementations fairly.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

2 Related Work

Exact persistent storage. Lossless storage systems exploit redundancy while preserving a byte-identical record (Collet & Kucherawy, 2021; Xia et al., 2016; Burns & Long, 1998). Their storage unit is usually a file or chunk, and their fidelity objective is exact replay. They provide an essential reference point for archival representations, but they do not by themselves specify how a constrained byte budget should preserve state needed for continuation.

Live inference state and compact context. H2O, ScissorHands, and StreamingLLM retain useful KV state during active generation (Zhang et al., 2023; Liu et al., 2023; Xiao et al., 2024). Gist Tokens, AutoCompressor-style encoders, LLMLingua, LongLLMLingua, and Compressive Transformers shorten or encode context for inference (Mu et al., 2023; Chevalier et al., 2023; Jiang et al., 2023; 2024; Rae et al., 2020). A trace archive adds a write/read boundary: a representation must survive serialization and cold recovery, and an evaluation must charge the state required to restore it.

Retrieval and long-term memory. Sparse and dense retrieval, REALM, RAG, RETRO, MemGPT, and LongMem select documents, passages, or memories for future generation (Robertson & Zaragoza, 2009; Karpukhin et al., 2020; Guu et al., 2020; Lewis et al., 2020; Borgeaud et al., 2022; Packer et al., 2023; Wang et al., 2023). These systems motivate a complementary design question: whether an archive should retain local facts, a global summary, retrieved material, or structured transition state. Any empirical comparison among these choices requires comparable budgets, access rules, and configuration records.

3 Recover-and-Continue Archival

3.1 Archive contract

Let a trace be $T = (x_1, \ldots, x_m)$ and let q be a later continuation request. A policy π serializes an archive record $A_\pi(T)$; recovery produces $\widehat{T} = D_\pi(A_\pi(T), q)$; and a fixed continuation procedure produces $C(q, \widehat{T})$. For the primary endpoint, let G be the task-success indicator. A concrete study may additionally report exact output identity or another endpoint, but these quantities are not interchangeable with task success.

For a collection of N prespecified trace/request pairs, the primary absolute-success endpoint is

$$S_{\mathrm{abs}}(\pi) = \frac{1}{N} \sum_{i=1}^{N} G\Big(C(q_i, \widehat{T}_i)\Big).$$

Every prespecified pair belongs in the denominator, including pairs for which a full replay does not succeed. A comparison to full replay may report $S_{\mathrm{abs}}(\pi) - S_{\mathrm{abs}}(\mathrm{full})$, but it must not replace task success with agreement to a failed replay.

The byte quantity relevant to a storage claim is the realized serialized size $B_\pi(T)$ after the stated pipeline. It must include the exact payload, interval representations, boundary and ordering metadata, indices, dictionaries, model-related state when applicable, and every fallback copy. A study may report a ratio only after defining both numerator and denominator on the same byte unit.

3.2 Why exact anchors can be insufficient

An exact-only archive selects a set of retained spans $E(T)$ and discards the remainder. Let $g(q, T)$ denote the correct continuation target for a trace/request pair. The target is *transition-sensitive relative to* E if there are traces T and T' and a request q such that

$$E(T) = E(T') \qquad \text{and} \qquad g(q, T) \neq g(q, T').$$

This condition holds whenever the same stored identifiers or actions must be used differently because an omitted dependency, branch condition, derivation, or tool state differs.

Write: segment a trace; select exact anchors; encode each omitted interval; record interval boundaries, order, and types; serialize any fallback according to the policy.
Read: use an exact fallback when selected; otherwise place anchors at their stored offsets, reconstruct each interval from its representation and boundary context, assemble the recovered trace, and issue the prespecified continuation request.

Figure 1: Conceptual AnchorWeave record lifecycle. The figure specifies information flow rather than a measured implementation or operating point.

Proposition 1. If g is transition-sensitive relative to an exact-only archive E, no deterministic recovery rule that receives only $(q, E(T))$ can return $g(q, T)$ for both witness traces.

Proof. The two witness inputs to the recovery rule are identical because q and $E(T) = E(T')$ are identical. A deterministic rule therefore returns the same value for both inputs. That value cannot equal two distinct targets $g(q, T)$ and $g(q, T')$. □

The proposition is a structural statement, not a measured effect size. It does not imply that every omitted interval matters, nor that every learned reconstruction is reliable. It identifies the information that an exact-only record has erased in the transition-sensitive case.

3.3 AnchorWeave record design

AnchorWeave represents a trace as

$$A_\pi(T) = \big(E, \{z_j\}_{j=1}^{J}, M, F\big),$$

where E contains exact anchor bytes, z_j represents one omitted interval, M records interval boundaries, ordering, and types, and F is an optional exact full-trace fallback. Recovery uses the stored positions and adjacent anchors to bind each z_j to its original gap before continuation. The record is self-contained only when all state needed by its chosen recovery implementation is charged and documented.

The representation separates three roles. Exact anchors preserve brittle local invariants. Interval representations retain information that is unavailable to an anchor-only reader. Metadata prevents a representation from being detached from the gap it is intended to recover. A fallback is not free: if present, it is part of $B_\pi(T)$ and must be counted under the same serialization rule as every other payload.

4 Evaluation Contract for Concrete Implementations

The preceding definitions do not establish empirical dominance of any archive policy. A concrete evaluation can do so only with a complete result record. Table 1 lists the minimum information needed to interpret a comparison without silently changing a denominator, byte unit, or protocol.

Matched budgets. Matching a nominal compression ratio is not sufficient. For each compared policy, the result record must expose unrounded realized bytes and the rule that maps a policy setting to that byte total. If a budget is tuned on a development split, the permitted data and selection rule must be the same kind of information for every learned comparator.

Endpoints and uncertainty. The evaluation must distinguish absolute success, agreement with a full replay, and repeated-recovery stability. Paired comparisons should preserve trace/request identities; uncertainty records must name the resampling unit, the number of independent runs or seeds, and the calculation that produced each displayed interval. Aggregate statistics require the component values and explicit weights.

Cost claims. Storage bytes and operational cost are different quantities. A cost comparison must state a cost model such as

$$C_\pi = c_{\text{write}} W_\pi + c_{\text{read}} R_\pi + c_{\text{recover}} H_\pi,$$

Table 1: Evidence required for an empirical recover-and-continue comparison

Record	Required content
Inputs and provenance	Immutable upstream release, revision or archive identifier for each input and evaluation tool; source-to-example mapping; split and lineage rules.
Endpoint and denominator	Prespecified request construction, continuation decoding, evaluator revision, unit of analysis, inclusion rule, and treatment of full-replay failures.
Storage accounting	Serialization pipeline; byte unit; variable-length aggregation; exact payload, metadata, index, dictionary, fallback, and shared-weight treatment.
Routing and recovery	Anchor-selection rule; calibration data and procedure; threshold-to-action mapping; fallback treatment; interval boundary and overlap rules.
Comparators	Architecture, parameter count, objective, training and search budget, selected configuration, permitted query access, and matching-budget procedure.
Statistics and economics	Trace/run identities; seed aggregation; resampling or interval method; all macro weights; price basis and dates when cost is reported; write, read, recovery, and aggregation terms.

then identify the price basis, time range, workload counts, and aggregation rule used to instantiate it. Without those inputs, a saving or a reread crossover is not an interpretable result.

5 Limitations and Discussion

The structural result applies only to continuation targets that are transition-sensitive relative to the stored anchors. It does not prove that a particular model can reconstruct the missing transition, that a fallback policy is calibrated, or that a representation improves success at a fixed budget. Those are empirical questions. They require the records in Table 1, including verified inputs, complete comparator configurations, and reproducible exports for every reported value.

The framework also leaves design trade-offs open. Dense interval representations may preserve useful state but create storage and recovery overhead. Aggressive anchor retention may reduce ambiguity while leaving too little capacity for transition state. Fallbacks can protect cases in which reconstruction is unsafe, but only an accounting rule that charges them can compare this protection fairly. These are hypotheses to be evaluated under a declared contract, not conclusions supported here by a numerical study.

6 Conclusion

Budgeted trace archives must distinguish exact local facts from the transition state that makes those facts actionable. AnchorWeave makes that distinction explicit through exact anchors, interval representations, boundary metadata, and optional exact fallback. Proposition 1 establishes the limited but useful fact that an exact-only record cannot determine every transition-sensitive continuation. The accompanying evaluation contract specifies what a future empirical comparison must disclose before it can support a claim about storage, recovery, routing, cost, or robustness.

Reproducibility Statement

Sections 3–4 define the archive contract, prove the non-empirical structural claim, and state the fields required for an empirical record. The manuscript reports no numerical experiment, benchmark result, cost estimate, or learned-baseline comparison.

AI Use Statement

Generative AI was used only for manuscript writing, editing, and presentation; the authors reviewed all edits and remain responsible for the final content.

REFERENCES

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W. Rae, Erich Elsen, and Laurent Sifre. Improving language models by retrieving from trillions of tokens. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pp. 2206–2240, 2022. URL https://proceedings.mlr.press/v162/borgeaud22a.html.

Randal C. Burns and Darrell D. E. Long. In-place reconstruction of delta compressed files. In *Proceedings of the ACM Symposium on Principles of Distributed Computing*, pp. 267–275, 1998. doi: 10.1145/277697.277747. URL https://doi.org/10.1145/277697.277747.

Alexis Chevalier, Alexander Wettig, Anirudh Ajith, and Danqi Chen. Adapting language models to compress contexts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 3829–3846, 2023. doi: 10.18653/v1/2023.emnlp-main.232. URL https://aclanthology.org/2023.emnlp-main.232/.

Yann Collet and Murray Kucherawy. Zstandard compression and the application/zstd media type, 2021. URL https://www.rfc-editor.org/info/rfc8878. RFC 8878 (obsoletes RFC 8478).

DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. doi: 10.48550/arXiv.2501.12948. URL https://arxiv.org/abs/2501.12948.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. Retrieval augmented language model pre-training. In *Proceedings of the International Conference on Machine Learning*, pp. 3929–3938, 2020. URL https://proceedings.mlr.press/v119/guu20a.html.

Huiqiang Jiang, Qianhui Wu, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. LLMLingua: Compressing prompts for accelerated inference of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 13358–13376, 2023. doi: 10.18653/v1/2023.emnlp-main.825. URL https://aclanthology.org/2023.emnlp-main.825/.

Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. LongLLMLingua: Accelerating and enhancing LLMs in long context scenarios via prompt compression. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1658–1677, 2024. doi: 10.18653/v1/2024.acl-long.91. URL https://aclanthology.org/2024.acl-long.91/.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 6769–6781, 2020. doi: 10.18653/v1/2020.emnlp-main.550. URL https://aclanthology.org/2020.emnlp-main.550/.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Zichang Liu, Aditya Desai, Fangshuo Liao, Weitao Wang, Victor Xie, Zhaozhuo Xu, Anastasios Kyrillidis, and Anshumali Shrivastava. Scissorhands: Exploiting the persistence of importance hypothesis for LLM KV cache compression at test time. *arXiv preprint arXiv:2305.17118*, 2023. doi: 10.48550/arXiv.2305.17118. URL https://arxiv.org/abs/2305.17118.

Jesse Mu, Xiang Lisa Li, and Noah Goodman. Learning to compress prompts with gist tokens. *Advances in Neural Information Processing Systems*, 36:19327–19352, 2023. doi: 10.52202/075280-0848. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/3d77c6dcc7f143aa2154e7f4d5e22d68-Abstract-Conference.html.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. MemGPT: Towards LLMs as operating systems. *arXiv preprint arXiv:2310.08560*, 2023. doi: 10.48550/arXiv.2310.08560. URL https://arxiv.org/abs/2310.08560.

Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, Chloe Hillier, and Timothy P. Lillicrap. Compressive transformers for long-range sequence modelling. *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=SylKikSYDH.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009. doi: 10.1561/1500000019. URL https://doi.org/10.1561/1500000019.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36:8634–8652, 2023. doi: 10.52202/075280-0377. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1b44b878bb782e6954cd888628510e90-Abstract-Conference.html.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024. doi: 10.48550/arXiv.2408.03314. URL https://arxiv.org/abs/2408.03314.

Weizhi Wang, Li Dong, Hao Cheng, Xiaodong Liu, Xifeng Yan, Jianfeng Gao, and Furu Wei. Augmenting language models with long-term memory. *arXiv preprint arXiv:2306.07174*, 2023. doi: 10.48550/arXiv.2306.07174. URL https://arxiv.org/abs/2306.07174.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022. doi: 10.52202/068431-1800. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html.

Wen Xia, Yukun Zhou, Hong Jiang, Dan Feng, Yu Hua, Yuchong Hu, Yucheng Zhang, and Qing Liu. FastCDC: A fast and efficient content-defined chunking approach for data deduplication. In *USENIX Annual Technical Conference*, pp. 101–114, 2016. URL https://www.usenix.org/conference/atc16/technical-sessions/presentation/xia.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=NG7sS51zVF.

John Yang, Carlos E. Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik R. Narasimhan, and Ofir Press. SWE-agent: Agent-computer interfaces enable automated software engineering. *arXiv preprint arXiv:2405.15793*, 2024. doi: 10.48550/arXiv.2405.15793. URL https://arxiv.org/abs/2405.15793.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=WE_vluYUL-X.

Zhenyu Zhang, Ying Sheng, Tianyi Zhou, Tianlong Chen, Lianmin Zheng, Ruisi Cai, Zhao Song, Yuandong Tian, Christopher Ré, Clark Barrett, Zhangyang Wang, and Beidi Chen. H2O: Heavy-hitter oracle for efficient generative inference of large language models. *Advances in*

Neural Information Processing Systems, 36:34661–34710, 2023. doi: 10.52202/075280-1506. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/6ceefa7b15572587b78ecfcebb2827f8-Abstract-Conference.html.

A Counterexample Schema

Consider two traces with the same retained anchor set E: both contain an exact file path and an exact function name, but one omitted interval establishes that a branch is enabled and the other establishes that it is disabled. A later request asks for the next tool action. The retained anchors are identical, while the correct action differs. This is an instance of transition sensitivity and therefore falls under Proposition 1. The example is a logical schema; it is not a claim about a measured trace, a benchmark, or a deployment.

B Result-Record Checklist

Before reporting a concrete implementation, authors should be able to answer each of the following questions from a preserved record:

- Which immutable upstream release and evaluator revision produced each example, and how were splits isolated from fitting and calibration?
- What bytes were serialized for each policy, including variable-length fields, metadata, fallback copies, indexes, dictionaries, and shared state?
- What request, decoder configuration, evaluator, inclusion rule, and unit of resampling produced each endpoint and interval?
- How was every comparator trained, selected, permitted to access queries, and matched to the reported budget?
- If cost is reported, what are the price basis, dates, workload counts, read/write/recovery terms, and aggregation rule?

C Failure Modes to Test

The record design suggests failure modes that a future evaluation should test directly. Reconstructed transition state can be locally plausible but incompatible with an exact anchor. Boundary metadata can be incomplete or misaligned, causing a valid interval representation to be attached to the wrong gap. A routing policy can be miscalibrated under a source shift, allocating exact fallback to the wrong cases. Finally, a nominal budget match can conceal a mismatch in serialized metadata or shared-state treatment. Reporting these tests requires the same provenance and accounting fields as a primary comparison.