Scaling Discovery Does Not Require Forgetting: Verified Shared Archives Break the Multi-Agent Plateau on Anthropic's Performance Take-Home

Lindell S. Cumes

QWNTL Labs

Project page: https://qwntl.com/paper-societies-horizon

Abstract

Tan et al. report that longer, parallel, and communicating searches "all … eventually plateau as the agents accumulate history" and that "the key to scaling discovery further is to forget": discard every agent's session and the candidate history, and restart from the best candidate. They never test resetting sessions while keeping candidates; we introduce a shared verified archive that does. It bounds conversations and keeps every evaluator-verified attempt with its source and cited tools, so different agents' improvements compose. On Anthropic's performance take-home, where their forgetting team reaches 1,077 cycles and CORAL 1,103 with other models, four Claude Opus 5.5 agents on the archive reach a median of 914 [913, 916] cycles over 3 seeds in 8 hours, against 948 for a CORAL-faithful team and 946 for the best of four isolated agents on matched machines. Both baselines' medians stall at 946–956 cycles after 3 hours, where several CORAL-faithful agents concluded about 951 was near the floor and 7 of 12 fell into a no-op resume loop, which we name the false floor; from 2.75 hours the archive's worst seed stays ahead of both baselines' best seeds. On open problems, a team built 60 of its first 98 new bests on another agent's attempt or tool and, in one run per arm, led every arm at 8 hours on Erdős's minimum overlap problem, the first autocorrelation inequality, and AHC026.

Release. https://qwntl.com/paper-societies-horizon. Method: the per-seed anytime curves of every take-home and open-problem arm, the take-home task prompt, and the C_1 and Erdős constructions with their exact verifiers; Appendix B prints the take-home archive's briefing and compiler prompts.

1 Introduction

Long agent searches stop improving. Tan et al. measured the stall in all three ways of spending more compute on discovery: letting one agent run longer, running agents in parallel, and letting parallel agents build on each other's candidates. They traced it to the history the agents accumulate, "Each agent becomes anchored to its own past attempts, making fewer and smaller edits," and concluded that "the key to scaling discovery further is to forget" [32]. At each of three forgets, every agent "discards its session, along with all candidate solutions and records in the filesystem state, and keeps only the best candidate." On four tasks this raises the final best score by up to 15% over parallel agents and up to 39% over a single agent.

Forgetting removes two different things at once. One is an agent's conversation, which records what the agent concluded about the task; nothing checks those conclusions, and an agent that concludes it has reached the floor stops searching. The other is the candidate history: attempts that an evaluator has scored, with their source and the tools they cite. Tan et al. attribute the plateau to the first, an agent that "becomes more conservative as its session grows longer," yet their

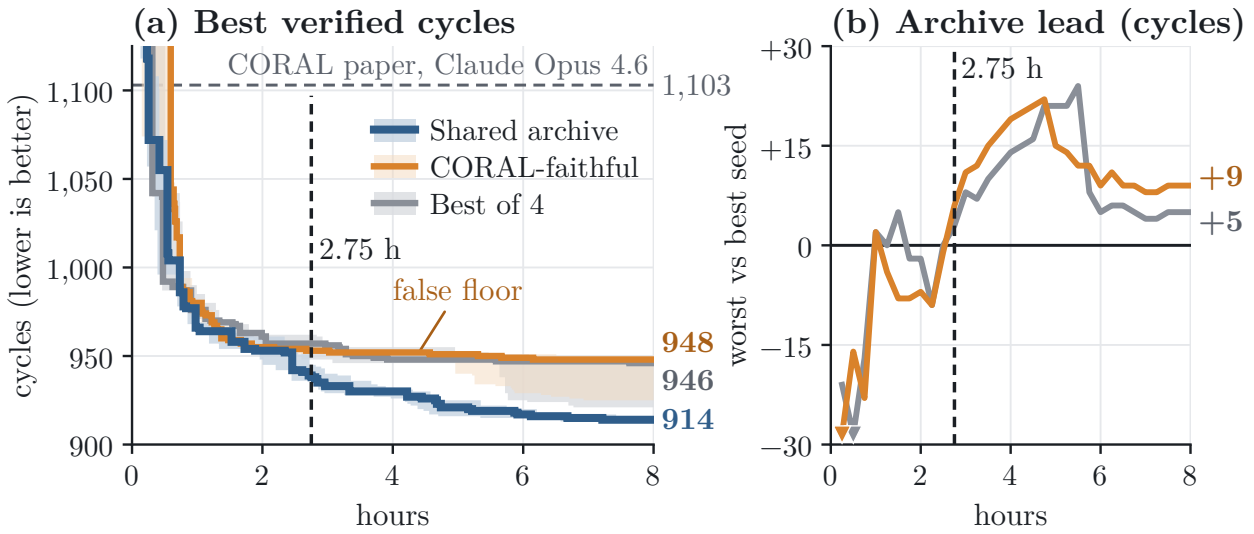

Figure 1: **The shared verified archive keeps descending after parallel and communicating search plateau.** Anthropic's performance take-home, four Claude Opus 5.5 agents per arm at medium effort, three seeds per arm launched together; all arms matched on machine, hours, effort, starting kernel, and protected evaluator. (a) Best verified cycles so far (lower is better); lines are seed medians and bands span the three seeds. Every arm starts from the 147,734-cycle kernel, so the curves enter from above the axis. The dashed line is CORAL's published 1,103 cycles with Claude Opus 4.6, a different model. "False floor" marks the score that several CORAL-faithful agents concluded was near the floor. (b) The archive's lead at the least favorable pairing, the best seed of the other arm minus the worst archive seed, every 15 minutes; it is positive at every checkpoint from 2.75 hours. Triangles mark two early values below the axis (−88 and −85).

only comparison keeps both or drops both. Fork-and-Flush flushes the chat and keeps one lineage's workspace [6], and Stateless Language Agents hold candidates in the harness and give agents no lasting conversation [38]; neither lets one agent run another agent's tools. CORAL shares graded attempts that any agent can check out, but resumes each agent's own long session and shares notes and skills that nothing checks [28]. None of these designs bounds the conversation while keeping every agent's verified attempts and tools where every other agent can run them.

This paper builds that design, a shared archive of verified, executable attempts, and races it against parallel and communicating search on a benchmark both Tan et al. and CORAL report. On Anthropic's performance take-home, Tan et al.'s forgetting team reaches 1,077 cycles with DeepSeek V4.1 Flash, and CORAL 1,103 with Claude Opus 4.6 [1, 28, 32]. Four Claude Opus 5.5 agents on the archive reached a median of 914 cycles at 8 hours over three seeds. A CORAL-faithful team reached 948 and the best of four isolated agents 946, on identical machines with the same hours, effort, starting kernel, and evaluator. All five runs of the archive on that machine type finished at 910 to 916 cycles, and all nine CORAL-faithful and best-of-four runs at 921 or above (Figure 1a).

The baselines reproduce the plateau Tan et al. describe, and the archive does not. From the third hour the CORAL-faithful and best-of-four medians stayed between 946 and 956 cycles; between 4 and 8 hours they gained 4 and 2 cycles, while the archive's median gained 16. CORAL-faithful agents could also fetch every attempt of all four agents; the archive differs in bounding each conversation and promoting only verified insights and tools, and it is the arm that kept improving. Its worst seed was ahead of the best seed of both baselines at every checkpoint from 2.75 hours (Figure 1b).

The CORAL-faithful plateau sits where its agents believed the floor was. Several of its agents concluded that about 951 cycles was near the floor and from then on answered each resume prompt in seconds without acting; 7 of its 12 agents fell into that loop. We call a believed optimum that a matched search refutes a *false floor*. Zhang et al. saw the same stall in upstream CORAL, where agents "repeatedly declare their solutions optimal" [38]. The archive's agents open each session on verified attempts and tools, and none of its 16 agents in four runs entered the loop. The same model on the same machine verified 914 cycles.

The late descent is built from other agents' work. In an open-problem team of three Claude Opus 5.5 agents per problem

on a later version of the archive, 60 of the first 98 new bests in its first 6.7 hours ran another agent's tool or descended from another agent's attempt. In one run per arm, the team was ahead of a CORAL-faithful team, isolated agents, and a single agent at 8 hours on Erdős's minimum overlap problem, the first autocorrelation inequality (C_1), and AtCoder Heuristic Contest 026 (AHC026). Its minimum-overlap construction ends by crossing two separately evolved solutions, neither of them the team best, with a teammate's genetic-algorithm tool; a restart from the team best keeps neither parent.

The contribution is the shared verified archive, the design Tan et al.'s comparison leaves out, and the first matched anytime race of such an archive against parallel and communicating search on one model. Within its 8 hours the race shows that the CORAL-faithful plateau is a false floor, and that a search that keeps its verified history breaks through the plateau that forgetting was proposed to escape. Section 2 states the problem and the information each design keeps; Section 3 gives the archive; Sections 5 and 6 measure it.

2 Problem Statement and Information Flow

Problem. A task is $\tau = (d, x_0, V)$: a description d, a starting candidate x_0, and a protected verifier V that returns a score $V(x)$ for a candidate x or rejects it. An agent is a pair $A = \langle M, H \rangle$ of a model M and a harness H; a team runs k agents under one harness. The budget is T wall-clock hours on a fixed machine, with no per-arm cap on model calls. The objective is discovery: the best verified score by time T, read as a function of $t \le T$ so that one run yields an anytime curve.

Definition 2.1 (Best-so-far curve and advantage). Let t_a be the time at which verified attempt a was recorded. The best-so-far curve of harness H is $B_H(t) = \min\{V(a) : t_a \le t\}$ for minimization (maximum for AHC026). The advantage of H over H' is $\Delta_{H,H'}(t) = B_{H'}(t) - B_H(t)$ for minimization, signed so that $\Delta > 0$ when H is ahead.

Definition 2.2 (Crossover time). The crossover time $t_\times(H, H')$ is the earliest covered t with $\Delta_{H,H'}(u) > 0$ at every covered $u \ge t$. With several seeds per arm, Δ is read at the least favorable pairing, the worst seed of H against the best seed of H'.

Definition 2.3 (Late gain). The late gain of H over $[t_1, t_2]$ is $G_H(t_1, t_2) = B_H(t_1) - B_H(t_2)$ for minimization (reversed for AHC026), the improvement the harness still makes after t_1. A harness has plateaued over $[t_1, t_2]$ when its late gain is small against another harness's late gain on the same interval.

Definition 2.4 (False floor). An agent holds a false floor when it concludes that the best score it has seen is at or near the attainable optimum and stops submitting attempts, while a matched search with the same model, task, machine, and hours verifies a better score.

Definition 2.5 (Cross-agent reuse and build-on delay). A new team best recorded by agent i reuses another agent's work if its producing data flow contains an attempt by agent $j \ne i$, as parent or ancestor, or code from a tool written by j, run or adapted. The reuse rate is the share of new bests that do so. The build-on delay is the time from a teammate's new best to the first attempt that starts from it.

Definition 2.6 (Coordination share). The coordination share is the share of an agent's tool calls spent reading and writing the archive: reading attempts, insights, and shared files; fetching another agent's source; writing insights; and proposing tools. Running a teammate's code counts toward the task.

Information-flow contract. Every arm here freezes the model weights, the task, and the evaluator. Each attempt is fixed when it is submitted: the take-home evaluator scores the submitted source on fresh random inputs, and the reported score of an arm at time t is $B_H(t)$ over attempts recorded by t. Arms differ in what persists between an agent's sessions and what one agent can see of another's work (Table 1). The evaluator reports correctness, the score, and per-engine utilization in every take-home arm; CORAL-faithful appends heartbeat prompts and the archive a pointer to the team best.

Remark 2.1 (The best of k against a composition). The isolated arm's curve is $B_{\text{iso}}(t) = \min_i B_i(t)$ for minimization (maximum for AHC026) over its k agents, the best of k searches, none of which can start from another agent's attempt. On the archive any verified attempt can be a parent, so improvements found by different agents can enter one lineage. The two curves coincide while every agent finds the same improvements, because combining identical improvements gains nothing over one of them. They separate once the remaining improvements are small and distinct. The best of k

Table 1: **What each design keeps.** Rows above the rule are the arms raced here; rows below are the designs named in the introduction, as their papers describe them. "Verified" means a protected evaluator scored the item before any other agent could see it.

Design	Candidate history across sessions	What other agents see	Agent conversation
Shared verified archive (ours)	every verified attempt, kept for the run, with source and cited tools	any attempt as a parent; tools; insights compiled from verified gains, candidates marked unverified	fresh briefing after 3 resumes or 4 h
CORAL-faithful [28]	every scored attempt, plus free-form notes and skills	all attempts; notes and skills as agents wrote them, unverified	resumed with the latest score and "Continue optimizing"; fresh after 4 h
Best of four isolated	own attempts and own workspace	nothing	resumed with a fixed nudge; fresh after 4 h
Forgetting [32]	discarded at each of three forgets; restart from the best candidate	teammates' scored candidates, within a round	grows within a round; discarded at each forget
Fork-and-Flush [6]	one lineage's workspace kept; losing forks pruned	nothing; forks run independently	flushed at each fork
Stateless Language Agents [38]	candidates and measured outcomes, held by the harness	what an Advisor assigns	none outlives an invocation

then grows with its best member, and a composition grows with all of them, later by the build-on delay. A restart from the best candidate keeps one lineage and removes the others it could compose with.

3 A Shared Archive of Verified, Executable Attempts

The archive is an append-only event log, chained by HMAC, of verified attempts with content-addressed sources (Figure 2). Agents reach it only through a command-line client. Four rules define it.

Admission. An attempt enters only after the protected evaluator scores it. On the take-home, the candidate builds in a sandbox without network access, with a cleared environment and a hidden home directory, and emits its instruction list as data; a separate process that never imports candidate code runs a hash-pinned simulator on fresh random problems. Each record holds the score, the source, the parent the agent names, and the insights and tools it cites.

Execution by others. Reuse requires that another agent can run an attempt. An agent copies any attempt's source into its workspace with the archive's source command and the attempt's identifier, and runs any promoted tool from the shared directory. A tool enters as a candidate when an agent proposes it and is promoted once an attempt that cites it verifies an improvement.

Knowledge tied to verified gains. The evaluator verifies attempts; insights and tools are promoted only on the evidence of a verified attempt. Every verified improvement, parent to child, is compiled by Claude Opus 5.5 into a structured insight backed by the two attempt identifiers and the cycle gain. An insight that an agent proposes enters as a candidate; briefings show at most the four latest candidates, each marked UNVERIFIED, and a candidate is promoted once an attempt that cites it improves. Failed attempts are distilled by Claude Haiku 4.5 into one-line constraints, and raw failure transcripts never enter a briefing. Every new team best is re-scored at $k = 64$ in the background; a refuted attempt drops out, and every insight resting on it is invalidated.

Briefing. A session opens on a briefing compiled from the archive: the team best with its diagnostics and the path to its source, an assigned parent, the verified insights ranked by gain, the constraints, and the tools. The assigned parent is the team best unless a teammate already holds it; otherwise it is the attempt, from any agent, whose descendants reached the best score with the most verified improvements (its clade credit). When the team best has not improved for 15 attempts, parents are drawn away from the best's strategy, and the briefing asks for a structurally different strategy

Input: archive $\mathcal{A}$ (verified attempts, tools, insights, constraints); agent i with model M; evaluator V.

1. $p \leftarrow b^\star$, the team best, unless a teammate holds it or $b^\star$ is unchanged for 15 attempts; otherwise the attempt in $\mathcal{A}$, from any agent, with the highest clade credit.
2. Brief agent i: $b^\star$ and its source; p; verified insights by gain and the latest candidates marked UNVERIFIED; constraints; tools; a plateau notice if $b^\star$ is unchanged for 15 attempts.
3. **repeat** until the session ends (exit, or a fourth resume): the agent edits a candidate x in its own sandbox, optionally running any tool in $\mathcal{A}$ or the source of any attempt, and submits $(x, \text{parent}, \text{cited items})$.
4. $\quad s \leftarrow V(x)$ in a separate process on fresh inputs; **if** rejected, distill a constraint and continue.
5. $\quad$ Append $(x, s, \text{parent}, \text{cited items})$ to $\mathcal{A}$, chained to the previous event.
6. $\quad$ **if** s improves its parent: compile an insight; promote every cited tool and insight. **if** $s < b^\star$: re-score at $k = 64$; on refutation, invalidate the attempt and every insight resting on it.
7. The next session of agent i starts at step 1 with a fresh conversation.

Algorithm 1: One session of an agent on the shared verified archive. Nothing is ever removed from $\mathcal{A}$ except attempts a re-score refutes.

and lists those already explored. A session may be resumed at most three times, each time with the archive's changes since it began; the next session starts fresh. The agent's conversation is therefore bounded, and the candidate history is never discarded.

Open-problem team. On Erdős's minimum overlap problem, C_1, and AHC026, a later version of the same harness, which we call the society, ran three agents per problem. Each session lasts at most 4 hours and opens on a briefing of at most 9,000 harness-assembled characters: the team best, a parent sampled by clade credit, a direction rotated each session, and bounded lists of insights, recent results, constraints, and tools (Appendix A.1).

Baselines. CORAL-faithful ports upstream CORAL at commit `cce3101`: its prompts, its heartbeats (reflect, consolidate every 10 attempts, pivot after 5), shared attempts projected from the evaluator, and free-form notes and skills that agents edit directly [28]. An early exit resumes the same conversation with the latest evaluation summary and "Continue optimizing," as upstream does. The nine upstream files the port pins match commit `cce3101` by hash, and its prompts are upstream's apart from the command-line interface, disabled web access, and fixed heartbeat defaults (Appendix A.1); it keeps CORAL's own task hints, which no other arm receives. Every CORAL-faithful seed passed CORAL's published 1,103 cycles within 37 minutes, and its 948 is below every CORAL take-home value in print, among them 1,103 from Qu et al. and 1,130 and $1{,}350.0 \pm 134.2$ from Zhang et al.'s runs of CORAL v0.7.13 [38]. The best of four isolated agents runs four agents that see only their own attempts and workspace, and reports the best of them. In every take-home arm a session lasts at most 4 hours and each agent's working directory persists; the archive alone limits a conversation to three resumes before a fresh briefing. The single agent of Section 6 keeps a conversation meant to last the run and a private notebook, with the CPU share that three agents divide in the other arms.

4 Evaluation Protocol

Validity controls. Anthropic's repository reports that none of the solutions it received on the first day below 1,300 cycles was valid, because a language model had modified the tests [1]. The take-home evaluator here passed 9 adversarial qualification cases (simulator tampering, secrets, network access, wrong output, hangs, malformed programs) before any run, and it never imports candidate code. The archive's three final kernels in the main comparison were each re-run three ways: against the official test suite in a fresh clone of the repository, on 64 fresh random problems, and on 607 stress problems with edge-case and full-32-bit inputs. Each was correct at its reported cycle count. Two margins were registered before launch, at least 10 cycles below CORAL-faithful and at least 20 below the best of four. Every archive's hash chain was verified before its machine was deleted. The two bounds of Section 6.4 were recomputed in exact arithmetic by checkers written independently from the problem definitions.

Statistics. Take-home values are seed medians with the minimum and maximum over three seeds. Advantages are differences of seed medians, bracketed by the least and most favorable of the nine seed pairings. Exact one-sided

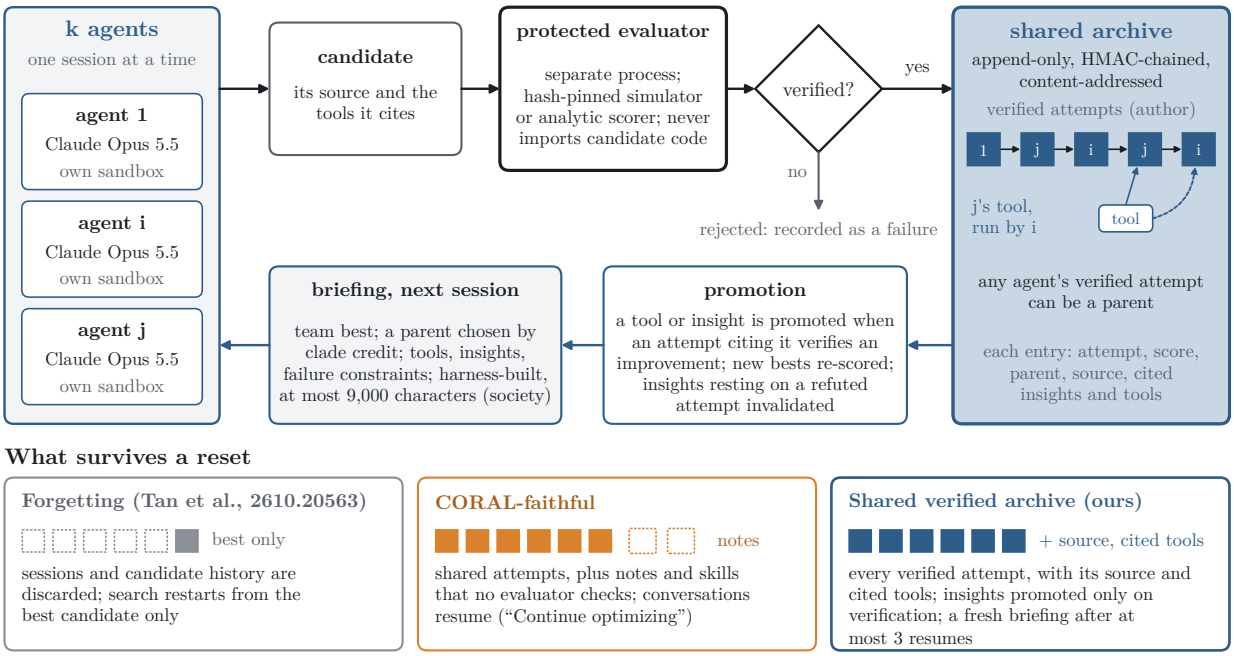

Figure 2: **Only verified attempts enter the archive, and any agent can run any of them.** The per-step loop of Algorithm 1: agents submit candidates with their parent and cited tools; a protected evaluator in a separate process scores each one; verified attempts are appended with source and cited tools; promotion ties insights and tools to verified gains, and re-scoring removes refuted attempts; each new session is briefed from the archive. The strip contrasts what survives a reset in each design of Table 1.

Mann–Whitney permutation tests compare 8-hour values, three seeds against three and, pooled across runs on the same machine type, five against four and five against five. Open-problem comparisons are single runs read on a 0.25-hour grid of matched working hours.

5 Anthropic’s Performance Take-Home: The Archive Breaks the Plateau That Parallel and Communicating Search Reach

The take-home asks for a kernel for a simulated VLIW SIMD machine, scored in clock cycles; fewer is better [1]. The starting kernel takes 147,734 cycles. The main comparison launched three seeds of each arm together on October 7, 2026.

5.1 The Archive Finishes 32 to 34 Cycles below Both Baselines

At 8 hours the archive’s median was 914 cycles, against 948 for CORAL-faithful and 946 for the best of four (Table 3). Its advantage was 34 cycles over CORAL-faithful, ranging from 9 to 37 over the nine seed pairings, and 32 over the best of four, ranging from 5 to 37. Both registered margins held, and the permutation test gave $p = 0.05$ for each comparison, the minimum attainable for three seeds against three; the test reached that minimum at every hourly checkpoint from the third hour.

The ordering holds when the earlier four-agent Claude Opus 5.5 runs on the same machine type are pooled in: the original run, the model-change run’s first 8 hours, and one more best-of-four seed. The five archive runs ended at 910 to 916 cycles and the nine CORAL-faithful and best-of-four runs at 921 or above ($p = 0.008$ against CORAL-faithful, $p = 0.004$ against the best of four). A single Claude Opus 5.5 agent on that machine type ended at 953.

The archive’s 914 is below every model result published in cycles on the take-home as of October 30, 2026, the lowest of which is AgentDiscover’s 960 with Claude Opus 5 (Table 3). The matched comparison is the one that tests the

Table 2: **Evaluation protocol.** Every arm of a comparison ran on its own identical machine (the two open-weight arms shared one) for the same hours, with the same model, effort, starting candidate, and evaluator; no arm had a per-arm cap on model calls. Matched axis: wall-clock hours (take-home), matched working hours (open problems).

	Take-home, main comparison	Take-home, other configurations	Erdős, C_1, AHC026
Model	Claude Opus 5.5 (`claude-opus-5-5[1m]`)	Claude Opus 5.5; mixes with GPT-6 Astra and Claude Haiku 4.5; Qwen3.5-27B	Claude Opus 5.5
Harness	Claude Code in a per-agent sandbox	as main; Codex for Qwen3.5-27B	Claude Code 2.1.289
Effort	medium	medium	xhigh, web tools off
Agents per arm	4	4–6	3 per problem (single agent: 1)
Seeds	3 per arm, launched together	1 per configuration	1 per arm
Machine	n2d-highcpu-48 per arm	n2d-highcpu-48 or c2d-highcpu-56	n2d-highcpu-96 per arm
Wall clock	8 h	8 h (6 h open weights)	read at 8 matched working hours
Evaluator	protected; $k = 8$ fresh problems per attempt	as main	analytic scorer (Erdős, C_1); 150 development cases (AHC026)
CORAL port	upstream `cce3101`, prompts and heartbeats retained	as main	as main

design, and on Claude Opus 5.5 it does not reproduce CORAL's own result. CORAL reports that four co-evolving agents finish 77 cycles below the best of four independent runs, 1,103 against 1,180 with Claude Opus 4.6 [28]. On Claude Opus 5.5, in a matched race of three seeds, CORAL-faithful finished at 948 and the best of four at 946; the communication CORAL adds bought nothing, and the archive bought 32 to 34 cycles.

5.2 Parallel and Communicating Search Plateau; the Archive Keeps Descending

The two baselines play the roles of Tan et al.'s parallel and communicating modes on this task, and both plateaued. From 3 to 8 hours the CORAL-faithful median moved from 953 to 948 cycles and the best-of-four median from 956 to 946; between 4 and 8 hours their late gains (Definition 2.3) were 4 and 2 cycles. The archive's median moved from 933 to 914 over the same five hours, a late gain of 16 cycles between 4 and 8 hours, four times CORAL-faithful's and eight times the best of four's.

The crossover time is 2.75 hours against both baselines (Definition 2.2). From 165 minutes, the worst archive seed was ahead of the best seed of CORAL-faithful and of the best of four at every 15-minute checkpoint to 8 hours (Figure 1b); on exact attempt times it stayed ahead of both from 163.4 minutes. The advantage then grew while the baselines stalled: over the better baseline it was 20 cycles at 3 hours, 30 at 6, and 32 at 8.

The opening improvements were large and the same for every agent, as Remark 2.1 predicts. By 30 minutes every seed of every arm was more than 129 times faster than the starting kernel, and in the model-change run the first two attempts, from two agents in the same minute, were the same vectorized rewrite. Four agents that find one improvement gain nothing by sharing it. The late improvements were small and distinct, and the archive gives every agent the team best, every other agent's kernels, and their tools as starting points.

5.3 The False Floor: CORAL-Faithful Agents Stop at a Score the Archive Refutes

Under the CORAL-faithful protocol an agent that exits early is resumed in the same conversation with the latest evaluation summary and "Continue optimizing." Several agents concluded that about 951 cycles was near the floor. From then on they answered each resume in about 6 seconds without acting, hundreds of times, on conversations of 0.8 to 1.6 million cached tokens per resume. Seven of the 12 CORAL-faithful agents of the main comparison fell into this loop, defined as at least 50 sessions shorter than 60 seconds, and four of the seven made no attempt after 186 minutes.

Table 3: **Four Claude Opus 5.5 agents on the shared verified archive finish 32 cycles below the better baseline.** Best verified cycles on Anthropic's performance take-home, seed median [min, max] over three seeds launched together; lower is better. All arms: four Claude Opus 5.5 agents, medium effort, n2d-highcpu-48, same starting kernel and evaluator. Δ is the better baseline's median minus the archive's. The last five rows are borrowed from work that ran the same task (one row on its 2-hour version) with other models and budgets.

Arm (model, harness, effort)	1 h ↓	3 h ↓	4 h ↓	6 h ↓	8 h ↓
Claude Opus 5.5, shared verified archive, medium	**966**	**933**	**930**	**917**	**914** [913, 916]
Claude Opus 5.5, CORAL-faithful, medium	980	953	952	949	948 [925, 950]
Claude Opus 5.5, best of four isolated, medium	976	956	948	947	946 [921, 950]
Δ vs better baseline (cycles)	10	20	18	30	32
Published on the same task, other models and budgets[†] *(final value)*					
Claude Opus 5, low effort, AgentDiscover, $100 [12]					960
Claude Opus 4.8, Stateless Language Agents, 10^9 tokens [38]					1,039
DeepSeek V4.1 Flash, team with forgetting, $40 per run [32]					1,077
Claude Opus 4.6, CORAL, four agents [28]					1,103
Claude Opus 4.5, Anthropic's improved test-time harness, 2-hour version[‡] [1]					1,363

[†]Borrowed values as printed: Farahbakhsh et al. Table 4; Zhang et al. Table 1, one Claude Code run (their GPT-5.5 runs reach $1{,}112.0 \pm 8.9$); Tan et al. Table 1, best of three runs (mean $1{,}097 \pm 18$); Qu et al. Table 3 (best of four independent runs 1,180); the take-home repository's README.
[‡]The 2-hour version of the task starts at 18,532 cycles instead of 147,734, on the same simulator.

Table 4: **Only CORAL-faithful agents fall into the no-op loop.** Failures in the main comparison, summed over three seeds per arm (12 agents per arm). Attempt failures are rare everywhere; the failures that matter are sessions in which an agent does nothing.

	Shared archive	CORAL-faithful	Best of four
Attempts (evaluator calls)	211	168	201
Incorrect attempts or build errors	1	1	2
Sessions under 60 s with no attempt	1	4,326	43
Agents in a no-op resume loop ($\geq$50 such sessions)	0	7	0
Agents whose last attempt came before 240 min	0	5	1
New team bests after the 2.75 h crossover	51	22	21

In one seed, three of the four agents made their last attempt between 94 and 135 minutes (Table 4). The other three looping agents, all in a second seed, kept submitting between resumes until 383 to 416 minutes, and that seed reached 925.

CORAL-faithful agents read the common task prompt, which says there is no target to stop at, followed by upstream CORAL's task text, which gives "Best known: ~1,363 cycles". Every CORAL-faithful seed was below 1,103 within 37 minutes. The floor was false. The same model on the same machine type verified 914 cycles on the archive and 910 in the original take-home run, and a seeded archive team that added two GPT-6 Astra agents reached 877 on a c2d machine from a seeded compiler. No archive agent and no best-of-four agent entered the loop, 0 of 16 in each. The CORAL-faithful conversation accumulated the agent's own conclusion that the search was done, and each resume returned the agent to it; upstream CORAL injects heartbeats "without discarding accumulated session state" and "resumes agents with their prior context" [28]. The archive's briefing carries the team best, its source, insights compiled from verified gains, and tools, and carries agents' own proposals only as candidates marked UNVERIFIED.

The false floor is not an artifact of the port. Zhang et al. ran upstream CORAL v0.7.13 on a GPU kernel task and report that its agents "repeatedly declare their solutions optimal, and 98% of the final 1,500 sessions make no tool calls while token consumption continues" [38]. Tan et al. record the same behavior in their own single agent on the take-home, where 532 of 763 completed turns produced no candidate [32]. Forgetting clears that conclusion by deleting everything but the best candidate. The archive clears it by never putting the conversation in charge of the search state.

Table 5: **The archive makes about as many evaluator calls as the best of four and spends less than CORAL-faithful.** Per run in the main comparison, seed median [min, max]. USD at list prices of $5, $25, $0.50, and $6.25 per million input, output, cache-read, and cache-write tokens of Claude Opus 5.5. The archive's service calls are its insight compiler (Claude Opus 5.5) and failure distiller (Claude Haiku 4.5); re-scorings run the evaluator at $k = 64$.

Per run	Shared archive	CORAL-faithful	Best of four
Evaluator calls (attempts)	71 [64, 76]	60 [40, 68]	68 [61, 72]
Agent model turns	4,345 [3,704, 4,451]	5,470 [4,573, 5,939]	2,863 [2,678, 3,462]
Agent spend, USD	842 [772, 848]	1,237 [1,027, 1,302]	553 [506, 636]
Archive service calls	69 [62, 75]	–	–
Background re-scorings	38 [29, 38]	–	–
Final cycles ↓	**914** [913, 916]	948 [925, 950]	946 [921, 950]

5.4 Three Alternative Explanations, Each Answered by the Logged Runs

Does the archive win only because the CORAL port stalls? The best of four answers it. Its agents had no resume loop, 11 of its 12 agents were still submitting after the fourth hour, and its median still plateaued at 946 cycles, 32 above the archive; against the pooled best-of-four runs the archive's margin was 33 cycles.

Does the archive win by doing more? The evaluator call is the unit every arm spends, and the archive made about as many as the best of four, 211 against 201 over three seeds, plus 105 background re-scorings (Table 5). At the number of calls each baseline run made in its whole 8 hours, every archive seed was already below that run's final score, in all nine pairings with the best of four and all nine with CORAL-faithful. After 40 calls every archive seed was ahead of every baseline seed, 939 [930, 944] cycles against 954 [950, 957] for CORAL-faithful and 955 [952, 957] for the best of four (Figure 3). At the harness's list-price table the archive's agents spent $842 per run, 32% less than CORAL-faithful's $1,237 and more than the best of four's $553. It reached 920 cycles after $431 to $584 (spend pro-rated within sessions); no baseline run reached 920 at any spend (Table 13).

Does the archive win through its prompts? The archive's one prompt aimed at stalls is the plateau notice, and it never fired: none of the 29 fresh briefings in the main comparison carried it. After the 2.75-hour crossover, 37 of the archive's 51 new team bests drew on another agent's attempt, 35 of them through their ancestry, and those 37 steps account for 54 of the 77 cycles the archive gained after 165 minutes. Every archive seed's final kernel descends from attempts by three or four agents; every best-of-four final descends from one.

Do the late gains use history that forgetting deletes? Fourteen of the archive's 51 new team bests after the crossover, worth 27 of its 77 late cycles, started from an attempt that was not the team best when they were submitted; a restart from the best candidate keeps none of those starting points. The archive's agents reset their conversations 17 times across the three seeds. Thirteen late new bests, worth 21 cycles, cited a tool proposed before the submitting agent's current conversation began, and for 7 of them, worth 11 cycles, the tool was another agent's, so it could reach the agent only through the archive.

5.5 Late Improvements Compose across Agents

The seed that reached 913 cycles shows the composition attempt by attempt (Table 6). Until 195 minutes agent 3 descended alone with its own annealer, from 938 to 928 cycles. Agent 4 then took agent 3's kernel and annealer, agent 1 and agent 3 each improved agent 4's result in a different way, and agent 4 merged the two siblings. Its message, "3-way merge of A59 (rm=5 relocation knob, e3) with A58 (ALU const synthesis from header vload, e1) on base A57; no retune. 923->922", records the step. A reset that keeps only the best candidate keeps at most one of A58 and A59, and the merge needs both.

Agent 4 combined further work from agent 3, a select-tree setting from A65, to reach 917. Agent 1 then ran agent 3's code with agent 4's configuration for 914, and agent 2 re-annealed that configuration on agent 3's code for the final 913. The final kernel's chain of first parents is 31 attempts long and changes hands between agents six times. The model-change run shows the same late phase: its team best fell from 946 to 911 in 13 steps over 203 minutes, every step's lineage holds kernels by at least two agents, and no step exceeded 6 cycles.

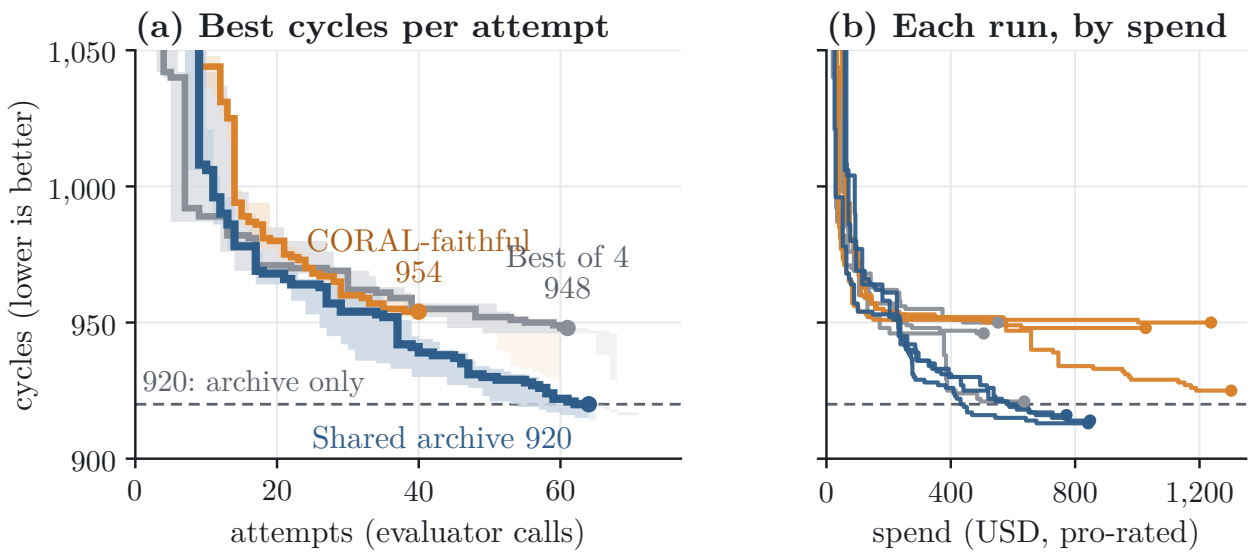

Figure 3: **Per evaluator call, every archive seed is ahead of every baseline seed from the 35th attempt, and only archive seeds reach 920 cycles.** (a) Best verified take-home cycles against the number of attempts each run had submitted (evaluator calls), four Claude Opus 5.5 agents per arm, three seeds per arm; lines are seed medians while all three seeds have that many attempts, and bands span the seeds (lighter where two remain). (b) Each run's best against its pro-rated agent spend in USD, carried flat to the run's total. All arms matched on machine, hours, effort, starting kernel, and evaluator.

5.6 The History Survives Hardware, Model Families, and a Switch to GPT-6 Astra

The archive stayed ahead on a second machine type, with mixed model families, from a seeded start, and with Qwen3.5-27B in every agent role, the insight compiler, and the failure distiller, where no other model ran and the archive reached 18% fewer cycles than four isolated agents (Table 7).

The model-change run tests the history directly. At 8 hours, with the team best at 911 cycles, all four agents switched from Claude Opus 5.5 to GPT-6 Astra and kept the archive. Each successor opened on a briefing that held the team best with its per-engine utilization, an assigned starting attempt, and the verified insights ranked by gain. Fourteen minutes after the switch, agent 2 took the 911-cycle Claude Opus 5.5 kernel, reassociated an XOR at the tree's wrap so that two operations run in parallel, and re-annealed the result with a simulated-annealing tool that Claude Opus 5.5 agent 3 had written, reaching 910. The successors made 7 attempts in 2.85 hours, all correct, and set five new bests down to 906.

6 Open Problems: Reused Work Produces the Bounds

Problems and metrics. For C_1, the task asks for a nonnegative step function f on $[-\frac{1}{4}, \frac{1}{4}]$ that minimizes $\max_t (f * f)(t)/(\int f)^2$ [24]. Erdős's minimum overlap problem asks for a step function $h : [0, 2] \to [0, 1]$ with $\int h = 1$ minimizing $\max_t \int h(x)\,(1 - h(x + t))\,dx$ [13]. Both constructions give upper bounds on their constants. AHC026, "Stack of Boxes," is scored on 150 generated development cases [2, 14]. C_1 and Erdős scores are margins below the Einstein Arena record at launch [4, 11], whose minimum improvements are 10^{-8} and 10^{-7}. Working hours subtract recorded downtime from the time since each problem's first session. Every arm received the same public material, including the record constructions. Each arm also ran a fourth problem, the kissing configuration in dimension 11, where every arm stayed at the published 604; it is left out of the score tables.

6.1 The Society Is Ahead of Every Arm at 8 Hours

At 8 matched working hours the society was ahead of CORAL-faithful, the isolated agents, and the single agent on all three problems (Table 8). Its advantage over CORAL-faithful at that hour was 2,191 times the Einstein Arena's minimum improvement on C_1 and 6.99 times on Erdős. Against CORAL-faithful the society was ahead on AHC026

Table 6: **The 913-cycle kernel is built by four agents.** New team bests after the crossover in the main comparison's third archive seed (selected rows; every listed attempt is a new team best). Minutes since launch; the parent's author in parentheses; the move as the attempt's message records it.

Min	Agent	Attempt	Cycles	Parent	Move
195	3	A55	928	A51 (3)	own annealer, new move
222	4	A56	927	A51 (3)	agent 3's kernel and annealer, extended
251	1	A58	924	A57 (4)	constants synthesized on the ALU from one header load
253	3	A59	923	A57 (4)	tree levels 6–7 left in place (relocation knob)
256	4	A60	922	A59 (3)	three-way merge of A59 and A58 on A57, no retune
273	4	A67	917	A66 (4)	adds the select-tree knob from agent 3's A65
320	4	A71	915	A68 (3)	agent 3's code, re-annealed with agent 4's driver
375	1	A73	914	A72 (3)	agent 3's A72 code with agent 4's A71 configuration
394	2	A75	913	A73 (1)	A73's configuration re-annealed on A72's code

Table 7: **The archive is ahead in four further take-home configurations.** Best verified cycles at 8 h (6 h for open weights), single runs; lower is better.

Configuration	Agents	Archive ↓	Compared with
Rerun, c2d-highcpu-56	4 Claude Opus 5.5	**915**	CORAL-faithful 927
Mixed model families	2 Claude Opus 5.5, 2 GPT-6 Astra, 1 Claude Haiku 4.5	**897**	isolated 951
Seeded, 1,006-cycle compiler	4 Claude Opus 5.5, 2 GPT-6 Astra	**877**	isolated 903
Open weights, 6 h	4 Qwen3.5-27B	**4,901**	isolated 5,979

from the first hour and on C_1 from 6.5 hours; against the isolated agents it was ahead on all three problems by 4.0 hours. From 6.5 hours to the end of each comparison it was ahead of both arms on all three problems at every quarter-hour reading.

6.2 60 of 98 New Bests Reused Another Agent's Attempt or Tool

The archive's citations under-count reuse, because a citation records only the items an agent names. Reuse was therefore counted from data flow, by a provenance pass that traced what each agent read and ran, replayed file writes and copies of shared code, and compared each new best's generator with the shared tools by code similarity. Over the society's first 6.7 hours, 60 of its 98 new bests on its four problems reused another agent's attempt or tool (Definition 2.5). Of these, 57 had another agent's attempt in their data flow and 38 ran or adapted a teammate's tool; all 13 strong detections checked by hand were correct, 12 of them meaningful use, and the agents' own citations credited 47. On Erdős, at one checkpoint 17 of the society's 21 new bests ran another agent's tool.

The lineage share, 57 of 98, is close to the 66% of CORAL's new records that came from a cross-agent parent [28]; the tool share, 38 of 98, is invisible to a parent count. Reuse has a price and a delay. Reading and writing the archive took 5.6 to 17.3% of the society's tool calls, and a teammate's new best was first built on after a median of 48.6 to 82.0 minutes, depending on the problem. Both are paid from the first session, and the return arrives after the crossover.

6.3 The Erdős Construction Crosses Two Separately Evolved Solutions

The minimum-overlap construction ends a lineage of 19 verified attempts by three agents, and each later segment ran code written by an earlier agent (Table 9). Agent 2 opened by upsampling the Einstein Arena record and wrote `lse_anneal`, a log-sum-exp continuation optimizer: L-BFGS-B over $h \in [0,1]^n$, an augmented Lagrangian for $\int h = 1$, FFT gradients, and a smoothing parameter β annealed geometrically. Agent 3 built a mirror-symmetric start with that annealer and broke the symmetry with antisymmetric noise. Agent 2 replaced the noise with the exact Hessian's unstable eigen-directions, 75 to 85 of them, and annealed one draw along each; its tool note reads "Run all modes: mode quality is not predicted by the eigenvalue." One of 84 modes gave 0.3808570146 at 7.42 hours.

Table 8: **Three Claude Opus 5.5 agents per problem on the archive are ahead of every arm at 8 matched working hours.** C_1 and Erdős: fast-evaluator margin below the Einstein Arena record at launch ($\times 10^{-6}$); AHC026: development score on 150 generated cases. Higher is better in every score column. All arms Claude Opus 5.5 at xhigh effort, one n2d-highcpu-96 machine per arm, one run each. $t_\times$: crossover time of the society against the CORAL-faithful and isolated arms, in working hours on a 0.25-hour grid. Fast-evaluator scores are never better than the exact score; the society's later C_1 constructions score lower than the bound of Section 6.4, which is the one recomputed in exact arithmetic.

	C_1 ($\times 10^{-6}$)		Erdős ($\times 10^{-6}$)		AHC026 (dev)	
Arm (Claude Opus 5.5, xhigh)	8 h ↑	$t_\times$	8 h ↑	$t_\times$	8 h ↑	$t_\times$
Society, shared verified archive	**38.12**	–	**1.602**	–	**1,433,414**	–
CORAL-faithful	16.21	6.5	0.904	4.0	1,431,607	1.0
Isolated agents	3.51	2.5	0.558	4.0	1,431,266	1.0
Single agent, conversation meant to last the run	28.17	–	0.172	–	1,431,767	–

Table 9: **The minimum-overlap construction passed between agents four times.** Its declared lineage of 19 verified attempts in five segments by agent; working hours of the segment's attempts; fast-evaluator bound at the segment's last attempt.

Segment	Agent	Attempts	Hours	Bound	Move
1	agent 2	4	0.32–1.11	0.3808582070	Upsample the Einstein Arena record; log-sum-exp continuation from a low starting β; coarse-to-fine population at $n = 1{,}792$
2	agent 1	1	1.63	0.3808581437	Integrator session: apply another lineage's upsample-and-re-anneal recipe to the team best
3	agent 3	4	3.89–5.78	0.3808575352	Symmetrize with agent 2's annealer, then break symmetry with antisymmetric noise near the center
4	agent 2	4	6.09–7.42	0.3808570146	Break symmetry along unstable eigen-directions of the exact Hessian
5	agent 3	6	8.71–10.82	0.3808567389	Memetic genetic algorithm with agent 2's tool, seeded from segment 4's construction; slow-cooling refinement; crossover of two separately evolved solutions

The last step combined two separately evolved solutions, and neither was the team best. The archive recorded the lineage's best, with 268 near-active shifts and 545 cells at $h = 1$, as the parent. Replaying the algorithm's seeded draws identifies two others. One is a solution that agent 3 had evolved separately at $n = 1{,}792$ from agent 2's construction with agent 2's genetic-algorithm tool, then upsampled with agent 2's annealer, and never submitted to the archive. The other is agent 3's earlier attempt, mirrored, two improvements behind the team best. A uniform-block crossover took blocks from each, mirrored into one orientation, and a re-anneal from low β smoothed the joins. The child had no cells at $h = 1$ and 326 near-maximal shifts, and the bound fell 1.19×10^{-8} below the lineage's previous best. A restart from the team best at that moment keeps neither parent.

6.4 Two Exactly Verified Bounds

The C_1 construction, found at 3.97 working hours on 262,144 cells, lies 3.05×10^{-6} below the Einstein Arena's first-place entry as of October 30, 2026, 305 times the arena's minimum improvement, and is below every published or listed value at that date (Table 10). The Erdős construction, found at 10.8 working hours on 3,584 cells, lies 1.84×10^{-6} below the Einstein Arena record at launch, 18.4 times its minimum improvement, and 3.55×10^{-8} below MILO's printed value (Table 10).

Each exact checker was written from its problem's definition. The C_1 checker packs the 262,144 heights into one integer in base 2^{216}, wide enough that no coefficient carries, and one integer square recovers all 524,287 autoconvolution coefficients exactly. The Erdős checker sums the overlap in integers at every shift and rechecks the peak with exact fractions. The constructions and verifiers are in the companion report [9].

Table 10: **Upper bounds on the two constants**; lower is better. Published values with their sources; the Einstein Arena records at launch, and the C_1 first place on October 30, 2026. The values of this paper are exact rationals recomputed with independent exact checkers.

Source	Erdős minimum overlap ↓	First autocorrelation, C_1 ↓
AlphaEvolve [26]	0.380924	1.5053
TTT-Discover [37]	0.380876	1.50286
Hill Sampling, gpt-oss-20b [3]	0.38088232	–
HarnessTTT, best of four configurations [21]	0.380868	1.503778
The Station [8]	0.380895	1.504473
MILO [16]	0.3808567744	1.5027435984
Einstein Arena record at launch, October 7 [11]	0.3808585748578584	1.5027435319539597
Einstein Arena #1, October 30 (C_1)	–	1.502743486709583
This paper	0.3808567388833174…	**1.5027404359227669**…

6.5 No Forgetting at All: A Lead Agent with One Whole-Run Conversation

A second open-problem design keeps even the conversation. In the lead-thread society, one lead agent per problem holds one conversation for the whole run, with unlimited resumes and a crash that resumes the same session, and receives no rotated direction. Two explorers send it their results through the archive, and a compute pool runs its long jobs. It ran once, sealed, from October 8, 2026, with its own CORAL-faithful arm and single agent. From 8.0 working hours to the end of the run at 14.5, it was ahead of both on all three problems at every quarter-hour reading, after the largest two-hour gain on C_1 of any arm in either run, 50.6×10^{-6} between 6.25 and 8.25 hours. Its lead agent never forgot its session, and the team stayed ahead of both arms to the end of the run.

7 Related Work

Forgetting and resets. Tan et al. are the immediate predecessor [32]. A team of four DeepSeek V4.1 Flash agents shares scored candidates as commits, forgets three times per run, and keeps only the best candidate; on the take-home it reaches 1,077 cycles at $40 per run, against 1,172 for four parallel agents and 1,262 for one agent. Their continuation test compares keeping every session and candidate with discarding both, so it cannot say which of the two caused the plateau. One of their own post-forget gains is a reuse: the fresh agent "reads a peer's diff and borrows a small trick." Fork-and-Flush forks an agent into fresh chats that keep its workspace, prunes all but the best fork, and improves on best-of-N by 44.4% on a 13-task normalized average. Flushing alone already lifts a single run from 0.45 to 0.55 [6]. Its forks never see each other. Stateless Language Agents keep candidates and outcomes in the harness and let an Advisor assign experiments, reaching $1{,}112.0 \pm 8.9$ take-home cycles against $1{,}350.0 \pm 134.2$ for CORAL at 10^9 tokens with GPT-5.5 [38]. Its Workers cannot read each other's workspaces. The archive is the nearest relative of this line, and its difference is exact: every agent fetches any verified attempt and any tool directly, and nothing verified is ever deleted.

Memory that is never wiped. CORAL runs long-lived agents around graded attempts, free-form notes, and skills, with heartbeats that resume each agent's session. Four Claude Opus 4.6 agents reach 1,103 take-home cycles against 1,180 for the best of four independent runs, and 66% of its new records come from a cross-agent parent [28]. Its own ablation shows that the kept knowledge matters: one agent without notes and skills ends at 1,601 cycles instead of 1,350. Evolving Agent Teams never forgets either. A supervisor restructures teams over a persistent file system of learnings, history, and artifacts, and with an extended budget reaches a 1.2× mean speedup over the production FlashInfer MLA paged-decode kernel, faster on 36 of 47 workloads [7]. GitSwarm keeps every contribution in a branchable Git repository, where 94.7% of ProgramBench contributions are built upon, and its admission check "enforces task-specific publication requirements, not solution correctness" [30]. The Station archives papers that a reviewer agent accepts, and removing that archive sharply reduced discovery on the kissing problem in dimension 11 [8]. AReaL-RSI puts a leader over specialized researchers with conditional memory sharing and reports 1.9–27.5% lower kernel latency than independent exploration [39]. These systems keep their history and improve with it. The archive adds admission by a protected evaluator, a library of promoted tools beside the attempts, and the hour-by-hour measurement of when the kept history starts to pay.

Evaluator-guided search. FunSearch and AlphaEvolve pair a language model with an exact evaluator and a program database [26, 29]; OpenEvolve and ShinkaEvolve are open systems in the same pattern [18, 31]. TTT-Discover and ThetaEvolve train the model at test time [35, 37], HarnessTTT trains a proposer that writes the harness [21], and MILO co-evolves harnesses and the strategy that discovers them [16]. Hill Sampling reduces the loop to one thread conditioned on its best verified program [3]. Each keeps a single search state, so no agent starts from another agent's attempt. Hill Sampling's finding describes the first hours of Figure 1, when every arm refines the same few improvements.

Teams and information. The best of independent samples is a strong baseline [5, 20, 34], and the isolated arm here is that baseline with four long-running agents. Kim et al. find that multi-agent gains shrink as single-agent ability rises [17], and Tran and Kiela find single agents ahead of multi-agent systems on multi-hop reasoning at equal thinking-token budgets [33]; both compare teams at one budget. TeamGym measures the price of splitting decision-critical facts across agents, 22.9 percentage points of success, and a further 19.9 when agents must retrieve the facts themselves [27]. The archive's coordination share and build-on delay are the analogous price in a discovery task, paid before the crossover. CrossProfile finds that group memories tagged with how many agents stated a fact mislead, and that removing the tags raises accuracy by 13 to 24 points [15]; the archive admits an entry by its evaluator outcome, never by how many agents assert it. March framed the trade-off between exploiting shared knowledge and exploring independently [22], and Lazer and Friedman, then Mason and Watts, found that densely connected groups converge fast and explore less [19, 23]. Island models and MAP-Elites counter that convergence inside one population [25, 36].

8 Conclusion

The two take-home searches that stalled did so for different reasons. Several CORAL-faithful agents concluded that about 951 cycles was near the floor and stopped acting. Isolated agents kept only the best of their separate results. Both medians held between 946 and 956 cycles. Agents on a shared archive of verified, executable attempts crossed both at 2.75 hours and reached 914. The late improvements were small, distinct, and found by different agents, and the archive let each agent start from all of them.

Forgetting removes the stalled conclusions and the composable history together. The archive keeps the history and bounds the conversation; the lead-thread society kept both the history and its lead's conversation, and stayed ahead of its run's other arms to the end. In the traces, every combining step ran code or started from an attempt written by another agent. The minimum-overlap construction crossed two solutions, neither of them the team best. The first new best of the GPT-6 Astra successors ran the annealer of an Claude Opus 5.5 agent on a kernel from Claude Opus 5.5. A search that runs for days has to keep each verified attempt with its source, and the tools agents build beside it, where later agents on any model can run both.

Acknowledgments

We thank Einstein Arena, AtCoder, and Anthropic for public tasks and verifiers, the CORAL authors for their code, and the authors of the constructions used to start these searches.

References

[1] Anthropic. Anthropic's Original Performance Take-Home. https://github.com/anthropics/original_performance_takehome, 2026. Public kernel-optimization task scored in simulated machine cycles. GitHub.

[2] AtCoder. Toyota Programming Contest 2023#6 (AtCoder Heuristic Contest 026). https://atcoder.jp/contests/ahc026, 2023.

[3] Jacob Beck, Philip V. Ogren, and Ari Kobren. Hill Sampling for Test-Time Scaling: A Simple and Better Alternative to Repeated Sampling, Evolution, and Training. *arXiv preprint arXiv:2609.25510*, 2026.

[4] Federico Bianchi, Yongchan Kwon, Aneesh Pappu, and James Zou. Harnessing the Collective Intelligence of AI Agents in the Wild for New Discoveries. *arXiv preprint arXiv:2606.10402*, 2026. Introduces the EinsteinArena platform, https://einsteinarena.com.

[5] Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V. Le, Christopher Ré, and Azalia Mirhoseini. Large Language Monkeys: Scaling Inference Compute with Repeated Sampling. *arXiv preprint arXiv:2407.21787*, 2024.

[6] Ziyang Cai, Christos Ziakas, Vasilis Kontonis, Tim Pearce, Siddhartha Sen, Akshay Krishnamurthy, Shivam Garg, and Dimitris Papailiopoulos. Fork-and-Flush: Escaping Idea Basins in Autoresearch Agents, 2026. URL https://arxiv.org/abs/2610.07447v1.

[7] Shiyi Cao, Ziming Mao, Dacheng Li, Joseph E. Gonzalez, and Ion Stoica. Evolving Agent Teams, 2026. URL https://arxiv.org/abs/2610.36848v1.

[8] Stephen Chung, Wenyu Du, and William J. Wesley. Autonomous Mathematical Discovery in an Open-World Multi-Agent Environment. *arXiv preprint arXiv:2608.23691*, 2026.

[9] Lindell S. Cumes. A New Upper Bound for the First Autocorrelation Inequality and 306 Circle-Packing Records via Active-Set Escape from Equioscillating and KKT Optima. QWNTL Labs preprint, https://qwntl.com/paper-new-constructions, 2026.

[10] Lindell S. Cumes. Evidence-Gated Recursion: An Experiment on a Self-Governing Research Organization of Frontier Language-Model Agents. QWNTL Labs; alphaXiv 2610.nenya-headroom-law-recursive-harnesses (v2), https://qwntl.com/frontier, 2026.

[11] EinsteinArena. EinsteinArena: An Open Arena Where AI Agents Collaborate and Compete on Unsolved Science Problems. https://einsteinarena.com, 2026. Problems with public verifiers and a live leaderboard with minimum-improvement thresholds. Leaderboard accessed 2026-10-13.

[12] Mahdi Farahbakhsh, Ilan Sela, Fatemeh Doudi, Vishnu Teja Kunde, Krishna Narayanan, Jean-Francois Chamberland, and Dileep Kalathil. AgentDiscover: Autonomous Discovery with Minimal Search Scaffolding, 2026. URL https://arxiv.org/abs/2610.05334v1.

[13] Jan Kristian Haugland. The minimum overlap problem revisited. *arXiv preprint arXiv:1609.08000*, 2016.

[14] Yuki Imajuku, Kohki Horie, Yoichi Iwata, Kensho Aoki, Naohiro Takahashi, and Takuya Akiba. ALE-Bench: A Benchmark for Long-Horizon Objective-Driven Algorithm Engineering. *arXiv preprint arXiv:2506.09050*, 2025. NeurIPS 2025 Datasets and Benchmarks Track.

[15] Anouar Imel, Imad Charradi, Abderrahman Sghir, and Khalil El Khammar. Rare, Relevant, and Still Wrong: How Memory Fails LLM Group Decisions, 2026. URL https://arxiv.org/abs/2610.37624v1.

[16] Prithwish Jana, Mononito Goswami, Hao Liu, Xinyu Li, Langlin Huang, Zhehui Huang, Zhishen Huang, Patrick Blöbaum, Anoop Deoras, Purak Jain, Nikos Kanakaris, and Sahika Genc. MILO: Automated Harness Discovery via Orchestrated Multi-Agent Evolution. *arXiv preprint arXiv:2609.38349*, 2026. Project page: https://jprithwish.github.io/MILO/.

[17] Yubin Kim, Ken Gu, Chanwoo Park, Chunjong Park, Samuel Schmidgall, A. Ali Heydari, Yao Yan, Zhihan Zhang, Yuchen Zhuang, Yun Liu, Mark Malhotra, Paul Pu Liang, Hae Won Park, Yuzhe Yang, Xuhai Xu, Yilun Du, Shwetak Patel, Tim Althoff, Daniel McDuff, and Xin Liu. Towards a Science of Scaling Agent Systems. *arXiv preprint arXiv:2512.08296*, 2025.

[18] Robert Tjarko Lange, Yuki Imajuku, and Edoardo Cetin. ShinkaEvolve: Towards Open-Ended And Sample-Efficient Program Evolution. *arXiv preprint arXiv:2509.19349*, 2025. Code: https://github.com/SakanaAI/ShinkaEvolve.

[19] David Lazer and Allan Friedman. The Network Structure of Exploration and Exploitation. *Administrative Science Quarterly*, 52(4):667–694, 2007. doi: 10.2189/asqu.52.4.667.

[20] Junyou Li, Qin Zhang, Yangbin Yu, Qiang Fu, and Deheng Ye. More agents is all you need. *arXiv preprint arXiv:2402.05120*, 2024.

[21] Yingzi Ma, Yulong Cao, Nanxi Li, Fang Wu, Yizhe Zhang, James Zou, Yejin Choi, and Chaowei Xiao. HarnessTTT: Learn to Optimize Harness at Test-Time for Scientific Discovery. *arXiv preprint arXiv:2610.17370*, 2026.

[22] James G. March. Exploration and Exploitation in Organizational Learning. *Organization Science*, 2(1):71–87, 1991. doi: 10.1287/orsc.2.1.71.

[23] Winter Mason and Duncan J. Watts. Collaborative learning in networks. *Proceedings of the National Academy of Sciences*, 109(3):764–769, 2012. doi: 10.1073/pnas.1110069108.

[24] Máté Matolcsi and Carlos Vinuesa. Improved bounds on the supremum of autoconvolutions. *Journal of Mathematical Analysis and Applications*, 372(2):439–447.e2, 2010. doi: 10.1016/j.jmaa.2010.07.030.

[25] Jean-Baptiste Mouret and Jeff Clune. Illuminating search spaces by mapping elites. *arXiv preprint arXiv:1504.04909*, 2015.

[26] Alexander Novikov, Ngân Vũ, Marvin Eisenberger, Emilien Dupont, Po-Sen Huang, Adam Zsolt Wagner, Sergey Shirobokov, Borislav Kozlovskii, Francisco J. R. Ruiz, Abbas Mehrabian, M. Pawan Kumar, Abigail See, Swarat Chaudhuri, George Holland, Alex Davies, Sebastian Nowozin, Pushmeet Kohli, and Matej Balog. AlphaEvolve: A coding agent for scientific and algorithmic discovery. *arXiv preprint arXiv:2506.13131*, 2025.

[27] Chanwoo Park, Jaywoong Jeong, Zachary Wojtowicz, Yubin Kim, Taekyun Lee, Kaiqing Zhang, and Asuman Ozdaglar. TeamGym: A Gym for Team Decision-Making by AI Agents with Distributed Information, 2026. URL https://arxiv.org/abs/2610.37571v1.

[28] Ao Qu, Han Zheng, Zijian Zhou, Yihao Yan, Yihong Tang, Shao Yong Ong, Fenglu Hong, Kaichen Zhou, Chonghe Jiang, Minwei Kong, Jiacheng Zhu, Xuan Jiang, Sirui Li, Cathy Wu, Bryan Kian Hsiang Low, Jinhua Zhao, and Paul Pu Liang. CORAL: Towards Autonomous Multi-Agent Evolution for Open-Ended Discovery. *arXiv preprint arXiv:2604.01658*, 2026. COLM 2026 (per the authors' repository). Code: https://github.com/Human-Agent-Society/CORAL.

[29] Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Matej Balog, M. Pawan Kumar, Emilien Dupont, Francisco J. R. Ruiz, Jordan S. Ellenberg, Pengming Wang, Omar Fawzi, Pushmeet Kohli, and Alhussein Fawzi. Mathematical discoveries from program search with large language models. *Nature*, 625(7995): 468–475, 2024. doi: 10.1038/s41586-023-06924-6. FunSearch.

[30] Vedant Shah, Ankur Samanta, Paras Dahal, Mikhail Plekhanov, Carole-Jean Wu, Scott Yih, Remi Munos, Rob Fergus, Jakob Foerster, Ruslan Salakhutdinov, Sanjeev Arora, Jason Weston, Aaron Courville, and Anirudh Goyal. GitSwarm: Decentralized Compounding Inference, 2026. URL https://arxiv.org/abs/2610.04862v1.

[31] Asankhaya Sharma. OpenEvolve: An Open-Source Evolutionary Coding Agent. https://github.com/algorithmicsuperintelligence/openevolve, 2025. Open-source implementation of AlphaEvolve. Software, GitHub.

[32] Shangyin Tan, Ben Zhou, Lakshya A Agrawal, Jialin Zhang, Yilin Chen, Qiuyang Mang, Donghyun Lee, Dennis Jacob, Koushik Sen, Alex Dimakis, and Matei Zaharia. Scaling Discovery Requires Forgetting, 2026. URL https://arxiv.org/abs/2610.20563v1.

[33] Dat Tran and Douwe Kiela. Single-Agent LLMs Outperform Multi-Agent Systems on Multi-Hop Reasoning Under Equal Thinking Token Budgets. *arXiv preprint arXiv:2604.02460*, 2026.

[34] 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. ICLR 2023.

[35] Yiping Wang, Shao-Rong Su, Zhiyuan Zeng, Eva Xu, Liliang Ren, Xinyu Yang, Zeyi Huang, Xuehai He, Luyao Ma, Baolin Peng, Hao Cheng, Pengcheng He, Weizhu Chen, Shuohang Wang, Simon Shaolei Du, and Yelong Shen. ThetaEvolve: Test-time Learning on Open Problems. *arXiv preprint arXiv:2511.23473*, 2025.

[36] Darrell Whitley, Soraya Rana, and Robert B. Heckendorn. The Island Model Genetic Algorithm: On Separability, Population Size and Convergence. *Journal of Computing and Information Technology*, 7(1):33–47, 1999.

[37] Mert Yuksekgonul, Daniel Koceja, Xinhao Li, Federico Bianchi, Jed McCaleb, Xiaolong Wang, Jan Kautz, Yejin Choi, James Zou, Carlos Guestrin, and Yu Sun. Learning to Discover at Test Time. *arXiv preprint arXiv:2601.16175*, 2026. TTT-Discover. Code: https://github.com/test-time-training/discover.

[38] Qizheng Zhang, Changxiu Ji, Isaac Sun, Yuetai Li, Shubhangi Upasani, Sherry Ruan, Boyuan Ma, Fenglu Hong, Vamsidhar Kamanuru, Yoonho Lee, Yuzhen Mao, Genghan Zhang, Rulin Shao, Qiuyang Mang, Andy Dimnaku, Changran Hu, Radha Poovendran, and Kunle Olukotun. Stateless Language Agents: Scaling Long-Horizon Automated Research, 2026. URL https://arxiv.org/abs/2610.07625v1.

[39] Wenhao Zhou, Jiajun Song, Haitao Wang, Zipeng Qiu, Yingqi Peng, Ruida Xu, Jiale Li, Wenshuang Li, Shusheng Xu, Wei Dong, Hao Dai, Tongkai Yang, Yi Gao, and Binhang Yuan. AReaL-RSI: Adaptive Scaling and Coordination of Recursive Self-Improvement. *arXiv preprint arXiv:2610.18083*, 2026.

A Experimental Details

This paper is one of five companion papers that report in full the results of a research organization of language-model agents; a separate paper reports the organization itself [10]. This paper re-analyzes the existing archives of the runs.

A.1 Harness Parameters

Take-home archive. Insights are compiled by Claude Opus 5.5; failures are distilled by Claude Haiku 4.5. A plateau is declared after 15 attempts without a new team best. New team bests are re-scored at $k = 64$. A session is resumed at most three times before a fresh one. Parents are chosen by clade credit, the best result and improvements among an attempt's descendants, with a diversity spread across agents. Briefings list at most 14 insights and 8 constraints.

Society. The part of a briefing that the harness assembles is at most 9,000 characters: the team best, a parent drawn by Thompson sampling over clade credit, a direction, and at most 10 insights, 4 recent results, 6 constraints, 8 tools, and 4 offers from the other problems. Directions rotate each session. The society declares a plateau after 15 attempts without gain, integrates every 6th session, and consolidates every 40 attempts. The number of islands is $\max(1, \min(4, \lfloor \text{agents}/2 \rfloor))$, one for three agents.

Lead-thread society. A pool job's CPU weight is 100, multiplied by 4 if the lead submitted it and by 3 if it starts from the team best's lineage, re-evaluated every 30 seconds. Explorers hold a direction for two sessions, take parents from their own line or the lead's attempts of the last 3 hours, and receive team digests at most once per 30 minutes.

CORAL-faithful. Upstream commit `cce3101` (MIT license), pinned by the SHA-256 of nine upstream files. `consolidate.md` is byte-identical to upstream; `reflect.md` changes one command; `pivot.md` changes command names and disables web access; the session prompt is upstream's template with the command-line interface swapped, web access disabled, and the heartbeat-customization section replaced by the fixed defaults: reflect after every evaluation, consolidate every 10 evaluations across agents, and pivot after 5 without improvement. Heartbeats arrive in the submission's response instead of by interrupt. The task text is upstream's kernel-task description, hints included. Attempts live in the archive's event log instead of a shared Git checkout, and the shared run server sorts attempts by the task objective, a fix applied to every arm. An early exit resumes the same conversation with "Eval #N: score=… Continue optimizing."; after 1,000 resumes, where upstream has no limit, a fresh session starts. Unlike upstream, the port sets no per-session turn cap. Outside the take-home, its configuration contains the problem statement and factual tips only.

A.2 Machines and Runs

Each open-problem arm used a 96-vCPU AMD EPYC Milan machine with one systemd unit per problem and equal CPU weight; under saturation each problem received about a quarter of the 96 cores, which the single agent had to itself and three agents divided in the other arms. The society started on October 7, 2026, at 20:31 UTC, the CORAL-faithful and isolated arms at 23:31 UTC, and the single agent at 17:38 UTC. For C_1 heights a_i on n equal cells, the objective is exactly $2n \max_k c_k / S^2$ with $c_k = \sum_{i+j=k} a_i a_j$ and $S = \sum_i a_i$, and the reported margin is $(r - B_H(t)) \times 10^6$ below the Einstein Arena record at launch, r.

B Prompts

Every take-home agent, in every arm, received the same task prompt. Angle brackets mark fields the harness fills. In the archive's briefing and resume prompts, the agent's label and the shared directory's path are generalized as `agent <agent>` and `<shared dir>`. CORAL-faithful appends an adapted version of upstream CORAL's session prompt at commit `cce3101`, with the task's name, description, candidate file, tips, and score direction filled in.

Task prompt (every take-home agent).

```
You are optimizing a kernel for a simulated VLIW SIMD machine: Anthropic's original performance
engineering take-home. Your working directory is your own copy of that repository. Read
`Readme.md`, `problem.py` (the machine and the reference kernel) and `perf_takehome.py` (the
kernel builder you optimize).
```

```
Goal: minimize the cycle count of the program produced by `KernelBuilder.build_kernel`, measured
on the frozen simulator, while the program's output values stay correct. The starting kernel
takes 147,734 cycles. Lower is better; there is no target to stop at.

Rules
- Edit `perf_takehome.py`. You may create scratch files and scripts in your working directory.
  Do not modify `tests/` or `problem.py`. A candidate is a single `perf_takehome.py` that uses
  only the Python standard library.
- Scoring happens only through the protected evaluator:
  `kx submit perf_takehome.py -m "<what changed and why>" [-p <parent attempt id>]`.
  It builds your kernel in a sandbox, runs it on fresh random inputs, and reports correctness,
  cycles and per-engine utilization diagnostics. Name the attempt you started from with `-p`.
  Local checks are available and fast: `python3 perf_takehome.py Tests.test_kernel_cycles` and
  `python3 tests/submission_tests.py` (they use your workspace copy; the protected evaluator is
  the authority).
- No internet access: do not fetch anything from the web.
- Keep working: after each result, decide the next improvement and continue. Submit every
  candidate you believe is correct and possibly better. `kx --help` lists the other commands.
```

Memory section and resume nudge (best of four).

```
Memory
- `kx list`, `kx best`, `kx show <id>` and `kx source <id>` show your own past attempts, including
  ones from earlier sessions. Your working directory persists across sessions.

Continue: keep improving your score. Check `kx best` and `kx list -n 10`, decide the
next improvement, implement it, verify it locally, and submit it with `kx submit`.
```

Archive briefing (appended to the task prompt at every fresh session). The PLATEAU line is shown only after 15 attempts without a new team best; each list is omitted when empty.

```
Team engine (shared memory compiled from verified attempts; the protected evaluator checked
everything listed here)
- You are agent <agent>, one of <k> agents working in parallel (<agent>=<model>, ...). Each
  session starts fresh; the store below carries the team's progress between sessions and models.
- Team best: <id>, <cycles> (agent <agent>, <model>). Diagnostics: <per-engine utilization>. Its
  source is <shared dir>/best/perf_takehome.py (or `kx source <id>`).
- Your assigned starting point: <parent> (<reason>). Start from it with `kx source <parent> -o
  perf_takehome.py` unless you have a concrete better plan; name your actual parent with `-p`
  when you submit.
- PLATEAU: the team best has not improved in <n> attempts. Incremental tuning of the current
  approach is exhausted. Change the structure: a different decomposition, a different engine
  balance, a new algebraic identity, or a new tool. Strategies already explored: <labels>.

Verified insights (largest gains first):
- <id> (gain <delta> cycles, <from>-><to>): <title>
  <body>
  Premises: <premises>
- <id> (UNVERIFIED, proposed by <agent>): <title>
  <body>

Constraints distilled from failed attempts:
- <constraint> (from <attempt ids>)

Tools (<shared dir>/tools = promoted; <shared dir>/tool_candidates = proposed):
- tool:<name> [<status>] <description>

Protocol
- Cite what you build on: `-p <parent id>`, and in `-m` name any insights (e.g. N4) and tools
  (e.g. tool:sched) you used. Cited candidates become verified when your attempt improves.
- When you learn something reusable, record it: `kx note --kind insight --title ... --body ...
  --ref <attempt ids> --premise <condition>` (stays unverified until an attempt that cites it
  improves).
- When you build reusable code (a scheduler, an emitter, an analysis or search script), propose
  it so every teammate and later model can use it: `kx x tool_propose '{"name": "NAME", "path":
  "<file or dir in your working directory>", "description": "...", "usage": "..."}'`.
- `kx x insights` and `kx x tools` show the current store at any time.
```

Archive resume prompt (at most three per conversation).

```
Continue improving the score.
Team best is now <id> at <cycles> (agent <agent>); source at
<shared dir>/best/perf_takehome.py.
New in the team store since your session started:
- <id> [<insight|tool|constraint>] <title>: <body>
Decide the next improvement, implement it, verify it locally, and submit with `kx submit`, citing
parents, insights and tools.
```

Insight compiler (Claude Opus 5.5, system prompt).

```
You compile verified experience for a team of agents. <task brief>
You are given a parent candidate and a child candidate that the protected evaluator verified as
correct and better. Explain precisely what change produced the gain and why, so another agent
(possibly a different model) can reuse it. Be concrete.
Reply with one JSON object only, no prose, with keys:
"title" (<= 12 words), "claim" (what changed, 1-3 sentences), "mechanism" (why it improves the
score), "premises" (list of conditions the gain depends on), "reuse" (how to apply it further or
what to try next), "strategy" (2-5 word label for the child's overall kernel strategy). Keep each
field under 120 words.
```

Failure distiller (Claude Haiku 4.5, system prompt).

```
You distil a failed attempt by an agent on an optimization task into one reusable
constraint for teammates: what breaks correctness (or the build) and why, stated so that others
avoid it. Reply with one JSON object only: {"constraint": "<one sentence>", "confidence":
"low|medium|high"}. Use low if the cause is not evident from the diff and error.
```

C Additional Results

Table 11: **The archive's advantage after the crossover**, main comparison. Difference of seed medians in cycles (positive: the archive is ahead), with the least and most favorable of the nine seed pairings. Every archive seed was ahead of every baseline seed at each of these checkpoints, and the exact one-sided permutation test gave $p = 0.05$ at each, the minimum for three seeds against three.

Compared with	3 h	4 h	5 h	6 h	7 h	8 h
CORAL-faithful	20 [11, 24]	22 [19, 26]	30 [15, 35]	32 [9, 36]	33 [8, 37]	34 [9, 37]
Best of four	23 [8, 28]	18 [14, 29]	27 [21, 36]	30 [5, 36]	32 [4, 37]	32 [5, 37]

Table 12: **Compute by role**, main comparison, totals over three seeds per arm. Agent tokens are summed from each session's usage record; output tokens are a lower bound, because sessions ended by the 4-hour cap or the deadline report no final usage. USD at the list prices of Table 5.

Role	Model	Archive	CORAL-faithful	Best of four
Agents: turns	Claude Opus 5.5	12,500	15,982	9,003
Agents: output tokens	Claude Opus 5.5	239,074	357,894	412,963
Agents: cache-read tokens	Claude Opus 5.5	3,329,718,135	5,684,578,258	1,947,952,938
Agents: cache-write tokens	Claude Opus 5.5	126,584,228	114,354,812	113,554,078
Agents: USD	Claude Opus 5.5	2,462	3,566	1,694
Compiler and distiller: calls	Claude Opus 5.5, Claude Haiku 4.5	206	–	–
Compiler and distiller: tokens in/out	Claude Opus 5.5, Claude Haiku 4.5	1,618,888/169,892	–	–
Evaluator: attempts at $k = 8$	–	211	168	201
Evaluator: re-scorings at $k = 64$	–	105	–	–

Table 13: **Resources each seed spent to first reach a score**, main comparison. Each cell lists the three seeds: attempts (evaluator calls) and agent spend in USD at the list-price table of Table 5, pro-rated within sessions by wall time. A dash means the seed never reached the score in 8 hours.

	Shared archive		CORAL-faithful		Best of four	
Cycles $\leq$	attempts	USD	attempts	USD	attempts	USD
960	22, 27, 35	180, 89, 227	29, 28, 33	83, 111, 127	30, 36, 39	105, 234, 139
950	26, 39, 37	231, 232, 238	40, 49, 57	1,002, 578, 505	44, 57, 55	171, 439, 232
940	30, 43, 40	286, 256, 262	–, 51, –	–, 658, –	65, –, –	378, –, –
930	39, 55, 49	393, 401, 280	–, 58, –	–, 973, –	67, –, –	385, –, –
920	53, 62, 63	583, 584, 431	–, –, –	–, –, –	–, –, –	–, –, –

Table 14: **Per-seed best verified cycles at 8 h**, main comparison and pooled runs on n2d-highcpu-48; lower is better.

Arm	Main comparison, seeds 1–3	Other runs on the same machine type
Shared verified archive	914, 916, 913	910 (original run), 911 (model-change run, first 8 h)
CORAL-faithful	950, 925, 948	928 (original run)
Best of four isolated	921, 950, 946	947 (original run), 921 (added seed)

The Erdős bound's exact value is

$$974993251541292596612630004798111806 03/(256 \cdot 10^{36}) = 0.38085673888331742055\ldots,$$

and the C_1 bound's is $1.50274043592276697641\ldots$.