WHEN CAN AGENTS FORGET THEIR REASONING? ICLR FOR LONG HORIZON AGENT CONTEXT COMPRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Long horizon language model agents continually accumulate reasoning history, increasing context length and inference cost even after earlier decisions have been executed and observed. Unlike static Chain of Thought compression, removing historical reasoning can change future actions and the resulting interaction trajectory. We study when such reasoning can be safely forgotten. We propose **Interaction Aware Compression for Long Horizon Reasoning (ICLR)**, a training free online method that ranks reasoning blocks using frozen proxy entropy while preserving actions, tool calls, and observations. Across four domains of long horizon agent tasks, ICLR improves average reward from 0.699 to 0.718, while reducing input, output, and cache read tokens by 25.5%, 14.4%, and 33.3%, respectively. Ablations reveal trajectory amplification, where local reasoning deletion produces nonlinear changes in total computation by altering subsequent interaction. Representation probing, activation patching, and controlled trajectory analyses further suggest that historical reasoning becomes more replaceable once task relevant derived state has been reliably externalized into code, files, tool outputs, or environmental feedback. These results characterize agent reasoning as dynamic working state rather than permanent interaction history.

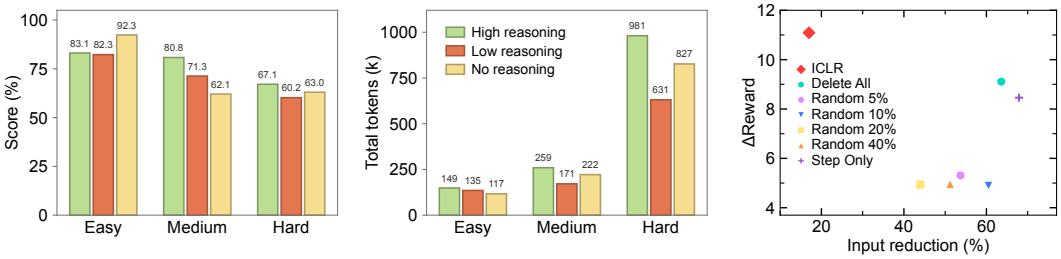


Figure 1: **Reasoning utility and cost are state dependent. (a,b)** Scores and token usage vary across reasoning settings and task difficulty. **(c)** Online interventions show a nonmonotonic reward and efficiency tradeoff. See Appendices J and F.

1 INTRODUCTION

Large language models are increasingly used in complex agent tasks, where reasoning, action, and environmental feedback form a repeated interaction loop (Yao et al., 2022; Shinn et al., 2023; Park et al., 2023). As a task progresses, earlier reasoning is repeatedly included in later requests, increasing context length and inference cost (Kang et al., 2025; Wei et al., 2026; Wu et al., 2026). Meanwhile, new observations continuously change the task state. This raises a central question for agent context management: once reasoning has produced an action and the environment has returned feedback, does that reasoning still need to remain in the active context?

Figure 1 motivates this question from two observations. The most effective reasoning setting varies with task difficulty, and more reasoning does not consistently yield higher scores. Our online interventions also show a nonmonotonic relation among reasoning removal, reward, and input reduction. Together, these observations suggest that the value of reasoning depends on the current state rather than on reasoning quantity alone.

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.

Prior work has shown that explicit Chain of Thought reasoning contains substantial redundancy (Wei et al., 2022; Wang et al., 2022; Li et al., 2026e), and existing methods can shorten intermediate reasoning while preserving answer quality (Li et al., 2026e;a; Qiao et al., 2025; Xia et al., 2025). These settings are largely static because a completed reasoning trace can be modified and then evaluated against the same final answer. Agent reasoning is different. Changing reasoning at step t may alter the next tool call, observation, and subsequent decisions. Reasoning compression for agents therefore intervenes in a closed loop decision process rather than simply modifying text (Li et al., 2026c; Wang et al., 2026; Li et al., 2026b).

Reasoning can move across different forms of state. A plan, constraint, or intermediate conclusion may exist only in the reasoning trace, but become available through tool calls, code, files, or environmental feedback. Context management and memory systems use persistent or external state to reduce dependence on the active context (Packer et al., 2023; Hu et al., 2026; Li et al., 2026d; Lin et al., 2026). We therefore hypothesize that the future necessity of reasoning depends on the current agent state and on whether task relevant information has already been externalized.

We propose **Interaction Aware Compression for Long Horizon Reasoning (ICLR)**, a training free compression method for long horizon agents. After each interaction step, a frozen proxy model estimates the predictive entropy of reasoning blocks and preferentially compresses low entropy blocks (Li et al., 2026e). The compressed history is directly used by the next agent request. Across long horizon agent tasks from multiple domains, ICLR improves reward from 0.699 to 0.718, while reducing input, output, and cache read tokens by 25.5%, 14.4%, and 33.3%.

Our ablations reveal that local deletion and total system computation are not proportional. Small changes to reasoning history can produce much larger changes in token consumption, while removing all reasoning does not necessarily produce the shortest trajectory because local interventions alter later decisions. We refer to this effect as *trajectory amplification*.

To understand which reasoning should remain available, we further study its *future necessity*. Using representation probing techniques (Alain & Bengio, 2016; Hewitt & Liang, 2019; Belinkov, 2022), we find that the hidden state at the current context boundary contains decodable information associated with later rederivation, additional reasoning, and replanning, reaching an AUROC of 0.844 under strict task level nested evaluation. Activation patching provides complementary evidence (Meng et al., 2022): replacing deleted history representations with their full history counterparts progressively restores the perturbed next token distribution, with KL Repair increasing from 0.047 at layer 4 to 0.459 at layer 20, 0.724 at layer 28, and 0.978 at layer 31.

Trajectory analysis further suggests an *internalization and externalization gap*. When task specific derived state remains available only in reasoning, future behavioral risk is 67.3%, compared with 36.2% after that state has been externalized. Controlled interventions support the same interpretation: keeping only the latest reasoning preserves the full history rule score, deleting it reduces the score, clearing reasoning after relevant code has been written leaves the final score unchanged, and hiding observed environmental feedback causes the agent to recover the same information through an additional interaction.

Together, these findings suggest that historical reasoning is most valuable when it remains the only reliable carrier of task relevant derived state. As that state is externalized, the original reasoning becomes increasingly replaceable.

Our contributions are threefold.

(1) Training free online reasoning compression for agents. We introduce ICLR, which continuously compresses accumulated reasoning inside a real agent interaction loop while preserving actions, tool calls, and observations.

(2) Trajectory level characterization of agent reasoning compression. Through full benchmark evaluation and systematic ablations, we show that reasoning compression can alter the execution trajectory, producing a nonlinear relation between local deletion and total system computation.

(3) Analysis of future reasoning necessity. We study historical reasoning through hidden state probing, activation patching, trajectory analysis, and information externalization. The results support a view of agent reasoning as dynamic working state whose future value depends on where task relevant information is stored.

2 Related Work

2.1 Efficient Reasoning and Chain of Thought Compression

Extended Chain of Thought reasoning has become a central mechanism for mathematical reasoning, code generation, and complex decision making, but longer reasoning traces also increase token usage, latency, and inference cost (Wei et al., 2022; Wang et al., 2022; Hao et al., 2024; Zelikman et al., 2024). A broad line of work therefore studies how much explicit reasoning is actually necessary, including adaptive reasoning budgets, early stopping, concise reasoning, and direct pruning of intermediate reasoning steps (Li et al., 2026e;a; Qiao et al., 2025; Sui et al., 2026; Xia et al., 2025). These studies collectively suggest that reasoning length and task performance are not simply proportional. A complete reasoning trace often mixes critical deductions with repeated checks, confirmations, and procedural elaboration. Identifying which parts can be removed without harming downstream behavior has therefore become an important direction in efficient reasoning.

Recent work further estimates redundancy at the level of individual reasoning steps. Step Entropy (Li et al., 2026e) ranks reasoning steps according to uncertainty in the predictive distribution and shows that many low entropy steps can be removed from mathematical reasoning traces while preserving final answer accuracy. CONCISE (Qiao et al., 2025), TokenSkip (Xia et al., 2025), and related compression methods similarly reduce explicit reasoning by identifying steps or tokens that contribute less to the final solution. These methods mainly optimize reasoning within a single completed generation. The decision to remove a step is therefore evaluated against a fixed downstream answer rather than an evolving sequence of actions and observations.

Our setting differs because historical reasoning remains part of the agent's future decision state. Removing a reasoning block can change the next action, which changes the next observation and all later decisions. The relevant objective is therefore not only whether compressed reasoning still supports the same answer, but whether repeated compression remains effective throughout a closed loop interaction.

2.2 Context Management and Memory for Language Model Agents

The growth of interaction history has motivated context management methods for long horizon agents (Liu et al., 2024; Chen et al., 2024; Munkhdalai et al., 2024; Xiao et al., 2024; Zhang et al., 2023). Token level prompt compression methods such as LLMLingua (Jiang et al., 2023), LongLLMLingua (Jiang et al., 2024), LLMLingua-2 (Pan et al., 2024), context compression (Li et al., 2023), and gist tokens (Mu et al., 2023) reduce redundant context. Agent oriented approaches manage evolving histories through relevance estimation, summarization, adaptive pruning, explicit context operations, or external memory (Sun et al., 2025; Wu et al., 2025; Ye et al., 2025; Verma, 2026). Representative examples include ACON (Kang et al., 2025), PACE (Wei et al., 2026), SWE Pruner (Wang et al., 2026), Self Compact (Li et al., 2026c), Self GC (Hao et al., 2026), Sculptor (Li et al., 2026b), ContextBudget (Wu et al., 2026), Context as a Tool (Liu et al., 2026), ARC (Yao et al., 2026), and structured context eviction (Semenov & Dorofeev, 2026).

A complementary line of work uses persistent or external memory to reduce dependence on the active context. MemGPT (Packer et al., 2023), SAM (Hu et al., 2026), CoMem (Zhang et al., 2026), ACM (Li et al., 2026d), Mem1 (Zhou et al., 2026), proactive memory extraction (Yang et al., 2026), and transferable agent memory (Liang et al., 2026) preserve or recover information outside the immediate interaction window. These systems ask what information should remain accessible. We instead ask whether reasoning that has already produced an action still needs to remain after its consequences have been observed.

This distinction follows from the information flow of an agent. Reasoning combines available evidence into plans and intermediate conclusions, while actions and observations can move the same task relevant information into code, files, tool outputs, or environmental feedback (Lin et al., 2026; Liu et al., 2026; Yao et al., 2026). Historical reasoning may therefore change from being the only carrier of useful derived state to being redundant with information already stored elsewhere. Prior work on hidden state probing (Alain & Bengio, 2016; Hewitt & Liang, 2019; Belinkov, 2022; Burns et al., 2022) and activation interventions (Meng et al., 2022) provides tools for studying this transition. We use them to examine whether the current model state predicts future reasoning reuse and whether reasoning becomes more replaceable after task relevant state is externalized.

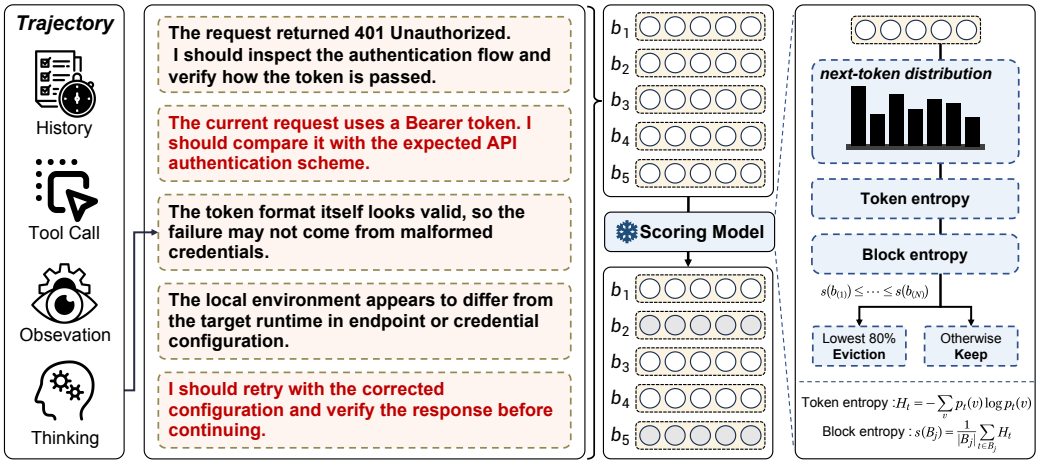


Figure 2: **Overview of ICLR.** At each completed agent step, newly generated reasoning is partitioned into blocks while actions, tool calls, and observations are preserved. A frozen proxy scorer estimates token level predictive entropy for each block. Low entropy blocks are removed before the compressed history is written back for the next agent request.

3 METHODOLOGY

Figure 2 summarizes the online compression process. ICLR operates inside the interaction loop rather than on a completed trajectory. It reduces accumulated reasoning while preserving the external records that define the evolving task state. We first describe online compression and entropy based selection, then the analyses used to study when historical reasoning remains useful.

3.1 ONLINE REASONING COMPRESSION

We consider a long horizon agent that repeatedly reasons, acts, and receives environmental feedback. Let the tth interaction step be

$$S_t = (R_t, A_t, O_{t+1}), \tag{1}$$

where R_t is the generated reasoning, A_t the resulting action or tool call, and O_{t+1} the returned observation. Before the kth model request, the interaction history is

$$\mathcal{H}_k = (P, S_1, \ldots, S_{k-1}), \tag{2}$$

where P contains the system prompt, tool definitions, and task instruction.

Static Chain of Thought compression modifies a completed reasoning trace before evaluating its final answer. Online compression instead changes the state from which future actions are generated. For the kth request, ICLR induces

$$\mathcal{H}_k \to \widetilde{\mathcal{H}}_k \to A_k \to O_{k+1} \to \mathcal{H}_{k+1}. \tag{3}$$

A local compression decision can therefore affect later tool selection, observations, recovery, and termination.

The first agent request is uncompressed. Before each subsequent request, ICLR compresses only the reasoning from the just completed interaction step while preserving its action, tool call, and observation. The compressed reasoning is written back into persistent history and is not restored, so all later model calls operate on the modified trajectory.

Local text removal is not equivalent to total computation saved. A small change to reasoning history can alter later decisions and therefore change both the number and content of future requests. Local deletion should thus be distinguished from total computation over the resulting trajectory.

3.2 ENTROPY GUIDED AND CONTEXT CONDITIONED PRUNING

For each completed reasoning trace, we partition the reasoning into blocks,

$$R_t = (B_1, B_2, \ldots, B_N). \tag{4}$$

The current implementation uses double newline boundaries as a deterministic segmentation rule and excludes empty blocks. Compression is applied only to reasoning. Actions, tool arguments, tool outputs, observations, and final answers remain unchanged. This design prevents the compression policy from directly deleting evidence that has already entered the external interaction record.

Proxy entropy. We score each reasoning block using a frozen proxy language model (Li et al., 2026e). Given scorer context c_t, let $z(c_t)$ denote the logits predicting the next token. The predictive distribution is

$$p_\theta(v \mid c_t) = \mathrm{softmax}(z(c_t))_v \,. \tag{5}$$

The token level predictive entropy is

$$h_t = -\sum_{v \in \mathcal{V}} p_\theta(v \mid c_t) \log_2 p_\theta(v \mid c_t), \tag{6}$$

and the score of block B_i is the mean entropy of its tokens,

$$H(B_i) = \frac{1}{|B_i|} \sum_{t \in B_i} h_t. \tag{7}$$

Lower entropy indicates greater predictive certainty under the frozen proxy model. We use this *proxy entropy* to rank reasoning blocks for compression. The acting agent is DeepSeek V4 Flash, while the scorer is a frozen Qwen3.5 9B model.

Context conditioned ranking. A reasoning block is not scored in isolation. The scorer receives the system prompt, tool definitions, accumulated interaction history, and the current completed step in their original causal order. The score can therefore be viewed as a context dependent quantity,

$$H(B_i \mid \mathcal{H}_t), \tag{8}$$

so the same reasoning text may receive a different score under a different trajectory state. This is important in agent settings because the relevance of earlier reasoning can change after new actions and observations have modified the task state.

The full context scorer has a 42K token input limit. If this limit is exceeded, pruning is skipped for that step rather than truncating the history observed by the acting agent.

Block selection. Given N reasoning blocks, ICLR removes the lowest entropy fraction,

$$\mathcal{D}_t = \mathrm{BottomK}_{\lfloor \rho N \rfloor} \left(\{H(B_i)\}_{i=1}^{N} \right), \qquad \rho = 0.8. \tag{9}$$

Each block in $\mathcal{D}_t$ is replaced with `[SKIP]`, while all retained blocks preserve their original order and text. When $N = 1$, no block is removed. Because the rule operates on blocks rather than individual tokens, the realized token reduction varies across interaction steps.

We use a fixed compression ratio rather than tuning a task specific threshold. This choice creates a consistent intervention across trajectories and allows us to study whether agents can repeatedly discard a large fraction of historical reasoning during real interaction.

Intervention variants. We construct three additional variants for controlled comparison. *Step Only Entropy* scores only the reasoning from the current completed step and removes earlier trajectory context from the scorer. *Delete All Thinking* removes the complete targeted reasoning trace while preserving actions and observations. *Random r* removes a deterministic random subset of reasoning tokens with

$$r \in \{5\%, 10\%, 20\%, 40\%\}. \tag{10}$$

All variants follow the same online write back protocol. The modified history is therefore consumed by every subsequent real agent request. These interventions separate the effects of context conditioning, selection strategy, and deletion strength.

3.3 Future Necessity Probing

The compression policy determines which reasoning is removed, but it does not explain which reasoning will matter again later. We therefore study whether the current model state contains information associated with future reasoning reuse before that later behavior occurs. Following standard representation probing methods (Alain & Bengio, 2016; Hewitt & Liang, 2019; Belinkov, 2022), we analyze the model state available at each compression boundary.

We use a frozen Qwen3.5 9B model as a representation sensor. For sample i, layer l, and pooling rule p, let $\mathcal{Z}_i^{(l)}$ denote the hidden states associated with the current context boundary. We define the pooled representation as

$$z_i^{(l,p)} = \mathrm{Pool}_p\left(\mathcal{Z}_i^{(l)}\right). \tag{11}$$

A linear probe then predicts a behavioral target y_i,

$$\widehat{p}_i = \sigma\left(w^\top z_i^{(l,p)} + b\right). \tag{12}$$

The target records whether the later trajectory contains rederivation, additional reasoning, or explicit replanning. We use this behavioral target as an operational measure of future reasoning reuse. The probe characterizes this signal from the current representation.

To prevent task leakage, evaluation uses strict task level nested splits. Layer selection, pooling choice, feature standardization, and PCA are selected using training tasks only. The outer evaluation folds contain held out tasks. This protocol asks whether the current representation contains information that generalizes across tasks rather than information specific to trajectories already seen during probe selection. Full extraction and evaluation details are reported in Appendix G.

3.4 Mechanistic Interventions for Reasoning Persistence

Representation probing tests whether future reasoning reuse can be predicted from the current state. We complement this analysis with interventions that examine how historical reasoning affects model predictions and how its role changes after task relevant information becomes available outside the reasoning trace.

Activation patching. We first test whether representations conditioned on historical reasoning participate in the next token prediction of the frozen sensor model (Meng et al., 2022). For sample i, let P_i^{full} denote the next token distribution under the full reasoning history and P_i^{del} the corresponding distribution after historical reasoning is removed. At layer l, we replace the prompt boundary residual under the deleted history condition with the residual from the full history condition. The resulting distribution is denoted by $P_{i,l}^{\mathrm{patch}}$.

Let

$$d_i = D_{\mathrm{KL}}\left(P_i^{\mathrm{full}} \| P_i^{\mathrm{del}}\right). \tag{13}$$

For samples with $d_i > 0$, we measure distributional recovery as

$$\mathrm{Repair}_i(l) = 1 - \frac{D_{\mathrm{KL}}\left(P_i^{\mathrm{full}} \| P_{i,l}^{\mathrm{patch}}\right)}{d_i}. \tag{14}$$

Larger values indicate that the patched representation more closely recovers the full history next token distribution. The intervention is performed only on the frozen sensor model and does not modify the online ICLR policy.

State externalization. We next examine whether reasoning becomes more replaceable after task specific derived state has been written into a persistent external carrier. Let $\mathcal{U}_i$ denote the unexternalized task specific derived state at compression boundary i. We define

$$G_i = \mathbb{I}\left(\mathcal{U}_i \neq \varnothing\right). \tag{15}$$

Here, derived state refers to an intermediate plan, relation, constraint, synthesis, or task specific conclusion rather than a direct copy of the task instruction or a raw observation. External carriers include code, files, tool outputs, and explicit environmental feedback.

The indicator G_i separates states in which reasoning remains the only available carrier of task relevant derived information from states in which that information has already been materialized elsewhere. This allows us to study whether future reasoning reuse changes as information moves from internal reasoning into persistent external state.

Controlled state interventions. Finally, we construct fixed state continuations that modify the available reasoning or feedback while keeping the surrounding task state unchanged. The interventions compare retaining or deleting recent reasoning, clearing reasoning after relevant code has been written to disk, and retaining or hiding previously observed environmental feedback.

These interventions test whether the next decision still depends on information available only in the reasoning trace. Together, probing, activation patching, state externalization analysis, and controlled interventions provide complementary views of reasoning persistence. The full protocols are reported in Appendices H and I.

4 EXPERIMENTS

We evaluate ICLR on WorkBuddyBench using DeepSeek V4 Flash as the task executing agent and a frozen Qwen3.5 9B model as the proxy entropy scorer. WorkBuddyBench contains long horizon tasks from Code, Office, Security, and Web domains. We report official task reward together with input tokens, output tokens, cache read tokens, and wins, ties, and losses. The full benchmark contains 260 tasks, consisting of 80 Code, 50 Office, 60 Security, and 70 Web tasks. A fixed 80 task set with 20 tasks from each domain is used for ablation analysis.

4.1 OVERALL PERFORMANCE

We first compare ICLR with representative context management baselines on the fixed Pilot40 task set used by prior WorkBuddyBench context management experiments. Table 1 includes simple sliding window baselines, periodic summarization, and representative context management methods including PACE, SelfCompact, ACON Core, SAM, and SWE Pruner. This comparison focuses on the tradeoff between task quality and token consumption. ICLR directly prunes low entropy reasoning from the online interaction history using a frozen scorer, providing a lightweight alternative to summary based or external memory based context management.

We next evaluate ICLR on the complete 260 task benchmark. Table 2 follows the score and token presentation used by prior WorkBuddyBench context management evaluations. Average reward increases from 0.699 to 0.718. Detailed token accounting is reported separately in the appendix. The method obtains $102/76/82$ wins, ties, and losses. These results show that aggressive online pruning of reasoning can reduce repeated processing of historical context without sacrificing aggregate task performance across domains.

Domain level results and component level token changes are reported in Appendix E.

4.2 ABLATIONS AND TRAJECTORY AMPLIFICATION

We next study whether the observed effect depends on entropy based selection, access to full history during scoring, or simply reducing the amount of reasoning. Table 3 reports results on the fixed 80 task ablation set. ICLR reaches a reward of 0.738, compared with 0.627 for the baseline. Step only entropy reaches 0.711, while deleting all thinking reaches 0.718. Random deletion also changes task performance, but none of the reported random variants reaches the reward of full history entropy. These results suggest that reasoning history contains substantial redundancy while also indicating that context conditioned ranking provides useful information beyond deletion alone. Appendix F reports the full intervention and token summaries used for this analysis.

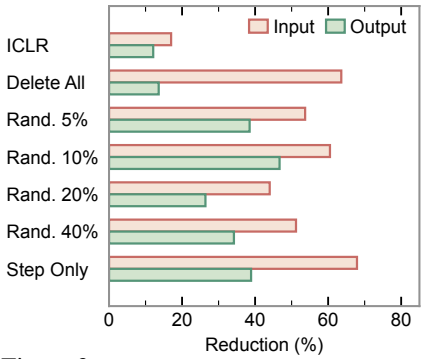


Figure 3: **Trajectory amplification.** Token savings do not track deletion.

Table 1: **Comparison with context management methods on WorkBuddyBench Pilot40.** All methods use the same fixed 40 tasks. Scores are multiplied by 100. Parenthesized values show changes from the DeepSeek-V4-Flash base agent.

Method	Code	Office	Sec.	Web	Avg.	Tokens ↓
Base and Ours						
DeepSeek-V4-Flash	72.9	81.6	44.6	71.0	67.5	1.89M
ICLR (Ours)	74.1 (+1.2)	81.1 (-0.5)	**64.1** (+19.5)	62.0 (-9.0)	**70.3** (+2.8)	2.02M (+6.9%)
Simple Context Baselines						
Sliding Window ($K = 5$)	66.8 (-6.1)	67.4 (-14.2)	32.3 (-12.3)	59.0 (-12.0)	56.4 (-11.2)	2.95M (+56.3%)
Sliding Window ($K = 10$)	74.4 (+1.4)	73.1 (-8.6)	40.1 (-4.5)	61.0 (-10.0)	62.1 (-5.4)	1.98M (+5.1%)
Sliding Window ($K = 20$)	79.3 (+6.4)	85.1 (+3.5)	40.3 (-4.3)	70.0 (-1.0)	68.7 (+1.1)	2.02M (+7.0%)
Periodic Summary ($n = 3$)	65.4 (-7.5)	67.7 (-14.0)	26.3 (-18.3)	**82.7** (+11.7)	60.5 (-7.0)	2.21M (+17.2%)
Periodic Summary ($n = 5$)	**83.6** (+10.7)	75.7 (-5.9)	40.0 (-4.5)	81.3 (+10.3)	**70.2** (+2.6)	2.37M (+25.5%)
API Based Methods						
PACE (Wei et al., 2026)	70.4 (-2.6)	55.8 (-25.9)	43.5 (-1.0)	64.0 (-7.0)	58.4 (-9.1)	1.10M (-41.6%)
LLMLingua-2 (Pan et al., 2024)	64.6 (-8.3)	73.3 (-8.3)	40.1 (-4.5)	70.0 (-1.0)	62.0 (-5.5)	2.40M (+27.3%)
SelfCompact (Li et al., 2026c)	76.8 (+3.9)	79.6 (-2.0)	36.2 (-8.4)	71.0 (0.0)	65.9 (-1.6)	1.57M (-17.0%)
ACON-Core (Kang et al., 2025)	71.8 (-1.1)	68.6 (-13.1)	**47.9** (+3.4)	65.0 (-6.0)	63.3 (-4.2)	1.40M (-25.7%)
Self-GC (Hao et al., 2026)	63.5 (-9.5)	71.7 (-10.0)	43.7 (-0.9)	71.0 (0.0)	62.5 (-5.1)	1.66M (-12.4%)
LRE (Jahan Lia & Mazumder, 2026)	62.6 (-10.3)	63.4 (-18.2)	27.1 (-17.4)	68.0 (-3.0)	55.3 (-12.2)	2.29M (+21.3%)
CoMem (Zhang et al., 2026)	56.4 (-16.5)	59.9 (-21.8)	2.5 (-42.1)	58.0 (-13.0)	44.2 (-23.3)	**0.77M** (-59.5%)
SAM (Hu et al., 2026)	67.6 (-5.3)	81.6 (0.0)	47.2 (+2.7)	67.0 (-4.0)	65.9 (-1.7)	1.35M (-28.7%)
SWE-Pruner (Wang et al., 2026)	63.5 (-9.5)	72.6 (-9.0)	40.9 (-3.7)	65.0 (-6.0)	60.5 (-7.1)	1.50M (-20.5%)
Sculptor (Li et al., 2026b)	51.2 (-21.7)	81.6 (0.0)	35.2 (-9.3)	69.0 (-2.0)	59.3 (-8.3)	1.29M (-31.5%)
Released Policy Method						
ACM (Li et al., 2026d)	38.7 (-34.2)	55.9 (-25.7)	6.2 (-38.4)	34.0 (-37.0)	33.7 (-33.8)	2.36M

Table 2: **Full benchmark system performance and token usage.** Scores are multiplied by 100. Green and red indicate favorable and unfavorable changes from the base system.

Model	Code	Office	Sec.	Web	Avg.	Tokens ↓
DeepSeek-V4-Flash	**77.0**	**81.8**	47.8	**72.1**	69.9	2.69M
ICLR (Ours)	70.6 (-6.4)	77.6 (-4.2)	**70.7** (+22.9)	69.9 (-2.2)	**71.8** (+1.9)	**2.19M** (-18.5%)

The most striking observation is that the local deletion ratio is not monotonically related to final system level computation. Random deletion of 10% of reasoning tokens reduces total input by 60.5%, whereas random deletion of 20% reduces input by only 44.0%. Deleting all thinking reduces input by 63.6%, which is not qualitatively proportional to deleting all reasoning text. Different interventions also produce substantially different numbers of agent calls. We refer to this behavior as *trajectory amplification*: a local change to reasoning history can be amplified through tool selection, observations, repeated reasoning, recovery, and termination, producing a much larger change in the final trajectory. Detailed intervention and token results are reported in Appendix F.

4.3 Future Necessity Analysis

We evaluate future necessity on 265 usable compression boundary samples from 31 tasks, with 112 positive behavioral labels. Under strict task level nested evaluation, the probe reaches an AUROC of 0.844, an AP of 0.811, and a balanced accuracy of 0.760. The prompt last representation is selected in all outer folds, while the selected layers span middle and later parts of the network. These results show that the current representation contains a readable signal associated with later rederivation, additional reasoning, and replanning across held out tasks. Full cohort and layer selection details are reported in Appendix G.1.

Activation patching provides complementary evidence at the representation level. On the 47 sample patching cohort, replacing deleted history residuals with their full history counterparts progressively

Table 3: **Ablation results on the fixed 80 task WorkBuddyBench set.** All variants use the same fixed 80 tasks, with 20 Code, 20 Office, 20 Security, and 20 Web tasks. Scores are multiplied by 100. Parenthesized values show changes from the Base Agent.

Method	Code	Office	Sec.	Web	Avg.	Tokens ↓
Base and Ours						
Base Agent	69.09	78.94	44.17	58.50	62.67	3.00M
ICLR (Ours)	63.85 (-5.24)	81.14 (+2.20)	72.03 (+27.86)	**78.00** (+19.50)	**73.76** (+11.09)	2.49M (-17.0%)
Reasoning Removal Controls						
Delete All Thinking	**69.20** (+0.11)	82.04 (+3.10)	67.41 (+23.24)	68.50 (+10.00)	71.78 (+9.11)	1.11M (-62.9%)
Random 5%	59.81 (-9.28)	81.12 (+2.18)	64.17 (+20.00)	66.83 (+8.33)	67.98 (+5.31)	1.40M (-53.5%)
Random 10%	55.08 (-14.01)	**82.65** (+3.71)	71.91 (+27.74)	60.61 (+2.11)	67.56 (+4.89)	1.19M (-60.3%)
Random 20%	67.23 (-1.86)	76.92 (-2.02)	64.82 (+20.65)	61.50 (+3.00)	67.62 (+4.95)	1.69M (-43.8%)
Random 40%	61.35 (-7.74)	76.17 (-2.77)	61.37 (+17.20)	71.62 (+13.12)	67.63 (+4.96)	1.47M (-51.0%)
Scoring Context Ablation						
Step Only Entropy	60.48 (-8.61)	80.58 (+1.64)	**73.10** (+28.93)	70.38 (+11.88)	71.14 (+8.47)	0.98M (-67.5%)

restores the next token distribution. Mean KL Repair rises from 0.047 at layer 4 to 0.459 at layer 20, 0.724 at layer 28, and 0.978 at layer 31. The complete layerwise results, confidence intervals, and top one agreement values are reported in Appendix H, Table 13.

4.4 State Externalization and Controlled Interventions

Trajectory analysis provides a state based explanation for when historical reasoning remains useful. When task specific derived state is still available only in reasoning, the future behavioral risk rate is 67.3%, compared with 36.2% when that state has already been externalized. The within task increase is 19.8 percentage points, and the corresponding hidden score difference is 0.283. Detailed group statistics and task level intervals are reported in Appendix I.1.

Controlled interventions support the same interpretation. Keeping only the latest reasoning preserves the full history rule score of 0.872, whereas deleting only the latest reasoning reduces the score to 0.760, and deleting all reasoning reduces it to 0.675. Once relevant code has been written to disk, clearing subsequent reasoning leaves the rule score unchanged at 0.759. Environmental feedback shows the same pattern. When an observed `stderr` remains available, the agent proceeds directly to Edit. When the same feedback is hidden, the agent issues another Bash call and observes the same error again. The fixed state experiments are reported in Appendix I.2, and the environmental feedback intervention is reported in Appendix I.3.

Taken together, these results suggest that historical reasoning is most useful while it remains the only reliable carrier of task relevant derived state. As that state is transferred into persistent artifacts or environmental feedback, the original reasoning becomes increasingly replaceable.

5 Conclusion

We study when a long horizon agent can forget its own reasoning. Unlike static Chain of Thought compression, online compression changes future decisions and the resulting trajectory. ICLR uses frozen proxy entropy to remove low entropy reasoning while preserving actions and observations, reducing token consumption and improving aggregate reward on the full 260 task evaluation. Ablations reveal trajectory amplification, where local deletion and final system savings are not proportional. Representation probing and activation patching show that current states encode information associated with later reasoning reuse, while controlled analyses indicate that historical reasoning becomes more replaceable as task relevant derived state moves into reliable external carriers. These results more clearly characterize agent reasoning as dynamic working state whose value changes throughout interaction.

AI USE STATEMENT

Generative AI tools were used for language editing, drafting assistance, and literature discovery during preparation of this manuscript. All AI assisted text, claims, citations, and experimental descriptions were reviewed by the authors. Experimental results and interpretations were checked against the underlying records, and the authors take full responsibility for the final content of the paper.

REPRODUCIBILITY STATEMENT

The main paper specifies the online compression protocol, proxy entropy definition, pruning rule, benchmark composition, and principal ablations. The appendix provides implementation details, evaluation set composition, extended system and ablation results, hidden state probing and activation patching protocols, controlled state interventions, and theoretical analysis.

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. *arXiv preprint arXiv:2212.03827*, 2022.

Yukang Chen, Shengju Qian, Haotian Tang, Xin Lai, Zhijian Liu, Song Han, and Jiaya Jia. Longlora: Efficient fine-tuning of long-context large language models. In *International Conference on Learning Representations*, volume 2024, pp. 8220–8238, 2024.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*, 2024.

Xubin Hao, Hongjin Meng, Xin Yin, Jiawei Zhu, and Chenpeng Cao. Self-gc: Self-governing context for long-horizon llm agents. *arXiv preprint arXiv:2607.00692*, 2026.

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp)*, pp. 2733–2743, 2019.

Yuyang Hu, Hongjin Qian, Shuting Wang, Jiongnan Liu, Ziliang Zhao, Jiejun Tan, Zheng Liu, and Zhicheng Dou. Sam: State-adaptive memory for long-horizon reasoning agent. *arXiv preprint arXiv:2605.24468*, 2026.

Nusrat Jahan Lia and Aritra Mazumder. Learning what not to forget: Long-horizon agent memory from a few kilobytes of learning. *arXiv e-prints*, pp. arXiv–2606, 2026.

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.

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.

Minki Kang, Wei-Ning Chen, Dongge Han, Huseyin A Inan, Lukas Wutschitz, Yanzhi Chen, Robert Sim, and Saravan Rajmohan. Acon: Optimizing context compression for long-horizon llm agents. *arXiv preprint arXiv:2510.00615*, 2025.

Juncai Li, Ru Li, Yuxiang Zhou, Boxiang Ma, and Jeff Z Pan. Chain of thought compression: A theoretical analysis. *arXiv preprint arXiv:2601.21576*, 2026a.

Mo Li, LH Xu, Qitai Tan, Long Ma, Hongyong Song, Ting Cao, and Yunxin Liu. Sculptor: Empowering llms with cognitive agency via active context management. In *International Conference on Learning Representations*, volume 2026, pp. 153411–153440, 2026b.

Tianjian Li, Jingyu Zhang, William Jurayj, Xi Wang, Chuanyang Jin, Mehrdad Farajtabar, Eric Nalisnick, and Daniel Khashabi. Self-compacting language model agents. *arXiv preprint arXiv:2606.23525*, 2026c.

Xiaochuan Li, Ryan Ming, Meng Chu, Shuai Shao, Rong Jin, and Chenyan Xiong. Acm: Agentic context management for long horizon tasks. *arXiv preprint arXiv:2607.23809*, 2026d.

Yucheng Li, Bo Dong, Frank Guerin, and Chenghua Lin. Compressing context to enhance inference efficiency of large language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 6342–6353, 2023.

Zeju Li, Jianyuan Zhong, Ziyang Zheng, Xiangyu Wen, Zhijian Xu, Yingying Cheng, Fan Zhang, and Qiang Xu. Making slow thinking faster: Compressing llm chain-of-thought via step entropy. In *International Conference on Learning Representations*, volume 2026, pp. 38331–38349, 2026e.

Sirui Liang, Pengfei Cao, Jian Zhao, Wenhao Teng, Xiangwen Liao, Jun Zhao, and Kang Liu. Learning how to remember: A meta-cognitive management method for structured and transferable agent memory. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 30733–30753, 2026.

Yin Lin, Elaine Ang, Erkang Zhu, Bolin Ding, and Jingren Zhou. Context as an environment: Programmatic context management for long-horizon agents. *arXiv preprint arXiv:2608.21690*, 2026.

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

Shukai Liu, Bo Jiang, Jian Yang, Yizhi Li, Jinyang Guo, Xianglong Liu, and Bryan Dai. Context as a tool: Context management for long-horizon swe-agents. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 20604–20617, 2026.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372, 2022.

Jesse Mu, Xiang Li, and Noah Goodman. Learning to compress prompts with gist tokens. *Advances in Neural Information Processing Systems*, 36:19327–19352, 2023.

Tsendsuren Munkhdalai, Manaal Faruqui, and Siddharth Gopal. Leave no context behind: Efficient infinite context transformers with infini-attention. *arXiv preprint arXiv:2404.07143*, 2024.

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.

Zhuoshi Pan, Qianhui Wu, Huiqiang Jiang, Menglin Xia, Xufang Luo, Jue Zhang, Qingwei Lin, Victor Rühle, Yuqing Yang, Chin-Yew Lin, et al. Llmlingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 963–981, 2024.

Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*, pp. 1–22, 2023.

Ziqing Qiao, Yongheng Deng, Jiali Zeng, Dong Wang, Lai Wei, Guanbo Wang, Fandong Meng, Jie Zhou, Ju Ren, and Yaoxue Zhang. Concise: Confidence-guided compression in step-by-step efficient reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 8021–8040, 2025.

Andrew Semenov and Svyatoslav Dorofeev. Beyond compaction: Structured context eviction for long-horizon agents. *arXiv preprint arXiv:2606.11213*, 2026.

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.

Yi Sui, Chaozhuo Li, and Dawei Song. Think less, know more: State-aware reasoning compression with knowledge guidance for efficient reasoning. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 22470–22491, 2026.

Weiwei Sun, Miao Lu, Zhan Ling, Kang Liu, Xuesong Yao, Yiming Yang, and Jiecao Chen. Scaling long-horizon llm agent via context-folding. *arXiv preprint arXiv:2510.11967*, 2025.

Nikhil Verma. Active context compression: Autonomous memory management in llm agents. *arXiv preprint arXiv:2601.07190*, 2026.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

Yuhang Wang, Yuling Shi, Mo Yang, Rongrui Zhang, Shilin He, Heng Lian, Yuting Chen, Siyu Ye, Kai Cai, and Xiaodong Gu. Swe-pruner: Self-adaptive context pruning for coding agents. *arXiv preprint arXiv:2601.16746*, 2026.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Lei Wei, Xiao Peng, Guannan Zhang, Chenhao Jiang, Hongyu Li, Lanbo Lin, Yuanwu Xu, Jiayao Liu, Kesu Wang, Bin Wang, et al. Pace: Predictive adaptive context extraction for long-horizon llm agents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 27184–27199, 2026.

Xixi Wu, Kuan Li, Yida Zhao, Liwen Zhang, Litu Ou, Huifeng Yin, Zhongwang Zhang, Xinmiao Yu, Dingchu Zhang, Yong Jiang, et al. Resum: Unlocking long-horizon search intelligence via context summarization. *arXiv preprint arXiv:2509.13313*, 2025.

Yong Wu, YanZhao Zheng, TianZe Xu, ZhenTao Zhang, YuanQiang Yu, JiHuai Zhu, Chao Ma, BinBin Lin, BaoHua Dong, HangCheng Zhu, et al. Contextbudget: Budget-aware context management for long-horizon search agents. *arXiv preprint arXiv:2604.01664*, 2026.

Heming Xia, Chak Tou Leong, Wenjie Wang, Yongqi Li, and Wenjie Li. Tokenskip: Controllable chain-of-thought compression in llms. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 3351–3363, 2025.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *International Conference on Learning Representations*, volume 2024, pp. 21875–21895, 2024.

Chengyuan Yang, Zequn Sun, Wei Wei, and Wei Hu. Beyond static summarization: Proactive memory extraction for llm agents. *arXiv preprint arXiv:2601.04463*, 2026.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022.

Yilun Yao, Shan Huang, Elsie Dai, Zhewen Tan, Zhenyu Duan, Shousheng Jia, Yanbing Jiang, and Tong Yang. Arc: Active and reflection-driven context management for long-horizon information seeking agents. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 18644–18659, 2026.

Rui Ye, Zhongwang Zhang, Kuan Li, Huifeng Yin, Zhengwei Tao, Yida Zhao, Liangcai Su, Liwen Zhang, Zile Qiao, Xinyu Wang, et al. Agentfold: Long-horizon web agents with proactive context management. *arXiv preprint arXiv:2510.24699*, 2025.

Eric Zelikman, Georges Harik, Yijia Shao, Varuna Jayasiri, Nick Haber, and Noah D Goodman. Quiet-star: Language models can teach themselves to think before speaking. *arXiv preprint arXiv:2403.09629*, 2024.

Yuwei Zhang, Chengyu Dong, Shuowei Jin, Changlong Yu, Hejie Cui, Hongye Jin, Xinyang Zhang, Hamed Bonab, Colin Lockard, Jianshu Chen, et al. Comem: Context management with a decoupled long-context model. *arXiv preprint arXiv:2605.30842*, 2026.

Zhenyu Zhang, Ying Sheng, Tianyi Zhou, Tianlong Chen, Lianmin Zheng, Ruisi Cai, Zhao Song, Yuandong Tian, Christopher Ré, Clark Barrett, et al. H2o: Heavy-hitter oracle for efficient generative inference of large language models. *Advances in neural information processing systems*, 36:34661–34710, 2023.

Zijian Zhou, Ao Qu, Zhaoxuan Wu, Sunghwan Kim, Alok Prakash, Daniela Rus, Bryan Kian Hsiang Low, and Paul Liang. Mem1: Learning to synergize memory and reasoning for efficient long-horizon agents. In *International Conference on Learning Representations*, volume 2026, pp. 58413–58438, 2026.

A THE USE OF LARGE LANGUAGE MODELS

Generative AI tools were used for language editing, drafting assistance, literature discovery, LaTeX preparation, and organization of supplementary explanations. All claims, citations, mathematical statements, and experimental descriptions were reviewed by the authors.

Scope. The supplementary material expands the definitions, protocols, theoretical properties, and experimental results reported in the main paper. In particular, the theoretical appendix distinguishes population information quantities from the realized frozen proxy score and states the scope of each sufficient condition explicitly.

B THEORETICAL ANALYSIS

B.1 NOTATION AND SCOPE

At one compression boundary, let C denote the context preceding the completed reasoning, $B = (B_1, \ldots, B_N)$ the ordered reasoning blocks, and U the unchanged nonreasoning records of the completed step, including its action, tool record, and returned observation. Let Y denote a downstream target under a fixed reference distribution P. We write $B_{<j} = (B_1, \ldots, B_{j-1})$, B_{-j} for all reasoning blocks except B_j, and B_D for the chronologically ordered tuple indexed by D.

We use H_P and I_P for population Shannon entropy and mutual information, while $H(B_j)$ in Section 3.2 denotes the realized proxy score produced by the frozen scorer. The distinction is important: the population quantities below characterize information structure, whereas the implemented score is a model dependent ranking signal. The results in this section establish information bounds and ranking stability conditions, together with trajectory perturbation bounds and token accounting decompositions.

B.2 RESIDUAL INFORMATION OF A REASONING BLOCK

Lemma B.1 (Residual information bound)**.** *For every block B_j,*

$$I_P(B_j; Y \mid C, U, B_{-j}) \le \mathsf{H}_P(B_j \mid C, B_{<j}). \tag{B.1}$$

Proof. By the definition of conditional mutual information,

$$\begin{aligned} I_P(B_j; Y \mid C, U, B_{-j}) &= \mathsf{H}_P(B_j \mid C, U, B_{-j}) - \mathsf{H}_P(B_j \mid C, U, B_{-j}, Y) && \text{(B.2)} \\ &\le \mathsf{H}_P(B_j \mid C, U, B_{-j}) && \text{(B.3)} \\ &\le \mathsf{H}_P(B_j \mid C, B_{<j}). && \text{(B.4)} \end{aligned}$$

The first inequality follows from nonnegativity of conditional entropy. The second follows because conditioning on U and all blocks in B_{-j} includes the earlier blocks $B_{<j}$ and can only reduce conditional entropy. □

Interpretation. Under the reference distribution, a block with low true conditional entropy has a correspondingly limited residual information contribution about Y. The statement is deliberately one sided: high entropy does not imply that a block is useful, while low conditional entropy upper bounds how much residual information the block can contribute after the retained context is known. This motivates conditional uncertainty as a compression signal by bounding the residual information carried by reasoning blocks.

B.3 SEVERAL REMOVED BLOCKS

Theorem B.2 (Fixed subset information bound)**.** *For any index set $D \subseteq \{1, \ldots, N\}$ fixed independently of the realized block values,*

$$I_P(B_D; Y \mid C, U, B_{\bar{D}}) \le \sum_{j \in D} \mathsf{H}_P(B_j \mid C, B_{<j}), \tag{B.5}$$

where $\bar{D}$ is the complement of D.

Proof. Write $D = \{j_1 < \cdots < j_m\}$. Then

$$I_P(B_D; Y \mid C, U, B_{\bar{D}}) \le \mathsf{H}_P(B_D \mid C, U, B_{\bar{D}}) \tag{B.6}$$

$$= \sum_{r=1}^{m} \mathsf{H}_P\left(B_{j_r} \mid C, U, B_{\bar{D}}, B_{j_1}, \ldots, B_{j_{r-1}}\right) \tag{B.7}$$

$$\le \sum_{r=1}^{m} \mathsf{H}_P(B_{j_r} \mid C, B_{<j_r}). \tag{B.8}$$

The first inequality bounds mutual information by conditional entropy. The equality is the entropy chain rule. In each term, the conditioning set contains every earlier block, either because that block is retained in $B_{\bar{D}}$ or because it appears among the previously selected blocks. □

Adaptive ranking. The implemented deletion set is selected from observed proxy entropy scores and is therefore data dependent. Theorem B.2 characterizes the information structure of a fixed subset. ICLR uses this principle to motivate its adaptive ranking heuristic, while the theorem concerns deletion sets fixed independently of the observed block values.

B.4 Proxy entropy and ranking stability

The preceding bounds involve population conditional entropy, while ICLR uses mean token level predictive entropy from a frozen proxy model. To compare the two on the same per token scale, consider a candidate block B_j containing m_j scorer token positions. Write the block tokens as $X_{j,1:m_j}$ and define the length conditioned population uncertainty

$$h_j^\star = \frac{1}{m_j} \mathsf{H}_P\left(X_{j,1:m_j} \mid C, B_{<j}, M_j = m_j\right)$$

$$= \frac{1}{m_j} \sum_{r=1}^{m_j} \mathsf{H}_P\left(X_{j,r} \mid C, B_{<j}, X_{j,<r}, M_j = m_j\right), \tag{B.9}$$

where the second equality follows from the entropy chain rule. Let

$$\widehat{h}_j = H(B_j) \tag{B.10}$$

denote the realized mean proxy entropy used by ICLR.

The next result relates ranking stability to the difference between proxy scores and $h_j^\star$. Under a bounded approximation error, sufficiently separated population scores preserve their ordering in the corresponding proxy ranking.

Lemma B.3 (Pairwise proxy ranking stability)**.** *Suppose that, for the candidate blocks at a fixed compression boundary,*

$$\left|\widehat{h}_j - h_j^\star\right| \le \delta \qquad \textit{for every } j. \tag{B.11}$$

For any two blocks B_i and B_j, if

$$h_i^\star + 2\delta < h_j^\star, \tag{B.12}$$

then

$$\widehat{h}_i < \widehat{h}_j. \tag{B.13}$$

Proof. By Equation equation B.11,

$$\widehat{h}_i \le h_i^\star + \delta. \tag{B.14}$$

Equation equation B.12 gives

$$h_i^\star + \delta < h_j^\star - \delta. \tag{B.15}$$

Applying Equation equation B.11 once more,

$$h_j^\star - \delta \le \widehat{h}_j. \tag{B.16}$$

Combining the three inequalities yields $\widehat{h}_i < \widehat{h}_j$. □

Corollary B.4 (Bottom-k selection stability)**.** *Let*

$$h^\star_{(1)} \le \cdots \le h^\star_{(N)} \tag{B.17}$$

be the ordered population uncertainty scores and let $k = \lfloor \rho N \rfloor$. If Equation equation B.11 holds and

$$h^\star_{(k+1)} - h^\star_{(k)} > 2\delta, \tag{B.18}$$

then the bottom-k set selected by the proxy scores is identical to the bottom-k set selected by the population uncertainty scores.

Proof. Take any i in the population bottom-k set and any j outside it. Then

$$h^\star_i \le h^\star_{(k)} < h^\star_{(k+1)} - 2\delta \le h^\star_j - 2\delta. \tag{B.19}$$

Lemma B.3 therefore gives $\widehat{h}_i < \widehat{h}_j$. Hence every block in the population bottom-k set receives a smaller proxy score than every block outside the set, so the two selections coincide. □

Interpretation. The relevant quantity is the approximation error relative to the score margin at the pruning boundary. A nonzero proxy error need not change the selected blocks when the retained and removed candidates are well separated. When the boundary margin is small, however, even small score perturbations may change the selected subset.

Scope. Corollary B.4 characterizes selection stability through a uniform score error and the pruning boundary margin. Here δ is an assumed error bound, not a measured quantity; the result specifies when the proxy and population selections coincide under the stated approximation condition at the compression boundary.

B.5 Predictive entropy and observed token surprisal

The implemented token score is predictive entropy rather than the surprisal of the token that appears in the reasoning trace. These quantities are related but measure different objects.

For a scorer context c, let

$$q(v \mid c) = p_\theta(v \mid c). \tag{B.20}$$

The scorer's predictive entropy is

$$\mathsf{H}_2\left(q(\cdot \mid c)\right) = -\sum_{v \in \mathcal{V}} q(v \mid c) \log_2 q(v \mid c). \tag{B.21}$$

For a realized token x, its surprisal under the same scorer is

$$s_q(x \mid c) = -\log_2 q(x \mid c). \tag{B.22}$$

If the next token X were sampled from the scorer itself, $X \sim q(\cdot \mid c)$, then

$$\mathbb{E}_{X \sim q}\left[s_q(X \mid c)\right] = -\sum_{v \in \mathcal{V}} q(v \mid c) \log_2 q(v \mid c) \tag{B.23}$$

$$= \mathsf{H}_2\left(q(\cdot \mid c)\right). \tag{B.24}$$

Thus predictive entropy is the expected surprisal under the scorer's own next token distribution.

The distinction is more important when the observed token is generated from a different distribution. Let $r(\cdot \mid c)$ denote an alternative next-token law, for example the distribution induced by a different acting model. Then

$$\mathbb{E}_{X \sim r}\left[s_q(X \mid c)\right] = -\sum_{v \in \mathcal{V}} r(v \mid c) \log_2 q(v \mid c) \tag{B.25}$$

$$= \mathsf{H}_2\left(r(\cdot \mid c)\right) + D_{\mathrm{KL},2}\left(r(\cdot \mid c) \| q(\cdot \mid c)\right), \tag{B.26}$$

where $D_{\mathrm{KL},2}$ denotes KL divergence measured in bits.

Proposition B.5 (Surprisal under distribution mismatch)**.** *For distributions r and q on the same vocabulary with $q(v) > 0$ whenever $r(v) > 0$,*

$$\mathbb{E}_{X\sim r}\left[-\log_2 q(X)\right] = \mathsf{H}_2(r) + D_{\mathrm{KL},2}(r\|q). \tag{B.27}$$

Proof. Starting from the left hand side,

$$-\sum_v r(v)\log_2 q(v) = -\sum_v r(v)\log_2 r(v) + \sum_v r(v)\log_2 \frac{r(v)}{q(v)} \tag{B.28}$$

$$= \mathsf{H}_2(r) + D_{\mathrm{KL},2}(r\|q). \tag{B.29}$$

□

Interpretation. Observed token surprisal depends on the particular realized token and, under a distribution mismatch, also includes a mismatch term $D_{\mathrm{KL},2}(r\|q)$. Predictive entropy instead summarizes uncertainty in the entire next token distribution of the frozen scorer. This distinction is relevant here because the scorer and acting agent are different models. ICLR therefore uses predictive entropy as a proxy uncertainty signal rather than interpreting the acting agent's realized token likelihood as the same quantity.

For a block with scorer positions $\mathcal{I}_j$, the implemented score can equivalently be written as

$$H(B_j) = \frac{1}{|\mathcal{I}_j|}\sum_{t\in\mathcal{I}_j} \mathsf{H}_2\left(q_t\right), \qquad q_t(v) = p_\theta(v \mid c_t). \tag{B.30}$$

Hence the block ranking aggregates distributional uncertainty across the scorer positions assigned to that block.

B.6 Online trajectory perturbation

The main paper emphasizes that a local edit changes the state from which later actions are generated. The following bound makes this dependence explicit. Consider a common finite horizon K and two policies interacting with the same environment from the same initial state. At any common observable history h, let $\pi_t(\cdot \mid h)$ and $\widetilde{\pi}_t(\cdot \mid h)$ denote the next action laws of the base and compressed systems.

Theorem B.6 (Finite horizon trajectory perturbation)**.** *Suppose that for every jointly reachable common history,*

$$\mathrm{TV}\left(\pi_t(\cdot \mid h), \widetilde{\pi}_t(\cdot \mid h)\right) \le \epsilon_t, \qquad 0 \le \epsilon_t \le 1. \tag{B.31}$$

Then the observable trajectory laws satisfy

$$\mathrm{TV}\left(\mathbb{P}_\pi, \mathbb{P}_{\widetilde{\pi}}\right) \le 1 - \prod_{t=1}^{K}(1-\epsilon_t) \le \min\left\{1, \sum_{t=1}^{K}\epsilon_t\right\}. \tag{B.32}$$

For any terminal score $F \in [0,1]$,

$$\left|\mathbb{E}_\pi F - \mathbb{E}_{\widetilde{\pi}} F\right| \le \min\left\{1, \sum_{t=1}^{K}\epsilon_t\right\}. \tag{B.33}$$

Proof. While the two histories agree, couple their next actions maximally. The coupled actions agree with probability at least $1-\epsilon_t$. Conditional on equal actions and equal observable histories, couple the next observation identically through the common environment transition. Induction gives a probability of complete trajectory agreement of at least $\prod_t(1-\epsilon_t)$. Total variation is bounded by the mismatch probability of any coupling, which gives the first inequality. The second follows from the union bound. Since $F \in [0,1]$, its values can differ by at most one on the mismatch event, which yields Equation equation B.33. □

Interpretation. The bound separates a local intervention from its trajectory level consequences. Even if each one step action distribution changes only slightly, those changes can accumulate over a long interaction. Conversely, if local action laws remain close at every reachable common history, the resulting trajectory and terminal score distributions are controlled. This provides a formal basis for the trajectory amplification phenomenon studied empirically in Section 4.

B.7 FORGETTING AFTER STATE EXTERNALIZATION

Let Z_t denote task relevant state that is available from the retained external interaction history, and let W_t denote reasoning content that would be removed. The key question is whether the next action still depends on W_t after Z_t has become a sufficient carrier of the derived task state.

Corollary B.7 (Exact externalized state sufficiency)**.** *Suppose that at every reachable history the full history and compressed policies share the same retained state Z_t, and that*

$$\pi_t(a \mid Z_t, W_t) = q_t(a \mid Z_t) = \widetilde{\pi}_t(a \mid Z_t). \tag{B.34}$$

Under the common environment assumptions of Theorem B.6, the two systems induce identical observable trajectory and terminal score distributions.

Proof. Equation equation B.34 implies

$$\mathrm{TV}\left(\pi_t(\cdot \mid Z_t, W_t), \widetilde{\pi}_t(\cdot \mid Z_t)\right) = 0 \tag{B.35}$$

at every reachable common state. Hence $\epsilon_t = 0$ for all t in Theorem B.6, so both the trajectory law distance and the terminal score difference are zero. □

The exact equality above is stronger than required in practice. A direct approximate version follows from the same trajectory bound.

Corollary B.8 (Approximate externalized state sufficiency)**.** *Suppose that at every jointly reachable common state,*

$$\mathrm{TV}\left(\pi_t(\cdot \mid Z_t, W_t), \widetilde{\pi}_t(\cdot \mid Z_t)\right) \le \eta_t, \qquad 0 \le \eta_t \le 1. \tag{B.36}$$

Then

$$\mathrm{TV}\left(\mathbb{P}_\pi, \mathbb{P}_{\widetilde{\pi}}\right) \le 1 - \prod_{t=1}^{K}(1-\eta_t) \le \min\left\{1, \sum_{t=1}^{K}\eta_t\right\}, \tag{B.37}$$

and for any $F \in [0,1]$,

$$\left|\mathbb{E}_\pi F - \mathbb{E}_{\widetilde{\pi}} F\right| \le \min\left\{1, \sum_{t=1}^{K}\eta_t\right\}. \tag{B.38}$$

Proof. Apply Theorem B.6 with $\epsilon_t = \eta_t$. □

Interpretation. Externalization need not make historical reasoning literally irrelevant. Rather, it is sufficient that the retained state Z_t makes the action law approximately insensitive to the removed reasoning W_t. The exact Corollary B.7 is recovered when $\eta_t = 0$ for every step. The controlled experiments in Section 4 examine local externalization effects, whereas η_t represents a uniform action-law bound over reachable common states and is not estimated in these experiments.

B.8 EXACT TOKEN ACCOUNTING AND TRAJECTORY AMPLIFICATION

Let

$$T^{\mathrm{b}} = \sum_{k=1}^{K_b}\left(I_k^{\mathrm{b}} + O_k^{\mathrm{b}}\right), \qquad T^{\mathrm{c}} = \sum_{k=1}^{K_c}\left(I_k^{\mathrm{c}} + O_k^{\mathrm{c}}\right) \tag{B.39}$$

be the processed token totals of the base and compressed trajectories, where I_k and O_k denote input and output tokens of request k.

Proposition B.9 (Trajectory-level token decomposition)**.** *Let $K_* = \min(K_b, K_c)$. Then*

$$\begin{aligned} T^{\mathrm{c}} - T^{\mathrm{b}} = &\sum_{k=1}^{K_*}\left[(I_k^{\mathrm{c}} - I_k^{\mathrm{b}}) + (O_k^{\mathrm{c}} - O_k^{\mathrm{b}})\right] \\ &+ \sum_{k=K_*+1}^{K_c}(I_k^{\mathrm{c}} + O_k^{\mathrm{c}}) - \sum_{k=K_*+1}^{K_b}(I_k^{\mathrm{b}} + O_k^{\mathrm{b}}). \end{aligned} \tag{B.40}$$

Proof. Split each total in Equation equation B.39 into the first K_* requests and its unmatched suffix, then subtract the two sums term by term. □

Interpretation. The first term in Equation equation B.40 measures changes within requests that both trajectories execute. The two suffix terms measure computation created or removed because the trajectories contain different numbers of requests. Therefore, direct text deletion at one boundary is only one component of the final system level token difference. Compression can change later reasoning, tool use, recovery, and termination, which in turn changes both later request sizes and the number of requests. This identity gives the accounting basis for trajectory amplification.

What the theory establishes. The results above serve four distinct roles. Lemma B.1 and Theorem B.2 motivate conditional uncertainty as an information limiting signal. Lemma B.3 and Corollary B.4 state when a proxy ranking would preserve a corresponding normalized population ordering. Proposition B.5 clarifies why predictive entropy and observed token surprisal are not interchangeable when scorer and actor differ. Finally, Theorem B.6 and Corollaries B.7 and B.8 characterize how local action law changes propagate through a trajectory. Together, these results describe conditional information, proxy ranking stability, and trajectory sensitivity under their stated assumptions, providing complementary mathematical perspectives on reasoning compression within the online agent interaction loop.

C Implementation and Online Compression Protocol

C.1 Runtime settings

Table 4: Recorded execution settings for the acting agent and frozen proxy scorer.

Component	Setting
Acting model	DeepSeek-V4-Flash
Thinking / reasoning effort	Enabled / `high`
Requested temperature	1
Requested maximum tokens	384,000
Streaming	Enabled
Frozen scorer	`acm-browsecompplus-qwen3.5-9b-opd-iter3`
Scorer execution	`eval()` and `inference_mode()`
Full context scorer limit	42,000 tokens
Deletion rule	Lowest $\lfloor 0.8N \rfloor$ nonempty reasoning blocks
Replacement text	Literal `[SKIP]`

Interpretation. The acting model generates actions and reasoning, while the frozen scorer is used only to compute the ranking signal. No scorer update is performed during evaluation. When the scorer input exceeds its 42K limit, pruning is skipped for that step rather than truncating the acting agent's history.

C.2 Reasoning extraction and entropy alignment

Reasoning is taken from `thinking` when available. The remaining fallbacks are `reasoning_content`, `reasoning`, and a `<think>...</think>` parser. Nonempty segments separated by `\n\n` define candidate blocks. Only reasoning spans are editable. Actions, tool arguments, tool outputs, observations, and final answers remain unchanged.

For scorer token x_t, let z_{t-1} denote the logits that predict that token. The implementation computes

$$\ell_t(v) = \mathrm{logsoftmax}(z_{t-1})_v, \tag{C.1}$$

$$h_t = -\frac{1}{\ln 2} \sum_{v \in \mathcal{V}} \exp(\ell_t(v)) \ell_t(v), \tag{C.2}$$

$$H(B_j) = |\mathcal{I}_j|^{-1} \sum_{t \in \mathcal{I}_j} h_t, \tag{C.3}$$

where $\mathcal{I}_j$ is the set of scorer token positions assigned to block B_j.

Interpretation. Equation equation C.3 is predictive entropy rather than observed token surprisal. The score is averaged within each block and used only for ranking candidate blocks. Appendix B.5 gives the corresponding entropy and surprisal decomposition and explains why the distinction matters when the frozen scorer and acting agent are different models.

C.3 Online write back

The first agent request is executed without compression. Before each subsequent request, the reasoning from the just completed interaction step is partitioned, scored, and rewritten in persistent history. All later requests consume this rewritten history rather than the original reasoning.

For N nonempty reasoning blocks, ICLR replaces the lowest $\lfloor 0.8N \rfloor$ blocks with `[SKIP]`. When $N = 1$, no block is removed. The nominal fraction is defined over blocks, so the realized token reduction varies with block length.

C.4 Intervention definitions

Table 5: Intervention families used in the ablation study.

Variant	Operation
Full history entropy	Score with the causal full context and replace the lowest 80% of reasoning blocks.
Step only entropy	Score the just completed reasoning without earlier trajectory context.
Delete all thinking	Remove the complete targeted reasoning while preserving actions and observations.
Random r	Remove a deterministic random subset of reasoning tokens for $r \in \{0.05, 0.10, 0.20, 0.40\}$.

Interpretation. The controls separate three questions. Delete all thinking tests wholesale removal, Random r isolates deletion without entropy ranking, and Step only entropy measures the contribution of earlier trajectory context to scoring. Random removal operates at token level, whereas ICLR operates at block level, so the percentages describe different intervention granularities.

D Evaluation Protocol

D.1 Evaluation sets

Table 6: Evaluation sets used for the reported system and representation analyses.

Set	Scope
Full benchmark	260 tasks: Code 80, Office 50, Security 60, Web 70
Pilot40	Fixed 40 task context management comparison
Ablation set	Fixed 80 tasks: 20 per domain
Representation cohort	265 usable boundary samples from 31 tasks
Patching cohort	47 samples from 8 tasks
Fixed state diagnostics	Controlled Office and error state continuations

Interpretation. These rows correspond to different evaluation units. Whole task results measure end to end agent behavior, while representation and fixed state analyses operate on local states sampled from trajectories. Each analysis is reported with its corresponding denominator.

D.2 METRICS AND TOKEN ACCOUNTING

For a consistently defined usage metric X, we report the relative change

$$\Delta_X = 100 \left(\frac{X_{\text{method}}}{X_{\text{base}}} - 1 \right). \tag{D.1}$$

The processed token total in the main tables is input plus output tokens. Cache read usage is reported separately rather than added again to input.

Task performance is measured by the official WorkBuddyBench reward. Domain means are reported together with the aggregate task result. Wins, ties, and losses compare the method and base system on the same benchmark tasks.

D.3 TASK LEVEL UNCERTAINTY

For the externalization analysis, reported confidence intervals use task level resampling so that multiple states from the same task remain grouped. If $\widehat{\theta}$ is a statistic computed from M task clusters, a percentile interval has the form

$$\left[Q_{0.025}(\widehat{\theta}^*), Q_{0.975}(\widehat{\theta}^*) \right], \tag{D.2}$$

where each bootstrap replicate resamples complete tasks with replacement. The appendix reports the recorded task level intervals for the externalization analysis.

E EXTENDED SYSTEM RESULTS

E.1 FULL BENCHMARK DOMAIN RESULTS

Table 7: Full260 domain results. Rewards are shown on the original $[0, 1]$ scale. Input and output columns report relative changes from the base system.

Domain	n	Base	ICLR	ΔR	ΔI (%)	ΔO (%)	W/T/L
Code	80	0.770000	0.706329	-0.063671	$+6.35$	$+16.21$	16/40/24
Office	50	0.818000	0.776152	-0.041848	-2.50	-2.55	16/8/26
Security	60	0.478000	0.706816	$+0.228816$	-35.23	-31.02	39/9/12
Web	70	0.721000	0.698551	-0.022449	-37.16	-19.04	31/19/20
Overall	260	0.698654	0.717775	$+0.019121$	-25.48	-14.41	102/76/82

Interpretation. The aggregate reward increases from 0.699 to 0.718, with the largest positive domain change appearing in Security. Code, Office, and Web show smaller domain scores than their base runs, and Code also uses more input. The complete 260 task aggregate combines these domain specific effects. The aggregate cache read reduction is 33.28%.

E.2 PILOT40 PRECISION

Table 8: Recorded Pilot40 precision for ICLR.

Code	Office	Security	Web	Mean score	Total $I + O$
74.10	81.06	64.11	62.00	70.32	80,812,679

Interpretation. The Pilot40 table corresponds to the fixed forty task context management comparison used in Table 1, while Full260 reports the complete benchmark evaluation.

F Ablation Details

F.1 Reward summary

Table 9: Ablation reward summary on the fixed 80 task set. Values correspond to the main ablation table and are shown on the original $[0, 1]$ scale.

Intervention	Code	Office	Security	Web	Average
Base Agent	0.6909	0.7894	0.4417	0.5850	0.6267
ICLR	0.6385	0.8114	0.7203	0.7800	0.7376
Delete All Thinking	0.6920	0.8204	0.6741	0.6850	0.7178
Random 5%	0.5981	0.8112	0.6417	0.6683	0.6798
Random 10%	0.5508	0.8265	0.7191	0.6061	0.6756
Random 20%	0.6723	0.7692	0.6482	0.6150	0.6762
Random 40%	0.6135	0.7617	0.6137	0.7162	0.6763
Step Only Entropy	0.6048	0.8058	0.7310	0.7038	0.7114

Interpretation. ICLR has the highest reported average reward in this ablation summary. Delete All Thinking also performs above the displayed base average, showing that the history contains substantial removable redundancy. Step Only Entropy remains competitive but below full history entropy, which is consistent with the use of earlier trajectory context in the scoring rule. The random controls show that deletion itself can change task performance, with outcomes depending on more than the configured random removal percentage.

F.2 Token reduction and trajectory amplification

Table 10: Input and output reductions used in the trajectory amplification analysis.

Intervention	Input reduction (%)	Output reduction (%)
ICLR	17.03	12.05
Delete All Thinking	63.59	13.55
Random 5%	53.68	38.46
Random 10%	60.53	46.74
Random 20%	44.01	26.35
Random 40%	51.21	34.22
Step Only Entropy	67.88	38.87

Interpretation. Input reduction varies nonmonotonically with the configured local deletion ratio. Random 10% reduces input by 60.53%, while Random 20% reduces input by 44.01%. This is the empirical pattern referred to as trajectory amplification. Proposition B.9 explains this behavior through changes to subsequent requests as well as the current serialized history.

Relation to Figure 3. The main figure visualizes the mismatch between local intervention strength and whole trajectory consequences, showing that final token and reward outcomes depend on the trajectory induced after the intervention.

G Hidden State Probing

G.1 Cohort and target

A frozen Qwen3.5 9B model reads the context visible at each compression boundary. For sample i, layer l, and pooling rule p,

$$z_i^{(l,p)} = \mathrm{Pool}_p\left(\mathcal{Z}_i^{(l)}\right), \qquad \widehat{p}_i = \sigma\left(w^\top z_i^{(l,p)} + b\right). \tag{G.1}$$

The behavioral target records whether the later trajectory contains rederivation, additional reasoning, or explicit replanning and serves as an operational measure of future reasoning reuse.

Table 11: Representation extraction cohort used for future necessity probing.

Quantity	Value
Usable boundary samples	265
Distinct tasks	31
Positive behavioral labels	112
Transformer layers	32
Pooling candidates	5
Outer task folds	5

Interpretation. The unit of analysis is a compression boundary rather than a complete benchmark task. Multiple boundary samples can come from the same task, which is why model selection and evaluation are separated at the task level.

G.2 Nested task level evaluation

Layer selection, pooling choice, feature standardization, and PCA are chosen using training tasks only. The outer folds contain held out tasks. This design prevents samples from the same task from appearing on both sides of the outer evaluation split.

Table 12: Strict nested task level probe results.

AUROC	AP	Balanced accuracy	Selected pooling
0.8444	0.8108	0.7597	`prompt_last`

Interpretation. The probe results show that the current hidden state contains a readable cross task signal associated with later reasoning reuse. The linear probe serves as an analysis instrument for characterizing that signal.

H Activation Patching

Figure interpretation. The curve summarizes a local intervention on the frozen representation sensor. Larger KL Repair means that the patched representation produces a next token distribution closer to the full history condition. Recovery increases toward later layers, showing that full history information is progressively expressed in representations closer to the output readout.

H.1 Intervention and measurement

The patching analysis uses 47 samples from 8 tasks on the frozen Qwen3.5 9B sensor. For sample i, let P_i^{full} be the next token distribution under full history, P_i^{del} the distribution after historical reasoning is removed, and $P_{i,l}^{\text{patch}}$ the distribution obtained after replacing the deleted history prompt boundary residual at layer l with the corresponding full history residual.

Define

$$d_i = D_{\text{KL}}\left(P_i^{\text{full}} \| P_i^{\text{del}}\right). \tag{H.1}$$

For $d_i > 0$, KL Repair is

$$\text{Repair}_i(l) = 1 - \frac{D_{\text{KL}}\left(P_i^{\text{full}} \| P_{i,l}^{\text{patch}}\right)}{d_i}. \tag{H.2}$$

A value near one indicates strong distributional restoration relative to the deleted history condition.

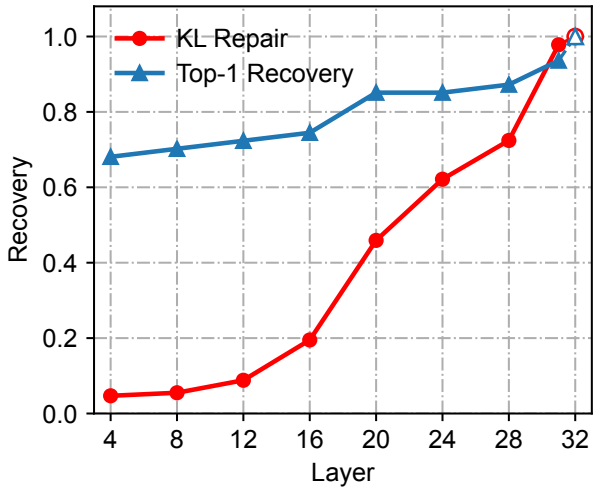


Figure 4: **Activation patching across depth.** Full history residuals progressively restore the perturbed next token distribution when they replace the corresponding deleted history residual at the prompt boundary.

Table 13: Activation patching results on the 47 sample cohort.

Layer	Mean KL Repair	95% CI	Top 1 agreement
4	0.047	[0.027, 0.076]	0.681
16	0.195	[0.170, 0.235]	0.745
20	0.459	[0.392, 0.537]	0.851
24	0.622	[0.545, 0.700]	0.851
28	0.724	[0.667, 0.788]	0.872
31	0.978	[0.964, 0.986]	0.936
32	1.000	[1.000, 1.000]	1.000

H.2 Layerwise results

Interpretation. Repair rises substantially from early to late layers, matching the main text summary at layers 4, 20, 28, and 31. Top 1 agreement increases in the same direction. Layer 32 is an implementation positive control because patching the final residual immediately before a deterministic readout restores the full history readout input.

Lemma H.1 (Final layer patching control)**.** *Suppose that the final residual h_L is followed by the same deterministic readout map g in the full and deleted history conditions. If the deleted run is patched with h_L^{full}, then*

$$P_{i,L}^{\text{patch}} = P_i^{\text{full}}. \tag{H.3}$$

Whenever $d_i > 0$, this implies $\text{Repair}_i(L) = 1$.

Proof. After patching, both conditions pass the same residual h_L^{full} through the same deterministic map g. Their resulting next token distributions are therefore identical. Substituting the equality into Equation equation H.2 gives zero numerator and repair equal to one. □

I Externalization and Fixed State Diagnostics

I.1 Internalization and externalization gap

Let $\mathcal{U}_i$ denote the unexternalized task specific derived state at compression boundary i. We define

$$G_i = \mathbb{I}\left(\mathcal{U}_i \neq \varnothing\right). \tag{I.1}$$

Derived state includes intermediate plans, relations, constraints, syntheses, or task specific conclusions that remain internal to reasoning before materialization in a persistent external carrier.

Table 14: Grouping by the internalization and externalization gap.

Group	n	Future risk	Mean hidden score
No gap	213	36.15%	0.4369
Gap present	52	67.31%	0.6662

Interpretation. Future behavioral risk is higher when the relevant derived state remains available only in reasoning. The hidden score shows the same ordering. These pooled values are descriptive and motivate the within task comparison below.

Table 15: Within task contrasts for gap present minus no gap states.

Quantity	Mean difference	Task bootstrap 95% CI
Future behavioral risk	+19.75 pp	[1.00, 37.05]
Hidden score	+0.2831	[0.1703, 0.3977]

Interpretation. The within task contrasts preserve the same direction after comparing states from the same tasks, strengthening the state based interpretation in the main text.

I.2 Controlled state interventions

Table 16: Controlled state interventions. The three groups probe where useful task state is stored.

Intervention	Rule score	Reasoning	New requests
Fixed intermediate Office state			
Keep	0.8717	1,817	5.00
Delete all	0.8146	7,491	5.33
Random 5%	0.8756	516	4.00
Random 10%	0.8756	562	7.33
Position intervention			
Keep all	0.8717	–	–
Delete all	0.6749	–	–
Keep latest only	0.8717	–	–
Delete latest only	0.7598	–	–
After code is written to disk			
Keep thinking	0.7587	1,289	–
Clear thinking	0.7587	3,222	–

Interpretation. The fixed intermediate state shows that complete deletion can trigger substantially more reasoning while lowering the rule score. The position intervention shows that keeping only the latest reasoning matches the keep all score, whereas deleting the latest reasoning causes a larger drop. After relevant code has been written to disk, clearing subsequent reasoning leaves the rule score unchanged at 0.7587. Together, these local interventions support the claim that the importance of reasoning depends on whether task relevant state has already been externalized.

I.3 Environmental feedback intervention

Interpretation. When the error remains visible, the agent can act directly on the observed state. When the same feedback is hidden, the agent first performs another Bash call and recovers the same error. This provides a concrete example in which environmental feedback functions as an external information carrier.

Table 17: Next action after retaining or hiding the already observed `stderr`.

Feedback	Next action	Count	Subsequent observation
`stderr` visible	Edit	9/9	No repeated error observation
`stderr` hidden	Bash	9/9	The same `ValueError` appears again

J Motivation Figure Details

Figure 1 is used to motivate the state dependent value of reasoning. Panels (a) and (b) compare task score and token usage across different reasoning settings and difficulty levels. Panel (c) summarizes the empirical reward and input reduction tradeoff under online interventions.

Interpretation. The figure highlights two motivating patterns. First, the relation between reasoning intensity and task score varies across difficulty levels. Second, the relation between local deletion strength and end to end savings varies across interventions. Appendix F provides the corresponding intervention results.

K Main Text to Appendix Map

Table 18: Correspondence between major main text claims and supplementary evidence.

Main text claim	Supplementary location	Supporting material
Online compression changes future interaction	Appendix B.6	Finite horizon perturbation theorem
Low entropy ranking is an information motivated heuristic	Appendices B.2 to B.5	Information bounds, proxy ranking stability, and entropy-surprisal distinction
Local deletion and total computation differ	Appendix B.8	Exact trajectory token decomposition
Full benchmark result	Appendix E.1	Domain rewards, token changes, and W/T/L
Trajectory amplification	Appendix F.2	Intervention specific input and output reductions
Future necessity is decodable	Appendix G	Cohort, nested protocol, and probe metrics
Historical reasoning affects representations	Appendix H	Activation patching curve and layerwise results
Externalization changes reasoning usefulness	Appendices B.7 and I	Exact/approximate sufficiency conditions, gap analysis, and controlled interventions

Interpretation. The table organizes the paper's supporting evidence into system level benchmark results, local mechanistic diagnostics, and mathematical analysis.

L Broader Impact and Future Directions

Reducing repeated processing of agent history lowers active context usage and can improve inference efficiency. ICLR preserves actions, tool calls, and observations while selectively compressing reasoning, keeping externally recorded task state available throughout interaction.

Several aspects of the current study motivate future work. Proxy entropy is computed by a frozen model distinct from the acting agent, motivating tighter alignment between scorer and policy representations. The future necessity target is behavioral, suggesting richer labels for different forms of reasoning reuse. Activation patching characterizes the frozen representation sensor, while future

studies can extend the same analysis to additional acting models and longer closed loop interventions. Externalization analysis also motivates explicit modeling of when code, files, tool outputs, and observations become sufficient carriers of task relevant derived state.

Raw trajectory access should respect the confidentiality of prompts, credentials, tool outputs, and generated artifacts.

M TASK MANIFEST

M.1 TASK SET SUMMARY

Table 19: Task set composition used in the main experiments.

Set	Code	Office	Security	Web	Total
Full benchmark	80	50	60	70	260
Ablation set	20	20	20	20	80
Pilot40	Fixed forty task comparison				40

Interpretation. The Full260, fixed 80 task ablation set, and Pilot40 comparison provide complementary views of complete benchmark performance, controlled ablation behavior, and context management comparison.