Beyond Lost in the Middle: Causal Locality in Tool-Using Agent Contexts

Anonymous authors
Paper under double-blind review

Abstract

Long-context evaluations usually move a relevant document while treating the remaining input as exchangeable. Agent histories are different: a tool call and its result form an atomic transaction, and later transactions can depend on values or decisions produced earlier. Reordering an agent context may therefore gain favorable absolute position while destroying causal locality. We introduce a paired intervention testbed that varies the producer position and the amount of unrelated evidence inserted inside a critical dependency chain, while holding the evidence multiset fixed and preserving a valid topological order. Across 24,144 greedy-decoded calls to two open-weight model artifacts, the separation contrast is -4.40 percentage points for GPT-OSS 20B, whereas Gemma 3 27B has no changed binary outcomes in the paired cells. Their equal-weight arithmetic mean is -2.20 points (95% trajectory-bootstrap CI: $[-4.05, -0.23]$), but is not a cross-model population estimate. Because the original separation treatment also moves the consumer endpoint, this pilot does not identify a position-independent locality coefficient. We find no reliable front-versus-middle benefit or pre-registered interaction. An exploratory end-pinning heuristic improves GPT-OSS by 4.63 points over chronological order, but has no effect on Gemma. These results expose capability, endpoint-position, and run-reliability checks that are necessary before asserting a universal agentic "lost-in-the-middle" rule or deploying an optimized placement policy.

1 Introduction

Language-model agents accumulate an unusual kind of long context. A trajectory interleaves instructions, model actions, tool calls, environment observations, errors, and state updates. ReAct made this interleaving a standard agent interface (Yao et al., 2023), and contemporary evaluations show that models still struggle with long-horizon decisions and tool use (Liu et al., 2024b; Trivedi et al., 2024; Yao et al., 2025). As histories grow, an agent runtime must decide what to retain, summarize, retrieve, and where to place it.

The best-known positional result comes from document question answering: models often use relevant evidence more successfully near the beginning or end than in the middle of a long prompt (Liu et al., 2024a). Subsequent benchmarks expand long-context evaluation to more tasks and longer inputs (Bai et al., 2024; Hsieh et al., 2024), and recent work continues to find position-sensitive behavior (Byerly & Khashabi, 2026). Applying this result directly to agents suggests moving important events to a context edge. That prescription ignores structure. A tool result can introduce an identifier that a later call consumes; an error can license a retry; a state refresh can supersede an earlier observation; and an authorization event can constrain a later action. Moving every individually relevant event toward an edge can separate transactions that must be interpreted together.

This paper asks: what is the behavioral tradeoff between absolute position and causal locality in a legally reordered agent trajectory? We study this question using paired prompt interventions. The evidence multiset, task, nonce values, and nominal prompt size are fixed within a base trajectory. We vary only (i) the normalized position of the first critical transaction and (ii) the normalized number of unrelated transactions placed inside the crit-

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.

ical dependency chain. Every tool call remains adjacent to its result, and every displayed sequence is a topological ordering of a declared dependency graph.

Our contributions are:

- We formalize legal agent-context placement: atomic call–result transactions may move, but their evidence multiset, internal atomicity, and declared causal order must be preserved. This separates an agent-specific structural question from unconstrained document shuffling.
- We introduce a paired factorial testbed spanning six kinds of causal dependence, six absolute positions, four dependency gaps, three context sizes, and two distractor regimes. It produces case-level outputs, programmatic exact-action scores, and trajectory-clustered intervals.
- We report a deliberately non-universal diagnostic. A separation effect appears in GPT-OSS but Gemma supplies no discordant paired outcomes; moreover, the original treatment moves the consumer endpoint. These facts motivate an endpoint-level response surface, matched controls, capability calibration, and repeated-run confirmation rather than a pooled rule.

This is a controlled diagnostic study, not yet a claim about arbitrary production agents. We make that boundary explicit because the negative and heterogeneous results are central to the scientific conclusion.

2 Related Work

Long-context position. Lost in the Middle varies the location of an answer-bearing document in multi-document QA and key–value retrieval, revealing a frequently U-shaped response curve (Liu et al., 2024a). LongBench broadens long-context evaluation across 21 datasets (Bai et al., 2024); RULER adds configurable retrieval, multi-hop tracing, and aggregation tests (Hsieh et al., 2024). These studies establish that nominal context length is not effective context use. They do not, however, model a trajectory whose units have call–result atomicity and directed dependencies. Our treatment keeps document content constant but intervenes on two distinct coordinates: absolute chain location and separation along a dependency edge.

Agent and multi-turn evaluation. AgentBench evaluates models in eight interactive environments (Liu et al., 2024b); AppWorld supplies stateful applications and programmatic end-state checks (Trivedi et al., 2024); and τ-bench tests tool–agent–user interactions under domain policies (Yao et al., 2025). ToolSandbox adds stateful tools, implicit dependencies, environment snapshots, and milestone-based evaluation (Lu et al., 2025). LLMs Get Lost in Multi-Turn Conversation controls whether an instruction is supplied at once or sharded over turns, finding a large reliability gap (Laban et al., 2026). This is close in spirit because information is dispersed through a history, but the intervention changes conversational disclosure rather than the legal placement of completed tool transactions. We evaluate the next action after the same completed evidence has been assembled.

Agent context management. Recent systems compress observations and histories, optimize natural-language compression policies, or make compression an explicit agent action (Kang et al., 2025; Liu et al., 2025). ACM lets an agent edit context, offload content, and retrieve it later (Li et al., 2026); CWL evicts typed, dependency-linked episodes (Semenov & Dorofeev, 2026); and TRACE evaluates compaction events by paired continuation from a common environment state (Min et al., 2026). Agent Context Compilation instead converts tool trajectories into long-context training examples (Su et al., 2026). These methods motivate the engineering importance of agent histories, but change the evidence set, representation, timing, or model weights. To our knowledge, existing work has not independently randomized absolute position and dependency separation over valid permutations of an exactly fixed agent evidence multiset. Our contribution is this placement-specific causal diagnostic, rather than a new compression or eviction mechanism.

3 Legal Agent-Context Placement

Let a completed context consist of atomic blocks $B = \{b_1, \ldots, b_n\}$. A tool transaction is one block containing a tool call and its corresponding result; the two are never separated. A directed acyclic graph $G = (B, E)$ represents declared dependencies. An edge (b_i, b_j) means that b_j consumes a value, state, decision, or permission established by b_i. A layout is a permutation π of B. It is legal iff

$$\Pi(G) = \{\pi : \pi(i) < \pi(j) \text{ for every } (b_i, b_j) \in E\}. \tag{1}$$

For a critical chain $C = (c_1, \ldots, c_k)$, define producer position

$$p(\pi) = \frac{\pi(c_1)}{n-1} \tag{2}$$

and noncritical dependency gap

$$g(\pi) = \frac{\pi(c_k) - \pi(c_1) - (k-1)}{n-1}. \tag{3}$$

The first coordinate asks where critical evidence begins; the second asks how many irrelevant atomic transactions interrupt the dependency chain. A layout generator assigns critical blocks to requested (p, g) cells, fills remaining slots with the same distractor blocks, rejects boundary-infeasible cells rather than clipping them, and verifies membership in $\Pi(G)$.

This formulation exposes a constrained placement objective. Given block relevance r_i and weights λ_p, λ_g, our exploratory DAGPlace baseline maximizes

$$J(\pi) = \lambda_p \sum_i r_i \, 2 \left| \frac{\pi(i)}{n-1} - \frac{1}{2} \right| - \lambda_g \sum_{(i,j) \in E} \frac{\max(0, \pi(j) - \pi(i) - 1)}{n-1}, \quad \pi \in \Pi(G). \tag{4}$$

We use deterministic multi-start legal local search. The present experiment tests whether this intuitive objective tracks model behavior; it is not trained on evaluation outcomes.

4 Experimental Design

4.1 Procedural task families

Each task contains random nonce identifiers, a natural-language goal, one critical dependency chain, and irrelevant tool transactions. The correct output is exactly one JSON next action. The six families instantiate different edge semantics:

1. Parameter propagation: resolve a customer and eligible account, then issue a refund using the discovered account identifier.
2. Conditional branching: inspect an object and apply a routing policy, then execute the returned branch decision.
3. Multi-hop data flow: resolve owner, workspace, and archive key before acting.
4. Error recovery: observe a schema error, verify a replacement identifier, and retry with the corrected argument.
5. State update: read a state, receive a newer state, and choose the action licensed by the latest value.
6. Authorization: combine an authorization event with a validation result before executing the scoped action.

We generate 12 instances for every family × block count × distractor-strength cell. Block counts are $n \in \{32, 64, 128\}$. Lexical distractors resemble historical tool logs without competing identifiers; "conflict" distractors contain plausible but unrelated identifiers and a negative approval field. Every layout of a base instance contains exactly the same strings. Section 7 discusses an important limitation: the conflict schema is generic rather than tailored to each task family.

4.2 Treatments and outcomes

Requested producer positions are $p \in \{.05, .20, .40, .60, .80, .93\}$ and gaps are $g \in \{0, .03, .10, .20\}$. Boundary-infeasible pairs are omitted, leaving 20 or 21 controlled layouts per model–base trajectory. The three pre-registered contrasts are:

$$\tau_p = Y(p = .05, g = 0) - Y(p = .40, g = 0), \tag{5}$$

$$\tau_g = Y(p = .40, g = .20) - Y(p = .40, g = 0), \tag{6}$$

$$\tau_{p\times g} = [Y(.05, .20) - Y(.40, .20)] - [Y(.05, 0) - Y(.40, 0)]. \tag{7}$$

The original registration called τ_p an edge–middle contrast. Because .05 is farther from .5 than .93 in the implemented grid, the code selects the front edge only. We therefore report it more precisely as front–middle; no symmetric edge claim is made.

The primary endpoint Y is exact next-action success: tool name and the entire argument dictionary must match. Secondary outcomes are JSON validity, tool-name accuracy, and argument-field F1. Invalid JSON is a behavioral failure; backend errors are tabulated separately. We also compare seven legal layouts: chronological, random topological order, relevance-first, front-pinning, end-pinning, edge-spreading, and DAGPlace.

4.3 Models, inference, and computation

We evaluate local Ollama tags gemma3:27b (Gemma Team, 2025) and gpt-oss:20b (OpenAI, 2025). The recorded model IDs are a418f5838eaf and 17052f91a42e; underlying blob digests are included in the supplement. Inference uses Ollama 0.32.13, temperature 0, a 32,768-token runtime context, at most 512 generated tokens, and the model's default reasoning mode. Four RTX 3090 GPUs run two shards per model. Observed prompt-token counts range from 5,737 to 30,767 across model tokenizers and conditions. All 24,144 requests completed without a backend error.

4.4 Inference and estimands

The independent unit is a procedural base trajectory, not a layout. Treatment outcomes are paired within the base trajectory. We estimate each contrast per model–base trajectory and average those contrasts; the pooled estimate is model-balanced because each model has 432 bases. Ninety-five percent intervals come from 5,000 nonparametric bootstrap resamples of whole base trajectories with fixed seed 20260903. We pre-specified model, task-family, block-count, and distractor-strength heterogeneity. Baseline differences are likewise paired against the same base trajectory's chronological layout. Comparisons that select the best heuristic after observing its mean are explicitly exploratory.

The pre-registration required all three primary effects to be measurable in at least two model families and required DAGPlace to beat the strongest legal heuristic. The results do not satisfy this compound success criterion.

5 Results

5.1 The separation contrast appears in one model artifact

Table 1 reports the paired primary contrasts. GPT-OSS loses 4.40 percentage points (pp) when the requested separation increases from zero to 20%, whereas Gemma exhibits no

Table 1: Pre-registered paired effects on exact next-action success. Entries are percentage-point differences with 95% trajectory-bootstrap intervals. Negative far–near values mean that separating dependent transactions hurts.

Subgroup	N	Front−middle	Far−near	Interaction
Pooled	864	-0.69 [-2.78, +1.27]	-2.20 [-4.05, -0.23]	+1.50 [-1.04, +3.94]
Gemma 3 27B	432	-0.23 [-0.69, +0.00]	+0.00 [+0.00, +0.00]	+0.23 [+0.00, +0.69]
GPT-OSS 20B	432	-1.16 [-5.09, +2.78]	-4.40 [-8.10, -0.69]	+2.78 [-2.31, +7.64]

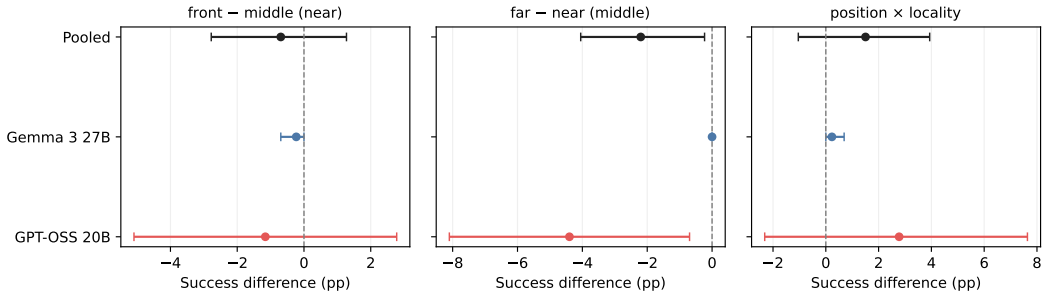


Figure 1: Paired primary effects. Points are mean differences in exact next-action success; bars are 95% nonparametric bootstrap intervals over base trajectories. The arithmetic-mean separation interval excludes zero, but Gemma contributes no changed paired outcomes and GPT-OSS supplies the entire directional signal.

changed binary outcomes in these paired cells. The equal-weight arithmetic mean is −2.20 pp and its trajectory interval excludes zero, but one of the two model artifacts contributes no discordant pairs. We therefore do not interpret the mean as evidence of a model-invariant effect. In addition, holding the producer location fixed while increasing separation necessarily moves the consumer endpoint; the contrast is a joint layout intervention rather than a pure locality coefficient.

The front–middle contrast at zero gap is −0.69 pp pooled, with an interval spanning zero. This rules out neither an end-edge advantage nor a symmetric edge effect; both must be computed from the existing end-position cells. The pre-registered front-position–separation interaction is +1.50 pp and also uncertain. Thus, the current analysis does not establish the position profile promised by a generic lost-in-the-middle claim.

The pre-specified distractor analysis further qualifies the locality result. The far–near contrast is −4.40 pp ($[-7.18, -1.85]$) under lexical distractors, but 0.00 pp ($[-2.55, 2.55]$) under generic conflict distractors. Context length also does not show a monotone gradient: estimated locality effects are −3.47, −3.47, and +0.35 pp at 32, 64, and 128 blocks, respectively, with all three length-specific intervals touching or crossing zero. These subgroup results were pre-specified but are not adjusted for multiplicity; we use them to diagnose heterogeneity, not to manufacture an additional confirmatory claim.

5.2 Placement policies are model dependent

Table 2 compares legal placement policies. For Gemma, all deterministic heuristics except random legal ordering produce the same aggregate exact score as chronological ordering. For GPT-OSS, end-pinning improves exact success from 68.75% to 73.38%, a paired difference of +4.63 pp ($[+1.15, +8.33]$). This comparison is exploratory because the study was not registered to privilege end-pinning. It is nonetheless a useful hypothesis for confirmation: keeping the critical chain adjacent at the generation-side edge may help this model.

DAGPlace reaches 71.06% on GPT-OSS, a non-significant +2.31 pp over chronological order, and 69.44% on Gemma. Pooled across models, the strongest observed non-optimizer heuristic is end-pinning at 71.41%, while DAGPlace reaches 70.25%; the paired DAGPlace-

Table 2: Legal placement baselines. Intervals are paired against chronological order within the same base trajectory. Heuristic selection is exploratory.

Model	Placement	Exact (%)	Δ vs. chrono. [95% CI] (pp)
Gemma 3 27B	Chronological	69.44	—
	Random legal	69.21	-0.23 [-0.69, +0.00]
	Relevance first	69.44	+0.00 [+0.00, +0.00]
	Pin front	69.44	+0.00 [+0.00, +0.00]
	Pin end	69.44	+0.00 [+0.00, +0.00]
	Edge spread	69.44	+0.00 [+0.00, +0.00]
	DAGPlace	69.44	+0.00 [+0.00, +0.00]
GPT-OSS 20B	Chronological	68.75	—
	Random legal	67.13	-1.62 [-6.02, +2.78]
	Relevance first	70.37	+1.62 [-2.08, +5.32]
	Pin front	68.98	+0.23 [-3.70, +4.17]
	Pin end	73.38	+4.63 [+1.15, +8.33]
	Edge spread	68.06	-0.69 [-4.63, +3.24]
	DAGPlace	71.06	+2.31 [-1.62, +6.02]

Table 3: Outcome decomposition. All values except N are percentages.

Model	Split	N	Valid JSON	Tool	Arg. F1	Exact
Gemma 3 27B	Baselines	3024	97.65	69.94	69.94	69.41
Gemma 3 27B	Controlled	9048	99.64	70.65	70.01	69.35
GPT-OSS 20B	Baselines	3024	85.25	79.63	74.66	69.68
GPT-OSS 20B	Controlled	9048	84.98	78.05	73.49	68.93

minus-selected difference is -1.16 pp ($[-3.01, +0.69]$). Consequently, the optimization success criterion fails. The fixed objective in Section 3 is at best a prior over layouts; these data do not validate its weights as a universal placement rule.

5.3 Failure modes reveal two distinct regimes

JSON validity and semantic correctness differ sharply by model (Table 3). GPT-OSS returns valid JSON on only 84.98% of controlled calls, yet its tool-name accuracy is 78.05%; enforcing structured decoding could therefore change absolute accuracy and possibly treatment sensitivity. Gemma's controlled JSON validity is 99.64%, but its overall exact score remains 69.35%.

Gemma exposes a more consequential regime change. On 32- and 64-block controlled contexts, it solves five task families almost perfectly and fails the state-update family almost completely. At 128 blocks, lexical distractors preserve that pattern, but generic conflict distractors reduce exact success to zero across all six families. Manual inspection shows well-formed responses that copy or act on irrelevant distractor records, rather than transport or parser errors. This is a real behavioral failure under the present generator, but the all-or-none structure makes it unsuitable as clean evidence for a smooth position–locality curve. It motivates task-matched distractors and capability-calibrated instances in the confirmatory design.

6 Discussion

Position is not a portable policy. The simplest extrapolation from document QA—move relevant evidence to an edge—does not hold uniformly here. We detect no front-edge benefit in the paired primary contrast, and legal heuristics behave differently across the two models. The result does not contradict document-level positional bias; it shows that its engineering consequence depends on the structure and model.

Separation effects are path specific. The original design varies producer position and inserted dependency separation while preserving the same evidence, but increasing separation also moves the consumer endpoint. The GPT-OSS result is therefore evidence that this joint legal layout intervention can change the selected next action, not that separation has a position-independent coefficient. The confirmatory design estimates a two-endpoint surface and matched translation/dilation paths before drawing a locality interpretation.

A conservative design rule. The current evidence supports a low-risk invariant, not a universal optimizer: keep a tool call adjacent to its result, preserve topological order, and avoid separating known producer–consumer chains unless a model-specific validation shows a benefit. Edge placement should be calibrated per model and prompt format. Structured decoding should be evaluated separately from layout so that format failures do not obscure evidence-use failures.

What would establish the stronger claim? A publishable general placement policy requires: symmetric front and end treatments; multiple current local and frontier models; task-matched distractors; repeated decoding seeds; closed-loop success on real trajectories; and held-out tuning of objective weights. Section G freezes this next-stage plan. The present data remain useful as the internally controlled first study and as evidence against overgeneralization.

7 Limitations

First, the study evaluates two locally served, quantized model artifacts; one model supplies the entire negative locality signal. Frontier API models and additional architectures are absent. Second, tasks are procedural and measure one exact next action, not end-to-end recovery, planning, or environment state. Third, conflict distractors reuse customer/account-like fields across all task families. Their semantics are therefore not uniformly matched, and at 128 blocks they induce an extreme Gemma collapse that can mask marginal layout effects. Fourth, several families are near a ceiling or floor, especially for Gemma, reducing treatment sensitivity. Fifth, each model is decoded once per layout at temperature zero; the bootstrap represents trajectory sampling, not generation stochasticity. Sixth, the implemented primary "edge" contrast is front-versus-middle rather than a symmetric edge average. Seventh, exploratory baseline and subgroup comparisons are not multiplicity corrected. Finally, token counts differ across model tokenizers, and no attention or internal-state measure is used; all causal language refers only to randomized prompt-layout effects on observed behavior.

8 Conclusion

Agent-context placement is not ordinary document ranking. Legal reordering must preserve atomic tool transactions and the directed dependencies that make later actions interpretable. In the completed 24,144-call pilot, the joint producer-fixed separation intervention changes GPT-OSS outcomes but leaves all paired Gemma outcomes unchanged; the equal-weight mean is therefore not a cross-model law. The study also tests only a front–middle primary position contrast and does not establish a symmetric edge curve. Its durable result is a clean fixed-evidence testbed plus a diagnosis: endpoint matching, capability calibration, controls, and repeated runs are required before a locality mechanism or universal placement policy can be asserted.

AI use statement

Generative AI tools were used in developing parts of the experimental design, implementing and debugging the research software, constructing statistical analysis scripts and figures, discovering candidate literature, and drafting and editing the manuscript. OpenAI Codex was the principal assisted tool. The human authors initiated the research question and are responsible for checking the experimental specification, inspecting raw outputs, validating reported statistics against deterministic scripts, verifying citations against primary sources, and approving all claims. Evaluated model outputs were scored by programmatic rules rather than an AI judge. We take responsibility for all AI-assisted text, code, claims, and artifacts in this work.

Ethics statement

The benchmark is procedurally generated and contains no human-subject or personal data. Tool actions are simulated strings and cause no external state changes. Better context placement could improve agent reliability, but it could also make unsafe automated actions more effective. The study therefore does not justify autonomous deployment; authorization scope, tool permissions, and independent safety checks remain necessary. We report null results, failed pre-registered criteria, and generator artifacts to reduce the risk of misleading downstream use.

Reproducibility statement

The supplementary repository contains the pre-registration, frozen YAML configuration, procedural generator, legal-layout verifier, exact scoring code, append-only raw JSONL responses, deduplicated case table, model and runtime identifiers, and a single script that regenerates every reported table, interval, and figure. Appendix F lists the commands and integrity checks. Intervals use a fixed seed and resample base trajectories, never individual layout rows. An anonymized archival copy should accompany submission.

References

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. LongBench: A bilingual, multitask benchmark for long context understanding. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 3119–3137, 2024. doi: 10.18653/v1/2024.acl-long.172. URL https://aclanthology.org/2024.acl-long.172/.

Adam Byerly and Daniel Khashabi. Self-consistency falls short! the adverse effects of positional bias on long-context problems. Transactions of the Association for Computational Linguistics, 14:292–317, 2026. doi: 10.1162/tacl.a.625. URL https://aclanthology.org/2026.tacl-1.15/.

Gemma Team. Gemma 3 technical report. arXiv preprint arXiv:2503.19786, 2025. doi: 10.48550/arXiv.2503.19786. URL https://arxiv.org/abs/2503.19786.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. RULER: What's the real context size of your long-context language models? In Conference on Language Modeling, 2024. doi: 10.48550/arXiv.2404.06654. URL https://openreview.net/forum?id=kIoBbc76Sy.

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. doi: 10.48550/arXiv.2510.00615. URL https://arxiv.org/abs/2510.00615.

Philippe Laban, Hiroaki Hayashi, Yingbo Zhou, and Jennifer Neville. LLMs get lost in multi-turn conversation. In International Conference on Learning Representations,

pp. 54738–54778, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/59f6421e64707225fdf5b28840679a07-Paper-Conference.pdf.

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, 2026. doi: 10.48550/arXiv.2607.23809. URL https://arxiv.org/abs/2607.23809.

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, 2024a. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9/.

Shukai Liu, Jian Yang, Bo Jiang, Yizhi Li, Jinyang Guo, Xianglong Liu, and Bryan Dai. Context as a tool: Context management for long-horizon SWE-agents. arXiv preprint arXiv:2512.22087, 2025. doi: 10.48550/arXiv.2512.22087. URL https://arxiv.org/abs/2512.22087.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, Shudan Zhang, Xiang Deng, Aohan Zeng, Zhengxiao Du, Chenhui Zhang, Sheng Shen, Tianjun Zhang, Yu Su, Huan Sun, Minlie Huang, Yuxiao Dong, and Jie Tang. AgentBench: Evaluating LLMs as agents. In International Conference on Learning Representations, pp. 52989–53046, 2024b. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/e9df36b21ff4ee211a8b71ee8b7e9f57-Paper-Conference.pdf.

Jiarui Lu, Thomas Holleis, Yizhe Zhang, Bernhard Aumayer, Feng Nan, Haoping Bai, Shuang Ma, Shen Ma, Mengyu Li, Guoli Yin, Zirui Wang, and Ruoming Pang. ToolSandbox: A stateful, conversational, interactive evaluation benchmark for LLM tool use capabilities. In Findings of the Association for Computational Linguistics: NAACL 2025, pp. 1160–1183, 2025. doi: 10.18653/v1/2025.findings-naacl.65. URL https://aclanthology.org/2025.findings-naacl.65/.

Guanghui Min, Liang Wu, Mayank Darbari, Chen Chen, and Liangjie Hong. Toward reliable context compression for long-horizon agents: An empirical study of execution instability. arXiv preprint arXiv:2608.06503, 2026. doi: 10.48550/arXiv.2608.06503. URL https://arxiv.org/abs/2608.06503.

OpenAI. gpt-oss-120b & gpt-oss-20b model card. arXiv preprint arXiv:2508.10925, 2025. doi: 10.48550/arXiv.2508.10925. URL https://arxiv.org/abs/2508.10925.

Andrew Semenov and Svyatoslav Dorofeev. Beyond compaction: Structured context eviction for long-horizon agents. arXiv preprint arXiv:2606.11213, 2026. doi: 10.48550/arXiv.2606.11213. URL https://arxiv.org/abs/2606.11213.

Qisheng Su, Zhen Fang, Shiting Huang, Yu Zeng, Yiming Zhao, Kou Shi, Ziao Zhang, Lin Chen, Zehui Chen, Lijun Wu, and Feng Zhao. ACC: Compiling agent trajectories for long-context training. arXiv preprint arXiv:2605.21850, 2026. doi: 10.48550/arXiv.2605.21850. URL https://arxiv.org/abs/2605.21850.

Harsh Trivedi, Tushar Khot, Mareike Hartmann, Ruskin Manku, Vinty Dong, Edward Li, Shashank Gupta, Ashish Sabharwal, and Niranjan Balasubramanian. AppWorld: A controllable world of apps and people for benchmarking interactive coding agents. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 16022–16076, 2024. doi: 10.18653/v1/2024.acl-long.850. URL https://aclanthology.org/2024.acl-long.850/.

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

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. In International Conference on Learning Representations, pp. 9965–10017, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/1b126cc38b8638e07bef37e7b2bb72bf-Paper-Conference.pdf.

A Pre-registration and Audit Trail

The analysis plan was frozen before inspecting the full sweep. It defines the base trajectory as the independent unit, exact next-action success as the primary endpoint, the three paired contrasts in Section 4, and 5,000 trajectory bootstrap repetitions. It also specifies model, task-family, context-size, and distractor-strength heterogeneity; five falsification tests; and the compound paper success criterion. The completed study passes the run health checks but fails the compound scientific criterion because the front-position and interaction effects do not replicate across the two tested artifacts and DAGPlace does not beat the strongest observed heuristic. This audit does not treat the untested symmetric edge contrast as a negative result.

The automated audit finds 24,144 unique case IDs: 18,096 controlled layouts and 6,048 baseline layouts. There are 432 unique procedural bases evaluated on two models, hence 864 model–base trajectory clusters. Every request is scorable and there are no backend errors. The audit records explicitly that the implemented edge is $p = .05$, the middle is $p = .40$, the near gap is zero, and the far gap is .20.

B Full Locality Heterogeneity

Table 4 reports the pre-specified far–near contrast by each stratification dimension. These intervals are descriptive and are not simultaneously adjusted.

Table 4: Pre-specified locality heterogeneity. Values are percentage-point differences with 95% trajectory-bootstrap intervals.

Dimension	Level	N	Far−near [95% CI]
Model	Gemma 3 27B	432	$+0.00$ $[+0.00, +0.00]$
	GPT-OSS 20B	432	-4.40 $[-8.10, -0.69]$
Blocks	32	288	-3.47 $[-7.64, +0.35]$
	64	288	-3.47 $[-7.29, +0.35]$
	128	288	$+0.35$ $[+0.00, +1.04]$
Distractor	Conflict	432	$+0.00$ $[-2.55, +2.55]$
	Lexical	432	-4.40 $[-7.18, -1.85]$

Across task families, the locality estimates are: authorization -5.56 pp ($[-11.11, -0.69]$), conditional branch -0.69 pp ($[-3.47, +2.08]$), error recovery -1.39 pp ($[-5.56, +2.78]$), multi-hop -3.47 pp ($[-9.03, +2.08]$), parameter propagation -0.69 pp ($[-6.25, +4.86]$), and state update -1.39 pp ($[-4.17, +1.39]$). Because there are six unadjusted family intervals, the authorization result is treated as a lead for replication, not a separate discovery.

C Controlled Response Surfaces

Figure 2 averages controlled cells across task family, block count, distractor type, and instances. It is a visualization of raw cell means, not the inferential model: paired contrasts in Table 1 are the pre-registered analysis. White cells are structurally infeasible and were not silently clipped.

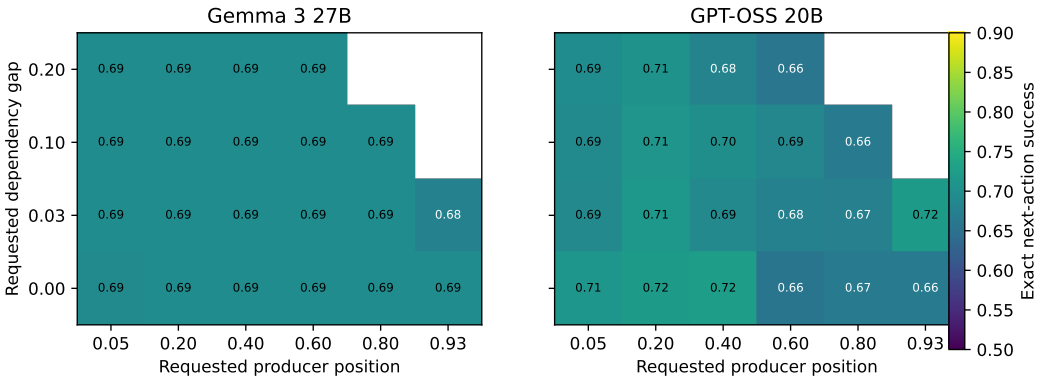


Figure 2: Mean exact next-action success by requested producer position and dependency gap. The color scale runs from 0.5 to 0.9 and all cells are numerically annotated.

D Task Construction Details

Each nonce is sampled from uppercase letters and digits using a seeded Python pseudo-random generator. Critical blocks receive a sequential dependency chain. All fillers are dependency-free. Chronological order places the critical chain after approximately one third of the fillers. Controlled layouts distribute critical blocks evenly over the requested span; all unoccupied slots are filled in a stable order. A validator rejects any layout with a duplicate or missing block, a reversed dependency, or a different evidence multiset.

For a representative parameter-propagation instance, the goal asks the model to issue an approved refund. The first critical transaction maps a random person nonce to a customer nonce; the second maps that customer nonce to an eligible account nonce. The expected next action uses the account nonce and a random amount. No nonce can be recovered from model pretraining. Equivalent programmatic templates implement branch decisions, three-hop lookup, corrected retry schemas, superseding state, and scoped authorization.

E Baseline Definitions

Chronological Original generated order.

Random legal A seeded random-priority topological sort.

Relevance first A topological sort prioritizing scalar block relevance.

Pin front/end The complete critical chain followed by, or following, every filler block.

Edge spread Dependency-connected components are relevance-ranked and alternated across the two ends while preserving internal order.

DAGPlace The best of the six initial layouts followed by 400 seeded legal swap proposals under the fixed objective in Section 3.

The optimizer uses $\lambda_p = \lambda_g = 1$. These values are not tuned on task success. A valid follow-up must tune them on development families and report held-out families, rather than selecting on the present test outcomes.

F Reproduction Instructions

From the repository root, the complete audit can be regenerated with:

```
python3 experiments/agent_placement/analyze_run.py \
  --run-dir experiments/agent_placement/runs/\
iclr-agent-placement-20260903-01
```

```
python3 paper/iclr2027/scripts/\
build_paper_artifacts.py
```

The second command reads only the deduplicated records.csv, asserts unique case IDs and complete paired baseline cells, and emits audited CSVs, LaTeX tables, vector figures, and audit_results.json. Unit tests for scenario generation, layout legality, rendering, resumption, and scoring are run with:

```
cd experiments/agent_placement
python3 -m unittest discover -s tests -v
```

G Confirmatory Expansion Plan

We scope the present claims to the two evaluated artifacts and treat Study 1 as a diagnostic pilot. A review-driven endpoint design will be independently pre-registered before any new confirmatory test call; Study 1 supplies hypotheses and a variance estimate but will not be pooled with that data. The complete auditable protocol is released as REVIEW_DRIVEN_CONFIRMATORY_V3.md.

1. Endpoint response surface. Measure producer and consumer token midpoints directly. Cross chain midpoints $\{.18, .50, .82\}$ with noncritical dependency gaps $\{0, .12, .24\}$, add producer- and consumer-anchored packed controls for the central far cell, and retain chronology. This yields fixed-gap translation and three explicit dilation paths without treating algebraically dependent coordinates as independent. Eight typed task families use task-specific semantic and stale-state distractors. The initial sample is 2,000 independent bases for each of four open architectures and 600 for each of three frozen frontier APIs; a direction-blind discordance rule can enlarge only the core contrasts.
2. Measurement and reliability gates. Require independent dev capability calibration, equal-length evidence removal, counterfactual nonce, irrelevant-order placebo, and consumer-displacement-matched translation. Repeat four key greedy-decoded cells in three fresh server/time blocks, randomize guided/native tool schemas versus unconstrained text on a fixed subset, and compare quantized serving with BF16 on a frozen subset. Backend failures count as failures in the intention-to-treat analysis.
3. Real counterfactual replay. On version-pinned τ^3-bench, AppWorld, and ToolSandbox, extract dataflow, state, control, authorization, and happens-before edges. Select one checkpoint per task by a structural rule fixed before model calls, audit graph precision and recall manually, and score next actions by environment-transition equivalence.
4. Closed-loop evaluation. Under 8k and 16k budgets, run two open and two frontier agents with five common seeds. All placement policies share a frozen evidence selector; end-state reward is primary, with policy violations, collateral changes, tokens, latency, and cost also reported.
5. CausalPlace-LCB. Learn a two-endpoint response surface only from synthetic development cells, optimize a lower-confidence bound subject to atomicity, topology, fixed-tail, and token constraints, and freeze it before seeing any real test outcome. Compare with chronology, recency, relevance, front/end pinning, dependency packing, random legal order, DAGPlace, and the strongest validation-selected heuristic.
6. Inference and falsification. Keep design-based paired contrasts primary and fit a prespecified two-dimensional logistic mixed model to all cells as secondary, with 10,000 joint base-cluster bootstrap draws and McNemar discordant-pair counts. Gate three diagnostic contrasts behind an omnibus test and reserve a separate $\alpha = .025$ family for the deployment comparison. Negative controls include irrelevant-only movement, no-edge tasks, evidence removal, counterfactual nonces, evidence checksums, illegal-layout rejection, template variation, and API drift anchors. Selection among

heuristics uses max-T correction; family/model follow-ups use Holm or are labeled exploratory.

The full claim requires a corrected diagnostic contrast in at least three of four open architectures, positive synthetic-to-real predictive validity, and a closed-loop improvement over the frozen strongest baseline without more than a two-point loss on any benchmark. Otherwise the result will be reported as a conditional measurement or a transfer failure, not as a universal optimizer. The exact design contract is released with the experiment artifact.