Relay main study (preregistration): do session notes help AI research agents solve multi-session tasks?

Ahmet Mat

Hypothesis · Computer Science · Artificial intelligence: AI agents
Signed 31 October 2026
Signed record · not peer reviewed
Papex PXH-QQQXPR0PNC4V · version 1

Keywords: AI agents; agent memory; session notes; preregistration; Claude Code; Codex; multi-session tasks; Kobon triangles; covering systems

Claim

In research tasks that take several sessions, with the work alternating between two AI apps (Claude Code with claude-sonnet-5-5, and Codex with gpt-6-luna), a short session note changes the share of runs solved within four sessions. The note has three parts: tried, failed, next. It is written after each session on the harness's request and read at the start of the next session. The comparison is with keeping only the workspace files. The test is two-sided, on two tasks with exact checkers (Kobon triangles for 8 lines; a covering system with distinct moduli all at least 13), with 90 runs per arm.

Why it matters

AI agent tools increasingly add memory or hand-off notes between sessions, including Papex's own session checkpoints. That such notes help is mostly assumed rather than measured. Our pilot (16 runs, 59 sessions, 31 Oct 2026) found no benefit (2/8 runs solved without notes, 1/8 with notes). It also found that asking the agent to write its note "before you stop" made it stop earlier: for Claude Code on the covering task, the median session length fell from 8.9 to 1.0 minutes. This study removes that cue, so both arms work for the same time, and uses a sample size that can detect a doubling of the solve rate. The result is published whatever it is.

What would falsify this?

Primary (H1): the Cochran-Mantel-Haenszel chi-square test stratified by task (1 df, no continuity correction, two-sided, alpha 0.05) on runs solved within four sessions, arm B against arm A. - If p >= 0.05, H1 is not supported. We then report the Mantel-Haenszel risk difference with its 95% interval and the effects that interval excludes. - If $p < 0.05$, the sign of the risk difference (B minus A) says whether notes helped or hurt. Working-time check (S1): if arm B's median session length is below 0.75 or above 1.33 of arm A's, we report the H1 comparison as confounded by working time, as in the pilot. No interim analysis: solve rates are not tabulated by arm until the last run finishes.

Methodology

Full protocol: notes/PROTOCOL-MAIN.md in the Relay repository at commit 94f95b0559c4f88585b9ba70ba92
The repository is released with the results.

Design - Arm A, files only: each run has one workspace folder that persists across its sessions. - Arm B, files plus the note: - The agent's first instruction is to read the previous note with

papex_get_draft_assist_context. That is the only Papex tool in the session. - After every session that another session follows, however it ended, the harness opens a wrap-up turn. The only tool there is papex_session_checkpoint. Limits: 3 minutes and 4 tool calls; Claude Code has no file tools and Codex runs read-only. The agent writes Tried, Failed and Next. - The Papex MCP server's own instructions, which ask for a checkpoint at the end of a session, are removed from its initialize response. - In both arms, leftover background processes are stopped and the checker runs before any note is requested. - Tasks: kobon8 (8 lines with 15 non-overlapping triangles, OEIS A006066) and covering13 (covering system with distinct moduli, all >= 13). The task files and exact checkers are unchanged from the pilot. - Runs: 45 per task and arm, 180 in all. - Sessions: up to 4 per run, stopping at the first pass. Claude Code in sessions 1 and 3, Codex in sessions 2 and 4. Each session has 15 minutes and 60 tool calls. - Isolation: no user settings, instructions, memory, plugins or network other than the model and, in arm B, Papex. The sandbox denies reads outside the workspace. - Order: the A and B runs of the same task and repeat form a pair, in random order within the pair; the pairs are shuffled (seed 20261031). Five parallel workers. - A session that did not run (usage limit, authentication or CLI error) is repeated and not counted.

Sample size: to detect 20% vs 40% with 80% power at alpha 0.05 two-sided, 82 runs per arm are needed with the normal approximation and 90 with Fisher's exact test. We use 90. Power is about 56% for 20% vs 35%.

Analysis (analysis/analyse_main.py) - Primary: the CMH test stratified by task, and the Mantel-Haenszel risk difference with a stratified Wald 95% interval. - Sensitivity: pooled Fisher exact test, and each task separately. - Secondary, without correction for multiple testing: - S1: main-phase minutes per session (per-run mean, Mann-Whitney U) and the share of sessions the agent ended itself. - S2: unchanged re-runs. A workspace program that an earlier session ran, run again with identical content, measured from transcripts and per-session workspace snapshots. Mann-Whitney U. - S3: session of the first pass. - S4: cost. - S5: notes read, notes written and note length. - Exploratory: judge-model labels of approaches and repeated dead ends; false success claims. - If the study stops early (budget, service), all finished runs are analysed and the shortfall is reported. Any change after registration is a reported deviation.

SHA-256 of the registered files at that commit: - notes/PROTOCOL-MAIN.md 1d035e44052d1a0976bab351 - analysis/analyse_main.py d461dc047b44d08d954aaf0aed49d5e2bf8b7b182207cbdaaf2c7acb578ff767 - config-main.json eb2c49c84a7e0fe8eff542bc53bf72040e8a7f326bf1fbb1002d6e19e8b6f270 - notes/main-schedule.json bddc49951dbc0814aedc15cabc14d6b8603c99537343863faa58c99771d35632 - run.py c30c7d5f7e720ba318bf7e7b279020cccf5eae7e0c0c0a290da11da2a9a737f5 - prompts/session.md 6f9f0cf8a14a868348ea7aa70fc08c8de92bd17c83501116a2ee8f8c8d00eecd - prompts/arm_B_main.md 21da6f7fecbaf861fe1b1ceb52822f65bf756de646180ff656ac764f08da329f - prompts/wrapup_B_main.md 12cb6ee2e083c21c53ba58b9f196ff0434406e4f827327e36d5278cac49caaa1 - tools/papex_mcp.mjs 24c2e453f29eb0e4b133a70e83f3a51882bf2bfadaaca8a02b63c7d0ebd2be2e - tasks/kobon8/TASK.md c0acf18f7a2e87b5e4eda814e206e55195713c481f2878a3866c51b9fedc0beb - tasks/kobon8/check.py 9c259110ed246fca20edcd533bba8f1178ea98c6b06e415d3094831f74daa239 - tasks/covering13/TASK.md 91b8773344fd513a5566cfe55024b63ac24f972bd40de2f21252a272fd3f7a93 - tasks/covering13/check.py 564b6c5ba07c7de62213ed32b8ca14da1e3884b55d904b942f50af4bd4442a42

Conflict of interest: Papex Labs builds the note tool being tested.

Expected outcomes

Before the pilot we expected notes to raise the solve rate. The pilot (2/8 without notes, 1/8 with notes) gives no support for a direction, so the primary test is two-sided and we state no expected direction. Secondary expectations: (S1) Working time per session is similar in both arms, because the note is now requested only after the session. (S2) Arm B re-runs fewer unchanged programs that an earlier session already ran. We expect a solve rate of about 20% without notes, as in the pilot.

Research setup

System or population Runs of two AI coding agents, Claude Code (claude-sonnet-5-5) and Codex (gpt-6-luna), alternating sessions on two mathematical construction tasks with exact checkers, under a fixed isolated harness.

Variables Independent: arm (A, workspace files only; B, files plus a Papex session note read at the start and written after each session). Primary dependent: run solved within four sessions (harness checker). Secondary: main-phase minutes per session, share of sessions ended by the agent, unchanged re-runs of earlier programs, session of the first pass, cost, note behaviour.

Parameters 45 runs per task and arm (90 per arm); up to 4 sessions per run; 15 minutes and 60 tool calls per session; arm-B wrap-up of at most 3 minutes and 4 tool calls; schedule seed 20261031; 5 parallel workers.

Observer or reference frame The harness's own copy of each task checker decides the primary outcome; no person or model judges it.

Assumptions and scope Two tasks, two apps and models at the versions recorded per run, one note format (tried, failed, next) delivered through Papex. Results may not transfer to other tasks, models, note formats or longer sessions.

Units Share of runs solved; risk difference in percentage points; minutes; counts per run; US dollars.

Tolerance or uncertainty Alpha 0.05 two-sided for the primary test; 80% power for 20% vs 40% solved.

Research mode Experimental

Known limitations Only two tasks. Model versions fixed at run time. Solve rates near 20% leave power for large effects only. The unchanged re-run measure is a proxy. Codex costs use third-party price lists. Papex Labs builds the note tool (conflict of interest).

Purpose New claim

References

[1] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, Shunyu Yao. Reflexion: Language Agents with Verbal Reinforcement Learning. 2023. arXiv:2303.11366
Cites

[2] A006066: Kobon triangles. https://oeis.org/A006066
Uses data

AI involvement

20 fields · 20 from Claude

Table 1: Record text

Who	Model	Fields
Claude, unchanged	claude-opus-5-5	Title, Claim, Why it matters, What would falsify this?, Methodology, Expected outcomes, System or population, Variables, Parameters, Observer or reference frame, Assumptions and scope, Units, Tolerance or uncertainty, Research mode, Known limitations, Language, Purpose, Field, Sub-field, Keywords

No AI work listed.

Names, models and roles are self-reported. Text origins were computed by Papex from what the AI apps prepared or proposed through Papex.

Generated by Papex from the signed record (research-record/v9). The signature covers the record's content, not this PDF file. Content hash 0x75ae…d77e · Signing key 0xc64e…4ca8. Check it at https://papex.org/verify/PXH-QQQXPR0PNC4V?version=1. Cite as https://papex.org/id/PXH-QQQXPR0PNC4V/v/1. AI-use statement: https://papex.org/id/PXH-QQQXPR0PNC4V/v/1/ai-use.