HUMANIZE MLE AGENTS: MULTI-AGENT SESSION ALTERNATION BEYOND SINGLE-AGENT LIMITS

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing LLM-based agent approaches for long-horizon machine learning engineering (MLE) often use context engineering, implemented through custom harnesses that manage tools, skills, and accumulated experience. However, we find that as single-session execution lengthens, the agent's reasoning intensity declines alongside slowing progress, summarized as *long-horizon deceleration*, and the benefits of these custom harnesses vanish as the underlying model improves, summarized as *harness staleness*. These findings suggest that their benefits may not persist across versions, which motivates a workflow-level approach in which agents build on one another's work across sessions. We model these alternating agent sessions as a *context-reset semi-Markov decision process* (**CR-SMDP**) and establish sufficient conditions for alternation to exceed single-agent reference limits, derive an explicit experiment-budget bound, and characterize when session caps preserve this advantage. Guided by this analysis, we introduce *Humanize MLE Agents* (**HMA**), which alternates two agents on a shared workspace, limits each session to at most k experiments and resets the context between sessions, without additional task-specific harness or prior knowledge supplements. Across 75 Kaggle tasks in MLE-bench, **HMA** reaches a *state-of-the-art* any-medal rate of 78.2%, exceeding the strongest open-source baseline by 8.0 points with a quarter of its time budget. Moreover, among six configurations spanning five heterogeneous agent pairs, HMA improves any-medal rate over the constituent baselines by 6.4 points on average, demonstrating cross-pair generality.

1 INTRODUCTION

LLM-based agents tackle machine learning engineering (MLE) through repeated experiments: writing training code, running it, and evaluating candidate solutions (Chan et al., 2025; Toledo et al., 2025; Karpathy, 2026). For long-horizon tasks, producing a working solution is only the beginning: the agent must keep finding ways to improve it. As further gains become harder to find, the agent is expected to spend more tokens reasoning about the limitations of the current solution and deciding the next action. In our experiment across six models, however, mean output tokens per response—a proxy for reasoning intensity—decline over time as score gains diminish (Figure 2). We refer to this phenomenon as *long-horizon deceleration* and analyze its implications for prior agent design.

Existing approaches often seek to sustain improvement through context engineering, supplying Markdown instructions, skills, and accumulated knowledge to guide an agent's next steps (Zhu et al., 2026; Du et al., 2026; Karpathy, 2026; Rajasekaran et al., 2025). However, guidance that helps one model need not help its successor. For example, OpenAI recommends rewriting `AGENTS.md`, skills, and task prompts when upgrading models (Provencher, 2026), while Anthropic advises removing instructions for checks that newer models can already perform on their own (Anthropic, 2026a). As model capabilities improve, guidance that once compensated for their limitations can thus become redundant or counterproductive. We observe a related pattern in our comparison of three custom MLE harnesses—systems that manage a model's tools, context, and execution loop (Zhu et al., 2026; Du et al., 2026; Zhao et al., 2026). Nevertheless, our comparison shows that the relative benefits of specialized harnesses diminish or reverse as the base model improves, even under longer execution budgets (Table 1). We refer to this decline in relative performance as *harness staleness*, which indicates the limitations of custom harness design.

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.

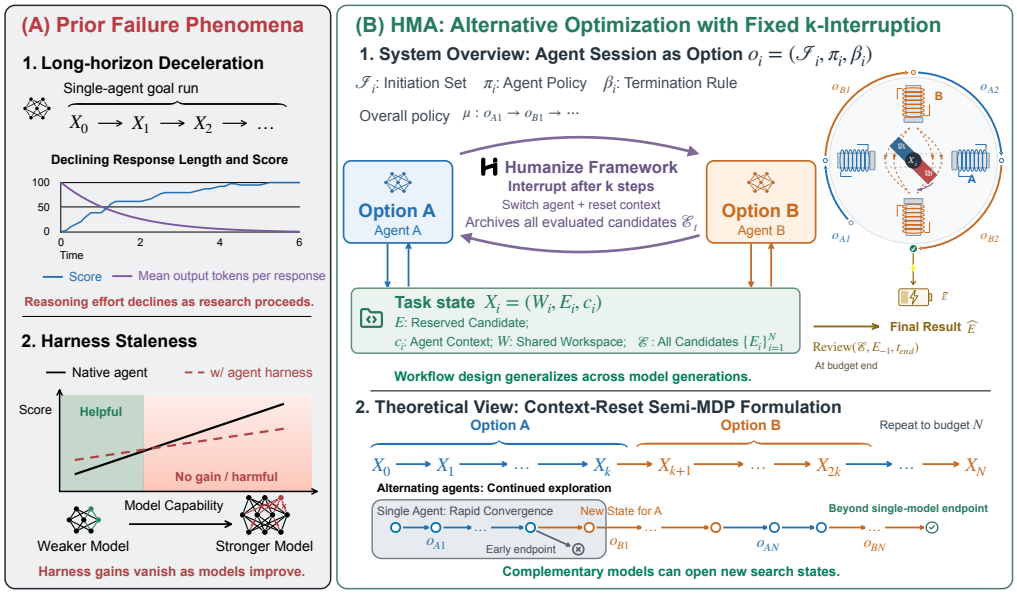

Figure 1: **HMA overview. (A)** Declining output tokens per response accompany diminishing score gains, indicating decelerating single-agent exploration; harness gains can vanish or reverse as models improve. **(B)** HMA alternates native agents after at most k experiments, resetting context while preserving shared work and candidates for final review. The motor analogy illustrates options A and B alternately energizing shared state X_i toward the final result $\widehat{E}$. The CR-SMDP view captures how complementary options can extend exploration beyond single-agent limits.

These limitations motivate workflow-level coordination across *agent sessions*, each defined as an agent's execution from context initialization to termination. In MLE, code, trained models, and evaluated candidates persist in a shared workspace, allowing successive agents to build on prior work (Zhao et al., 2026; Young, 2025). This persistence enables complementary roles: the agent that discovers a useful candidate need not be the one best suited to refine it, and subsequent changes can provide a new starting point when the first agent returns. We therefore ask: *Can structured alternation convert complementary agent capabilities into a consistent advantage over the best single agent under a matched budget?*

To address this question, we represent alternating agent sessions as a *context-reset semi-Markov decision process* (**CR-SMDP**). We establish sufficient conditions and derive experiment-budget bounds under which alternation exceeds single-agent reference limits, and identify when fixed-k interruption retains this advantage (Sections 3.1–3.3). Building on these results, we introduce *Humanize MLE Agents* (**HMA**), which coordinates two native agents—complete model—harness pairs using official harnesses—through a shared workspace (Figure 1). Each session is capped at k experiments; after each handoff, the other agent continues with the shared work and a reset context. The workflow leaves each agent's tools, context management, and execution loop unchanged (Bolin, 2026; Anthropic, 2026b), and provides no task-specific knowledge at initialization. In summary, our contributions are:

- **We identify single-agent limitations and show how HMA sustains exploration.** We characterize *long-horizon deceleration* across six models and *harness staleness* across three custom MLE harnesses in Sections 2. In contrast, from the early to the late phase of execution, HMA retains 86.4% of its mean output tokens per response, compared with 60.7% for matched single-agent baselines, indicating more persistent exploration across sessions.

- **Alternation and workflow design extend single-agent limits.** Our CR-SMDP analysis establishes sufficient conditions and experiment-budget bounds for exceeding single-agent reference limits (Theorem 3.3), and characterizes when fixed-k interruption preserves this

advantage Theorem 3.5). Algorithm 1 instantiates these principles with context reset, alternating agents, and final review of candidates (Sections 3.1–3.4).

- **HMA achieves state-of-the-art any-medal performance with cross-pair generality.** On MLE-bench, HMA achieves a *state-of-the-art* any-medal rate of 78.2%, exceeding the strongest published open-source workflow by 8.0 points. Across the six heterogeneous agent-pair configurations, HMA improves the any-medal rate over the constituent baselines by 6.4 points on average, with gains in every configuration (Sections 4.2).

2 Motivation: Limits of Single-Agent Exploration

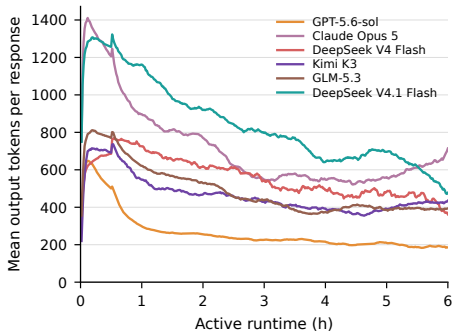

Figure 2: **Single-session exploration decelerates over time.** Each curve reports mean output tokens per recorded model response for one `/goal` configuration, pooling responses across 75 tasks and repeats. Across configurations, mean response length falls by 29.8%–45.0% from hours 0–2 to hours 3–5, indicating slower single-session exploration over time.

Table 1: **Specialized harnesses become less effective as the backbone improves.** We report medal rates on the 16-task MLE-bench subset, compared on DeepSeek-V4-Flash with DeepSeek-V4.1-Flash. The DeepSeek Harness row is the native baseline. Bold and underlined entries indicate the largest and second-largest rates within each model column. With the weaker backbone, some specialized harnesses match or exceed the native baseline, whereas the stronger backbone consistently favors the native harness. This gap persists despite the longer execution budgets of the specialized workflows, showing the empirical pattern we name *harness staleness*.

Harness	Time (h)	DeepSeek-V4-Flash Medals (%)	DeepSeek-V4.1-Flash Medals (%)
DeepSeek Harness	6	16.7	**77.1**
ScienceFlow	24	**25.0** +8.3	<u>68.8</u> −8.3
ML-Master 2.0	24	6.3 −10.4	37.5 −39.6
MLEvolve	12	<u>18.8</u> +2.1	50.0 −27.1

In this section, we investigate two limitations of single-agent MLE workflows: diminishing exploration over extended execution and limited transfer of harness improvements across model versions. These observations motivate HMA, which coordinates alternating agents from different providers to extend exploration beyond the capabilities of an individual model–harness system.

Long-horizon deceleration. We examine six single-session configurations under `/goal` mode. Figure 2 reports mean output tokens per response in a trailing 30-minute window, used as a proxy for the agent's reasoning intensity. Across configurations, response length decreases by 29.8%–45.0% from hours 0–2 to hours 3–5. Together with the rapid saturation of prefix gains in Figure 3, this reveals diminishing returns from continued single-agent exploration; detailed measurement definitions and supporting statistics are given in Appendix D.1.

Harness staleness. To counter diminishing single-agent exploration, existing harnesses introduce planning, review, and context engineering (Young, 2025; Rajasekaran et al., 2025). With task-specific cold-start knowledge disabled for fairness, our comparison of recent MLE harnesses (Zhu et al., 2026; Du et al., 2026; Zhao et al., 2026) reveals *harness staleness*: their benefits relative to the official harness diminish as the backbone improves (Table 1). The stronger performance of the official harness is consistent with a possible benefit from model–harness alignment acquired during training. This motivates treating each agent as an integrated model–harness system and studying how agents from different providers collaborate through workflow-level coordination. We present detailed configurations and results in Appendix D.2.

3 METHOD

In this section, we formulate *Humanize MLE Agents* (**HMA**) as a context-reset semi-MDP (Section 3.1), derive conditions and experiment budgets for alternation to exceed single-agent reference limits (Section 3.2), analyze fixed-k interruption and empirical cap selection (Section 3.3), and present the complete HMA workflow (Section 3.4).

3.1 HMA PRELIMINARIES

Workflow policy and task state. A multi-agent workflow that alternates between agent sessions is governed by a fixed *policy over options* μ (Sutton et al., 1999). Without loss of generality, we consider two agents, A and B, working on a shared task. The state $x = (W, E, c^{(u)})$ comprises the shared workspace W, retained result E, and context $c^{(u)}$ of the active agent $u \in \{A, B\}$. Session i of the workflow, executed by agent u, is modeled as an option $o_i^{(u)} = (\mathcal{I}_i^{(u)}, \pi_i^{(u)}, \beta_i^{(u)})$, with initiation set $\mathcal{I}_i^{(u)} := \{x : \mu(o_i^{(u)} \mid x) > 0\}$, internal policy $\pi_i^{(u)}$, and termination rule $\beta_i^{(u)}$. Termination creates an *option boundary*, where μ selects the next option.

Task initialization and result quality. For a general MLE task, N is the total experiment budget of the workflow; each primitive step completes one experiment and updates (W, E) within an option. The initial state $x_0 = (W_0, E_0, \emptyset) \sim \nu_0$ is drawn from the task-state distribution ν_0, with workspace W_0, empty result E_0, and empty context. Let E_n be the result retained after n experiments, with normalized quality $q(x) := q(E) \in [0, 1]$. Expected quality and target-success probability are $\mathrm{Perf}_\mu(N) := \mathbb{E}_{\mu,\nu_0} q(E_N)$ and $p_\mu(N; \tau) := \Pr_{\mu,\nu_0}(q(E_N) \geq \tau)$ for $\tau \in [0, 1]$.

Workflow trajectory, rewards, and values. The trajectory is $(x_0, o_0^{(u_0)}, x_1, o_1^{(u_1)}, \ldots)$, where session i starts at x_i and selects $o_i^{(u_i)} \sim \mu(\cdot \mid x_i)$ with $u_i \in \{A, B\}$. The option kernel is $P^{(u)}(x, x') := \Pr(x_{i+1} = x' \mid x_i = x, u_i = u)$. With a discount factor $\gamma = 1$ and reward $r_{n+1} := q(E_{n+1}) - q(E_n)$. Let $\bar{r}^{(u)}(x)$ be the expected option reward from x. Values $V^\mu(x)$ and $Q^\mu(x, o_i^{(u)})$ are expected future quality increments, with Q^μ executing $o_i^{(u)}$ first. The semi-MDP Bellman recursion (Bradtke & Duff, 1994; Dietterich, 2000) gives

$$Q^\mu(x, o_i^{(u)}) = \bar{r}^{(u)}(x) + \sum\nolimits_{x'} P^{(u)}(x, x') V^\mu(x'), \quad V^\mu(x) = \sum\nolimits_u \mu(o_i^{(u)} \mid x) Q^\mu(x, o_i^{(u)}). \quad (1)$$

Context-Reset SMDP. The HMA workflow is formulated as a *context-reset SMDP* (CR-SMDP), whose option boundaries preserve (W, E) and reset context to $\emptyset$. For integer cap $k \geq 1$, remaining experiment budget t, and natural duration $T_i^{(u)}(x_i)$ measured in experiments, option i completes $d_i^{(u)} = \min\{T_i^{(u)}(x_i), k, t\}$ experiments before natural termination, fixed-k interruption, or budget exhaustion. If $T_i^{(u)}(x_i) > k$ almost surely and $t \geq k$, then $\beta_i^{(u)}(\ell) = \mathbf{1}_{\{\ell \geq k\}}$ and $d_i = k$, where ℓ counts completed experiments within the option. Further formulation details and early natural termination analysis appear in Appendix A.1.

3.2 ALTERNATION BEYOND SINGLE-AGENT LIMITS

For $u \in \{A, B\}$, let $\mathfrak{B}^{(u)}$ be a finite set of single-agent workflow variants of u with ordinary or specified stronger harnesses. Define the reference workflow family $\mathfrak{B}_{A,B} := \mathfrak{B}^{(A)} \cup \mathfrak{B}^{(B)}$. All workflows initialize from $x_0 \sim \nu_0$ and satisfy the following retention assumption.

Assumption 3.1 (Retention of the best result)**.** $q(E_{n+1}) \geq q(E_n)$ for all n.

We adopt the fixed-k termination rule from Section 3.1. For simplicity, assume the alternating policy $\mu_k = (A, B, u_0, k)$ starts with $u_0 \in \arg\max_{u \in \{A,B\}} \mathrm{Perf}_{(A,B,u,k)}(N)$, where $\bar{A} = B$ and $\bar{B} = A$. With option kernels $P_k^{(u)}$, including context reset, define the cycle kernel and reference ceiling:

$$M_{\mu_k} = P_k^{(u_0)} P_k^{(\bar{u}_0)}, \qquad U_* := \max_{b \in \mathfrak{B}_{A,B}} \lim_{N \to \infty} \mathrm{Perf}_b(N). \quad (2)$$

On each task, let $\mathcal{X}_{\text{ref}}$ and $\mathcal{X}_{\mu_k}^{\text{alt}}$ contain states whose shared components (W, E) are reachable by the references and alternation, respectively. For $\tau > U_*$, the target set $\mathcal{T} := \{x \in \mathcal{X}_{\mu_k}^{\text{alt}} \setminus \mathcal{X}_{\text{ref}} : q(x) \geq \tau\}$ contains high-quality states beyond reference reachability; the next assumption requires persistent cooperative access to them.

Assumption 3.2 (Cooperative access and use)**.** Fix μ_k and $\tau \in (U_*, 1]$. In cycle m, agents u and $\bar{u}$ start at x_{2m} and x_{2m+1}, respectively. There exist constants $\alpha, \beta \in (0, 1]$ such that, for every m with $\Pr_{\mu_k}(q(x_{2m}) < \tau) > 0$,

$$\Pr_{\mu_k}(x_{2m+1} \in G \mid q(x_{2m}) < \tau) \geq \alpha, \quad G := \{x : q(x) < \tau,\ \textstyle\sum_{x' \in \mathcal{T}} P_k^{(\bar{u})}(x, x') \geq \beta\}. \tag{3}$$

Here α bounds u's access to states useful to $\bar{u}$, and β bounds $\bar{u}$'s probability of reaching $\mathcal{T}$.

Theorem 3.3 (Joint alternation exceeds the reference limits)**.** *Under Assumptions 3.1 and 3.2, let $a = \alpha\beta$ and $F_0 = \Pr_{\mu_k}(q(x_0) < \tau)$. For finite k and integers $m \geq 0$,*

$$\text{Perf}_{\mu_k}(2km) \geq \tau[1 - F_0(1-a)^m], \qquad U_{\mu_k} := \lim_{N \to \infty} \text{Perf}_{\mu_k}(N) \geq \tau > U_*. \tag{4}$$

Proof sketch. With $F_m = \Pr_{\mu_k}(q(x_{2m}) < \tau)$, persistent access and quality preservation give $F_{m+1} \leq (1-a)F_m$. Iteration and $q \geq 0$ imply the bound; monotone convergence gives the limit. □

Corollary 3.4 (Experiment budget)**.** *Under Theorem 3.3, with $F_0 = 1$, fixed $p_0 \in (0, 1)$, and $0 < a \ll 1$, a sufficient budget for $p_{\mu_k}(N; \tau) \geq p_0$ has the leading form*

$$N_{\text{suff}}(k, p_0, \tau) = \frac{2k}{a} \log \frac{1}{1 - p_0} \big[1 + O(a)\big]. \tag{5}$$

Taking $p_0 = (U_* + \eta)/\tau$, $0 < \eta < \tau - U_*$ guarantees mean quality $U_* + \eta$. The budget form also applies to a single agent with a uniform conditional success probability of at least $a > 0$. Appendix A.2 gives the proofs, exact budgets, and startup analysis.

3.3 Fixed-k Interruption and Empirical Choice

In this section, we study when fixed-k interruption maintains the advantage in Theorem 3.3 and motivate short caps empirically. We use *Natural-Termination Alternation* (**NTA**), $\mu_\infty := (A, B, u_0, \infty)$, as the reference without fixed-k interruption, with $d_i^{(u)} = \min\{T_i^{(u)}(x_i), t\}$.

Let $x_{i,\ell}$ be the state after ℓ experiments within option i, with entry $x_{i,0} = x_i$ and final state $x_{i,-1} := x_{i,d_i^{(u_i)}}$ before context reset. At a boundary, $x_{i+1,0} = f(x_{i,-1})$, where $f(W, E, c) := (W, E, \emptyset)$. We define $A^{\mu_\infty}(x_{i,-1}) := V^{\mu_\infty}(x_{i+1,0}) - Q^{\mu_\infty}(x_{i,-1}, o_i^{(u_i)})$. Here Q^{μ_∞} continues the current option, whereas V^{μ_∞} starts the reset option; the interruption theorem (Sutton et al., 1999, Theorem 2) gives nondegradation when $A^{\mu_\infty}(x_{i,-1}) \geq 0$.

For the capped policy μ_k, define the interrupted option indices $\mathcal{J}_{\mu_k}(N) := \{i \geq 0 : \sum_{j=0}^{i} d_j^{(u_j)} < N,\ d_i^{(u_i)} = k < T_i^{(u_i)}(x_{i,0})\}$. The expected net gain and NTA margin are $\mathcal{A}_{\mu_k}(N) := \mathbb{E}_{\mu_k} \sum_{i \in \mathcal{J}_{\mu_k}(N)} A^{\mu_\infty}(x_{i,-1})$ and $\Delta_{\mu_\infty}(N) := \text{Perf}_{\mu_\infty}(N) - U_*$.

Theorem 3.5 (Expected advantage under fixed-k interruption)**.** *For the optimal initial agent u_0, a sufficient condition to exceed U_* is*

$$\Delta_{\mu_\infty}(N) + \mathcal{A}_{\mu_k}(N) > 0 \quad \Longrightarrow \quad \text{Perf}_{\mu_k}(N) > U_*. \tag{6}$$

Proof sketch. Telescoping the option-level Bellman recursion gives $\text{Perf}_{\mu_k}(N) - \text{Perf}_{\mu_\infty}(N) = \mathcal{A}_{\mu_k}(N)$. Substitution proves the claim; see Appendix A.3. □

Corollary 3.6 (Upper bound on the required cap)**.** *Suppose $\Delta_{\mu_\infty}(N) > 0$ and interruption losses decay geometrically with k, then some finite $1 \leq k \leq k_0$ satisfies $\text{Perf}_{\mu_k}(N) > U_*$.*

Empirically selected cap range: $k \leq 5$. Figure 3 compares six goal and five NTA configurations on $n = 75$ MLE-bench tasks (Chan et al., 2025). The prefix curves show that $k = 5$ preserves most observed gains with fewer within-option experiments, motivating candidate caps $k \leq 5$. Detailed analysis are presented in Appendix A.3; statistical and empirical details are provided in Appendix B.

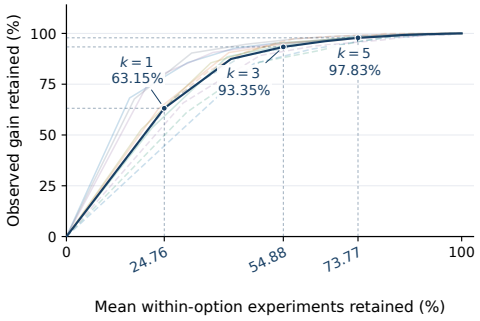

Figure 3: **Short prefixes retain most observed gains.** The horizontal axis reports the mean percentage of within-option experiments retained by the first k experiments, and the vertical axis reports the corresponding percentage of observed gains retained. Faint curves represent individual configurations, the dark curve averages gain retention equally across configurations, and guides mark $k = 1, 3, 5$. Across the eleven configurations, $k = 5$ retains 97.83% of observed gains with a mean within-option experiment retention of 73.77%, supporting candidate caps $k \leq 5$.

Algorithm 1 HMA with Fixed-k Interruption

Require: Workflow $\mu_k = (A, B, u_0, k)$; task distribution ν_0; total time t_{total}; final review time t_{rev}.
Ensure: Final candidate $\widehat{E}$.

Stage 1: Workflow Initialization
$x_0 = (W_0, E_0, \emptyset) \sim \nu_0$.
$t_{\text{end}} \leftarrow t_{\text{total}}$; $t_{\text{s}} \leftarrow t_{\text{end}} - t_{\text{rev}}$.
$i \leftarrow 0$; $u \leftarrow u_0$; $\mathcal{E} \leftarrow \emptyset$.
Stage 2: Alternate Options
while $t < t_{\text{s}}$ **do**
  $\ell \leftarrow 0$; $x_{i,0} \leftarrow x_i$.
  while $\beta_i^{(u)}(\ell, t; k) = 0$ **do**
    $x_{i,\ell+1} \leftarrow \pi_i^{(u)}(x_{i,\ell})$.
    Append $(E_{i,\ell+1}, u)$ to $\mathcal{E}$.
    $\ell \leftarrow \ell + 1$.
  end while
  $x_{i+1} \leftarrow f(x_{i,-1})$.
  $i \leftarrow i + 1$; $u \leftarrow \bar{u}$.
end while
Stage 3: Final Review
$\widehat{E} \leftarrow E_{-1}$.
if $|\mathcal{E}| \geq 2 \wedge t < t_{\text{end}}$ **then**
  $\widehat{E} \leftarrow \textsc{Review}^{(\bar{u}_{-1})}(\mathcal{E}, E_{-1}, t_{\text{end}})$.
end if
return $\widehat{E}$.

3.4 HMA Workflow Overview

In this section, we describe the implementation of the HMA workflow, including alternating agent execution and final review, as summarized in Algorithm 1.

Agent Execution. Algorithm 1 implements $\mu_k = (A, B, u_0, k)$ from $x_0 \sim \nu_0$. Here t is elapsed time, $t_{\text{end}} = t_{\text{total}}$ is the final deadline, and $t_{\text{s}} = t_{\text{end}} - t_{\text{rev}}$ ends exploration. Agent u executes $\pi_i^{(u)}$ until $\beta_i^{(u)}(\ell, t; k) = \mathbf{1}\{\ell \geq \min\{T_i^{(u)}(x_i), k\} \ \vee \ t \geq t_{\text{s}}\} = 1$. Then $x_{i,-1} = x_{i,\ell}$ and $x_{i+1} = f(x_{i,-1})$ reset context for agent $\bar{u}$.

Final review. The archive $\mathcal{E}$ stores evaluated candidates and their authors in order; (E_{-1}, u_{-1}) is the latest pair. We reserve the final t_{rev} for the other agent $\bar{u}_{-1}$ to review the archive and select $\widehat{E} \leftarrow \textsc{Review}^{(\bar{u}_{-1})}(\mathcal{E}, E_{-1}, t_{\text{end}})$, with E_{-1} as the fallback. This cross-agent check seeks agreement on the best archived candidate without test-set scores, approximating the result-retention objective of Assumption 3.1 at final selection.

4 Experiment

4.1 Experiment Setup

Implementation details. We implement HMA on Humanize (Liu et al., 2026), following Algorithm 1 with GPT, Kimi, GLM, DeepSeek, and Claude agents, each using their native harness and `max` reasoning effort. Each task run uses one NVIDIA A10 GPU and a six-hour budget: agents alternate under $k = 5$, reset context at each handoff, and reserve the final 15 minutes for final review. Exact model–harness mappings, hardware, baseline configurations, and framework details are given in Appendix C.

Benchmark. We evaluate on MLE-bench (Chan et al., 2025), comprising 75 Kaggle competitions requiring agents to develop machine learning solutions. We report the percentage of tasks earning

any medal (All) or a gold medal (Gold), supplemented by complexity-specific and leaderboard-based metrics in Table 2. Test-set scores are unavailable to agents and used only for offline evaluation.

4.2 Main Comparison

Table 2: **HMA exceeds the stronger constituent agent in overall medal rate.** All performance metrics are percentages. Lite, Medium, and High report medal rates by task complexity. Med+ measures the percentage outperforming the human leaderboard median; Mean Percentile averages leaderboard percentiles across tasks. *Avg. of Single* averages the two constituent six-hour goal baselines, and signed gains measure HMA's improvement over this average in percentage points. HMA starts with the first listed model. Bold and underlined medal rates mark the best and second-best results. Statistical details and available HMA standard deviations appear in Appendix C.3.

Workflow	Time (h)	Medal Rate (%)			Overall Metrics (%)			Mean Percentile (%)
		Lite (22)	Medium (38)	High (15)	All (75)	Gold	Med+	
Single-Agent Baselines with Native Harnesses (`/goal`)								
Claude Opus 5	6	80.3 ± 1.5	<u>78.1</u> ± 2.3	46.7 ± 0.0	72.4 ± 1.6	<u>50.7</u> ± 0.8	<u>85.3</u> ± 1.3	84.8 ± 0.6
GPT-5.6-sol	6	80.3 ± 1.5	69.3 ± 2.3	46.7 ± 6.7	68.0 ± 1.3	47.6 ± 2.5	81.3 ± 0.8	81.3 ± 1.3
GLM-5.3	6	77.3 ± 0.0	57.0 ± 0.9	40.0 ± 0.0	59.6 ± 0.4	38.2 ± 2.2	75.1 ± 1.2	75.7 ± 0.9
DeepSeek-V4.1-Flash	6	77.3 ± 2.6	61.4 ± 6.3	37.8 ± 2.2	61.3 ± 4.1	37.8 ± 0.9	74.2 ± 2.4	74.7 ± 2.3
Kimi K3	6	77.3 ± 2.6	45.6 ± 2.3	42.2 ± 2.2	54.2 ± 1.6	34.7 ± 1.3	73.3 ± 2.3	73.8 ± 1.6
DeepSeek-V4-Flash	6	54.5 ± 6.9	24.6 ± 1.8	24.4 ± 5.9	33.3 ± 3.5	15.6 ± 1.6	50.2 ± 3.2	51.8 ± 2.8
Open-Source Agent Baselines								
ScienceFlow DeepSeek-V4-Flash-Preview	24	80.3 ± 1.5	74.6 ± 0.9	44.4 ± 2.2	70.2 ± 1.2	–	–	–
MLEvolve Gemini-3.1-Pro-preview	12	80.3 ± 1.5	64.0 ± 0.9	46.7 ± 0.0	65.3 ± 0.8	34.7 ± 0.0	76.0 ± 2.3	–
ML-Master 2.0 DeepSeek-V3.2-Speciale	24	75.8 ± 1.5	50.9 ± 3.5	42.2 ± 2.2	56.4 ± 2.5	19.6 ± 0.9	63.1 ± 1.2	–
HMA *(Ours)*								
Claude Opus 5 ↔ GPT-5.6-sol								
Avg. of Single	6	80.3 ± 1.1	73.7 ± 1.6	46.7 ± 3.3	70.2 ± 1.0	49.1 ± 1.3	83.3 ± 0.8	83.0 ± 0.7
HMA (ours)	6	<u>84.8</u> +4.5	**80.7** +7.0	**62.2** +15.6	**78.2** +8.0	**53.8** +4.7	**90.7** +7.3	**89.0** +5.9
GPT-5.6-sol ↔ Claude Opus 5								
Avg. of Single	6	80.3 ± 1.1	73.7 ± 1.6	46.7 ± 3.3	70.2 ± 1.0	49.1 ± 1.3	83.3 ± 0.8	83.0 ± 0.7
HMA (ours)	6	**86.4** +6.1	76.3 +2.6	46.7 +0.0	<u>73.3</u> +3.1	<u>50.7</u> +1.6	<u>85.3</u> +2.0	<u>85.4</u> +2.3
GPT-5.6-sol ↔ DeepSeek-V4.1-Flash								
Avg. of Single	6	78.8 ± 1.5	65.4 ± 3.4	42.2 ± 3.5	64.7 ± 2.1	42.7 ± 1.3	77.8 ± 1.2	78.0 ± 1.3
HMA (ours)	6	81.8 +3.0	76.3 +10.9	46.7 +4.5	72.0 +7.3	<u>50.7</u> +8.0	<u>85.3</u> +7.6	84.4 +6.4
GPT-5.6-sol ↔ Kimi K3								
Avg. of Single	6	78.8 ± 1.5	57.5 ± 1.6	44.4 ± 3.5	61.1 ± 1.0	41.1 ± 1.4	77.3 ± 1.2	77.6 ± 1.0
HMA (ours)	6	81.8 +3.0	71.1 +13.6	<u>53.3</u> +8.9	70.7 +9.6	<u>50.7</u> +9.6	<u>85.3</u> +8.0	83.9 +6.4
GPT-5.6-sol ↔ GLM-5.3								
Avg. of Single	6	78.8 ± 0.8	63.2 ± 1.2	43.3 ± 3.3	63.8 ± 0.7	42.9 ± 1.7	78.2 ± 0.7	78.5 ± 0.8
HMA (ours)	6	77.3 −1.5	73.7 +10.5	46.7 +3.4	69.3 +5.6	<u>50.7</u> +7.8	82.7 +4.4	81.5 +2.9
DeepSeek-V4.1-Flash ↔ Kimi K3								
Avg. of Single	6	77.3 ± 1.9	53.5 ± 3.4	40.0 ± 1.6	57.8 ± 2.2	36.2 ± 0.8	73.8 ± 1.6	74.3 ± 1.4
HMA (ours)	6	77.3 +0.0	63.2 +9.6	40.0 +0.0	62.7 +4.9	38.7 +2.4	80.0 +6.2	78.1 +3.8

HMA outperforms single-agent and specialized workflow baselines. Table 2 compares HMA with single agents using native harnesses, evaluated in our experiments, and specialized workflows using results reported in their original papers (Zhu et al., 2026; Du et al., 2026; Zhao et al., 2026). Opus-first HMA achieves 78.2% overall medals, exceeding our strongest single-agent baseline at 72.4% and the highest published workflow baseline at 70.2%. HMA and our single-agent baselines share a six-hour budget, whereas the published workflows use 12–24 hours. HMA achieves these

gains without task-specific knowledge bases or additional prompt engineering, while retaining the agents' existing internal harnesses.

HMA improves heterogeneous agent pairs across capability levels. Across five model pairs with constituent medal rates of 54.2%–72.4%, all six HMA configurations improve overall medal rate over the stronger constituent, and Gold, Med+, and mean percentile over constituent averages. For Opus–GPT, both starting orders retain the advantage, reaching 78.2% and 73.3% medals versus the stronger constituent's 72.4%, indicating that the observed advantage is not tied to a single favorable initial agent.

4.3 Experiment Analysis and Ablation Study

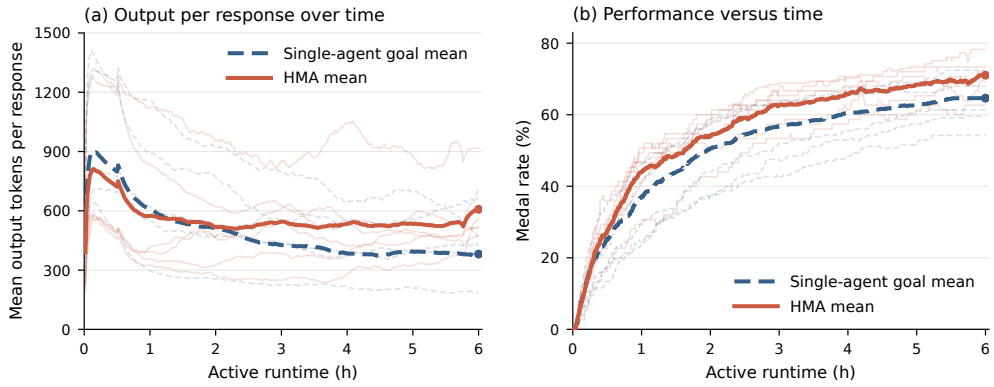

Figure 4: **HMA sustains response length and continued improvement.** (a) Mean output tokens per response and (b) medal rate, both plotted against active runtime. Bold red and dashed blue curves report weighted means for HMA and its matched constituent goal references, respectively; each HMA configuration receives equal weight, and each goal reference is the equally weighted mean of its two constituent agents. Faint curves show individual configurations.

HMA sustains exploration and continued improvement. The response-level curve in Figure 4 (a) uses the same 30-minute trailing window. From hours 0–2 to hours 3–5, HMA retains 86.4% of its mean output tokens per response, versus 60.7% for matched goal references. In Figure 4 (b), the six HMA configurations reach 42.7%–60.0% medals within two hours and continue improving through six hours, contrasting with the single-agent deceleration in Section 2.

Final review improves historical any-medal retention. Holding exploration fixed, final review changes 8.9% of selections and increases any-medal outcomes from 311 to 319, with no losses relative to the pre-review candidate. Among the 450 task runs, 330 produce at least one medal-winning exploration candidate; final selection retains a medal in 319 of these runs (96.7%), compared with 311 (94.2%) before final review. These results quantify how terminal review supports the best-result retention objective in Assumption 3.1.

Additional analyses and ablations. Appendix E details the weighting and aggregation used to construct the mean curves in Figure 4, together with trajectory construction and task-level comparisons. Appendix F further examines starting-order effects, model time allocation, and partner participation (Appendix F.1); the interruption-parameter ablations(Appendix F.2); natural termination and the fixed-duration approximation (Appendix F.3); and final review's effects on final selection and historical any-medal retention (Appendix F.4).

4.4 Case Study

On INGV Volcanic Eruption Prediction, GPT and Opus improve the historical best MAE in five of six post-handoff options (Figure 5). Their final MAE of 34,234 is 46.5% below the best of six constituent goal runs. Mean response length also increases on four of five returns to a previously used

model, illustrating continued refinement of shared task state alongside renewed output. Appendix G provides detailed measurements and additional cross-model cases.

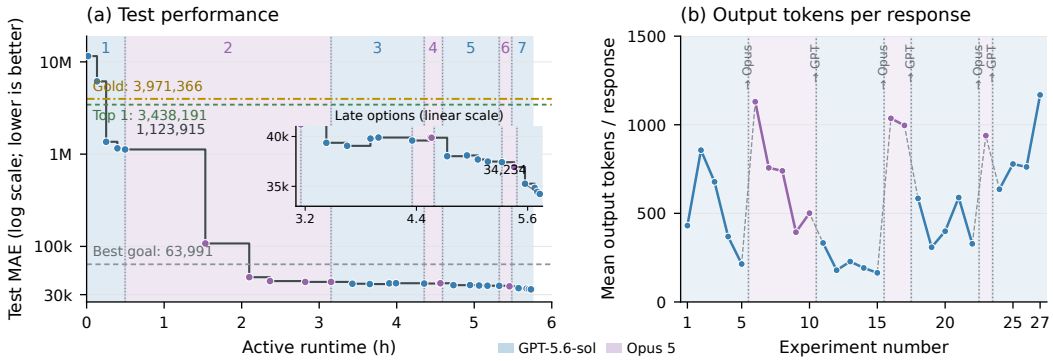

Figure 5: **HMA GPT–Opus alternation sustains improvement on INGV.** (a) Test MAE across 27 experiments, using a logarithmic axis with a linear inset. (b) Mean output tokens per response for each experiment. Blue and purple regions identify GPT and Opus; dotted lines mark handoffs, and horizontal lines show reference scores.

5 Related Work

MLE agents and harness-level systems. MLE-bench evaluates autonomous agents on Kaggle competitions (Chan et al., 2025). Recent MLE systems improve individual agents through search, reflection, memory, branch reuse, recoverable workspaces, or task-conditioned interfaces (Toledo et al., 2025; Chen et al., 2026; Du et al., 2026; Zhu et al., 2026; Zhao et al., 2026; Yu et al., 2026; Zhang et al., 2026; Ye et al., 2026). These methods modify how one model acts within its harness. HMA instead leaves each native model–harness pair unchanged and coordinates complete agents at the workflow level.

Multi-agent and long-horizon workflows. Prior workflows sustain long-horizon execution through reusable experience, reflection, tree search, workflow search, communication, role assignment, checkpointing, or independent review (Shinn et al., 2023; Zhao et al., 2024; Zhou et al., 2024; Zhang et al., 2025; Wu et al., 2024; Qian et al., 2024; Li et al., 2025). Recent handoff studies examine candidate transfer, learned control exchange, and the quality cost of continuing another model's trajectory (Luo et al., 2026; Li et al., 2026; Ganz et al., 2026). HMA studies fixed-cap alternation of complete native agents on one shared workspace, with a context reset at every handoff.

Options and semi-MDPs. Semi-Markov decision processes model actions with variable duration, while options represent such actions through an internal policy and termination rule (Bradtke & Duff, 1994; Dietterich, 2000). The interruption theorem characterizes when terminating an option early preserves or improves continuation value (Sutton et al., 1999); subsequent work studies learned and off-policy termination (Bacon et al., 2017; Harutyunyan et al., 2018). HMA models each agent session as an option and applies this framework to fixed-cap alternation with context reset.

6 Conclusion

HMA alternates native agents, resetting context while preserving shared task state. Our CR-SMDP analysis gives sufficient conditions and budget bounds for exceeding single-agent reference limits and conditions for preserving this advantage under fixed-cap interruption. Across six MLE-bench configurations, HMA outperforms the stronger constituent in medal rate under equal six-hour budgets, reaching 78.2% at best. These findings support workflow-level coordination as a route to sustained exploration without additional harness or context engineering.

AI USE STATEMENT

In this work, we used generative AI tools for language polishing, grammar checking, and generating portions of the code. The research ideas, methodology, experimental design, and interpretation of results were developed by the authors. We reviewed all AI-assisted text to ensure that it accurately conveyed our intended meaning. LLM-generated code was verified and tested for correctness by two authors. We take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the assistance of generative AI.

REPRODUCIBILITY STATEMENT

We provide an anonymous HMA code supplement containing the workflow implementation, native-agent runtime, benchmark integration, configuration templates, dependency specifications, and protocol tests. The anonymous supplement is available at `https://anonymous.4open.science/r/hma-CE5F/README.md`. Algorithm 1 and Section 3.4 specify the execution and final review mechanisms. Our modeling assumptions and complete proofs appear in Appendix A, while Appendix B details the statistical analysis of interruption caps. Section 4.1 and Appendix C document hardware, model–harness configurations, budgets, evaluation, and aggregation procedures. The supplementary README provides installation and validation instructions and identifies the separately required MLE-bench data, grader, native-agent CLIs, and provider access.

ETHICS STATEMENT

This study uses the public MLE-bench benchmark and does not involve human participants, human-subject interventions, or the collection of personally identifiable information. Hidden test labels and evaluator data are kept outside the agent environment and are used only for offline scoring; the agents receive no test-set feedback. The anonymous supplement contains no provider credentials, private task data, experimental transcripts, or unpublished result files. The workflow executes code in isolated task and evaluator environments, with explicit separation between public inputs and private evaluation data. We report the protocol, exclusions, and aggregation rules used for evaluation and do not claim that alternation is beneficial for every model pair or task. The authors have no competing interests to disclose.

REFERENCES

Anthropic. Prompting Claude Opus 5. Claude Platform documentation, 2026a. URL `https://platform.claude.com/docs/en/build-with-claude/prompt-engineering/prompting-claude-opus-5`. Accessed September 25, 2026.

Anthropic. How Claude Code works. Claude Code documentation, 2026b. URL `https://code.claude.com/docs/en/how-claude-code-works`. Documentation version updated September 26, 2026. Accessed September 26, 2026.

Pierre-Luc Bacon, Jean Harb, and Doina Precup. The Option-Critic architecture. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1), February 2017. ISSN 2159-5399. doi: 10.1609/aaai.v31i1.10916. URL `http://dx.doi.org/10.1609/aaai.v31i1.10916`.

Michael Bolin. Unrolling the Codex agent loop. OpenAI Engineering, January 2026. URL `https://openai.com/index/unrolling-the-codex-agent-loop/`. Published January 23, 2026. Accessed September 21, 2026.

Steven Bradtke and Michael Duff. Reinforcement learning methods for continuous-time Markov decision problems. In G. Tesauro, D. Touretzky, and T. Leen (eds.), *Advances in Neural Information Processing Systems*, volume 7. MIT Press, 1994. URL `https://proceedings.neurips.cc/paper_files/paper/1994/file/07871915a8107172b3b5dc15a6574ad3-Paper.pdf`.

Jun Shern Chan, Neil Chowdhury, Oliver Jaffe, James Aung, Dane Sherburn, Evan Mays, Giulio Starace, Kevin Liu, Leon Maksin, Tejal Patwardhan, Lilian Weng, and Aleksander Mądry.

MLE-bench: Evaluating machine learning agents on machine learning engineering, 2025. URL https://arxiv.org/abs/2410.07095.

Jiefeng Chen, Bhavana Dalvi Mishra, Jaehyun Nam, Rui Meng, Tomas Pfister, and Jinsung Yoon. MARS: Modular agent with reflective search for automated AI research, 2026. URL https://arxiv.org/abs/2602.02660.

DeepSeek-AI. DeepSeek Harness: Everything is a plugin. https://github.com/deepseek-ai/deepseek-harness, 2026.

T. G. Dietterich. Hierarchical reinforcement learning with the MAXQ value function decomposition. *Journal of Artificial Intelligence Research*, 13:227–303, November 2000. ISSN 1076-9757. doi: 10.1613/jair.639. URL http://dx.doi.org/10.1613/jair.639.

Shangheng Du, Xiangchao Yan, Jinxin Shi, Zongsheng Cao, Shiyang Feng, Zichen Liang, Boyuan Sun, Tianshuo Peng, Yifan Zhou, Xin Li, Jie Zhou, Liang He, Bo Zhang, and Lei Bai. MLEvolve: A self-evolving framework for automated machine learning algorithm discovery, 2026. URL https://arxiv.org/abs/2606.06473.

Roy Ganz, Mor Shpigel Nacson, Adi Kalyanpur, and Ron Litman. The handoff tax: Continuing non-native trajectories in LLM agents, 2026. URL https://arxiv.org/abs/2608.24358.

Aurélien Garivier and Olivier Cappé. The KL-UCB algorithm for bounded stochastic bandits and beyond. In Sham M. Kakade and Ulrike von Luxburg (eds.), *Proceedings of the 24th Annual Conference on Learning Theory*, volume 19 of *Proceedings of Machine Learning Research*, pp. 359–376, Budapest, Hungary, 09–11 Jun 2011. PMLR. URL https://proceedings.mlr.press/v19/garivier11a.html.

Anna Harutyunyan, Peter Vrancx, Pierre-Luc Bacon, Doina Precup, and Ann Nowé. Learning with options that terminate off-policy. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), April 2018. ISSN 2159-5399. doi: 10.1609/aaai.v32i1.11740. URL http://dx.doi.org/10.1609/aaai.v32i1.11740.

Andrej Karpathy. autoresearch: AI agents running research on single-GPU nanochat training automatically. GitHub, 2026. URL https://github.com/karpathy/autoresearch. Released March 2026. Accessed September 25, 2026.

Jiayu Li, Enpei Zhang, Dawei Zhou, Elynn Chen, and Yujun Yan. Learning to hand off: Provably convergent workflow learning under interface constraints, 2026. URL https://arxiv.org/abs/2605.19140.

Yang Li, Siqi Ping, Xiyu Chen, Xiaojian Qi, Zigan Wang, Ye Luo, and Xiaowei Zhang. AgentGit: A version control framework for reliable and scalable LLM-powered multi-agent systems, 2025. URL https://arxiv.org/abs/2511.00628.

Sihao Liu, Ligeng Zhu, Zijian Zhang, Changye Li, Song Han, and Tony Nowatzki. Humanize: Orchestrate, execute, and observe agent flows. https://github.com/humanfia/humanize, 2026. GitHub repository.

Sichun Luo, Yi Huang, Guanzhi Deng, Haibo Wang, Haochen Luo, Lei Li, Zefa Hu, Junlan Feng, and Qi Liu. Relay, don't route: Adaptive population handoff for cost-efficient LLM-driven evolution, 2026. URL https://arxiv.org/abs/2608.05651.

Moonshot AI. Kimi Code CLI. GitHub, 2026. URL https://github.com/MoonshotAI/kimi-code. Official open-source software. Accessed September 21, 2026.

Eric Provencher. Rethinking skills and prompts for GPT-6 Astra. OpenAI Developers Blog, September 2026. URL https://developers.openai.com/blog/rethinking-skills-and-prompts-for-gpt-6-astra. Accessed September 25, 2026.

Chen Qian, Wei Liu, Hongzhang Liu, Nuo Chen, Yufan Dang, Jiahao Li, Cheng Yang, Weize Chen, Yusheng Su, Xin Cong, Juyuan Xu, Dahai Li, Zhiyuan Liu, and Maosong Sun. ChatDev: Communicative agents for software development. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15174–15186, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.810. URL https://aclanthology.org/2024.acl-long.810/.

Prithvi Rajasekaran, Ethan Dixon, Carly Ryan, and Jeremy Hadfield. Effective context engineering for AI agents. Anthropic Engineering, September 2025. URL https://www.anthropic.com/engineering/effective-context-engineering-for-ai-agents. Published September 29, 2025. Accessed September 21, 2026.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: language agents with verbal reinforcement learning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 8634–8652. Curran Associates, Inc., 2023. doi: 10.52202/075280-0377. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/1b44b878bb782e6954cd888628510e90-Paper-Conference.pdf.

Richard S. Sutton, Doina Precup, and Satinder Singh. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1-2):181–211, August 1999. ISSN 0004-3702. doi: 10.1016/s0004-3702(99)00052-1. URL http://dx.doi.org/10.1016/S0004-3702(99)00052-1.

Edan Toledo, Karen Hambardzumyan, Martin Josifoski, Rishi Hazra, Nicolas Baldwin, Alexis Audran-Reiss, Michael Kuchnik, Despoina Magka, Minqi Jiang, Alisia Maria Lupidi, Andrei Lupu, Roberta Raileanu, Kelvin Niu, Tatiana Shavrina, Jean-Christophe Gagnon-Audet, Michael Shvartsman, Shagun Sodhani, Alexander H. Miller, Abhishek Charnalia, Derek Dunfield, Carole-Jean Wu, Pontus Stenetorp, Nicola Cancedda, Jakob Nicolaus Foerster, and Yoram Bachrach. AI research agents for machine learning: Search, exploration, and generalization in MLE-bench, 2025. URL https://arxiv.org/abs/2507.02554.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W White, Doug Burger, and Chi Wang. AutoGen: Enabling next-gen LLM applications via multi-agent conversations. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=BAakY1hNKS.

Zihao Ye, Yingyi Huang, Hongyi Jin, Bohan Hou, Junru Shao, Zhongming Yu, Jinqi Chen, Meghan Cowan, Shiyi Cao, Shanli Xing, Hanfeng Chen, Vinod Grover, Tianqi Chen, and Luis Ceze. CAKE: Compiler-agent co-design for frontier kernel evolution, 2026. URL https://arxiv.org/abs/2608.12629.

Justin Young. Effective harnesses for long-running agents. Anthropic Engineering, November 2025. URL https://www.anthropic.com/engineering/effective-harnesses-for-long-running-agents. Published November 26, 2025. Accessed September 21, 2026.

Zhaochen Yu, Yingcheng Wu, Zhenfei Yin, Kaiyuan Chen, Zhe Zhao, Mengdi Wang, Shuicheng Yan, and Ling Yang. Recursive experiential-working memory evolution for long-horizon agent harnesses, 2026. URL https://arxiv.org/abs/2608.24876.

Guibin Zhang, Leo Lu, Fangzhou Xie, Kang Zhu, Junhao Wang, Zhifei Xie, Zhaochen Yu, Zihang Liu, Zhongxiang Sun, Qiankun Li, Yue Liao, Heng Chang, Xiaobin Hu, Qibing Ren, Wangchunshu Zhou, Chuanrui Hu, Yafeng Deng, and Shuicheng Yan. JIT-Agent: Scaling harness intelligence via just-in-time harness evolution, 2026. URL https://arxiv.org/abs/2608.25593.

Jiayi Zhang, Jinyu Xiang, Zhaoyang Yu, Fengwei Teng, Xiong-Hui Chen, Jiaqi Chen, Mingchen Zhuge, Xin Cheng, Sirui Hong, Jinlin Wang, Bingnan Zheng, Bang Liu, Yuyu Luo, and Chenglin

Wu. AFlow: Automating agentic workflow generation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=z5uVAKwmjf.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. ExpeL: LLM agents are experiential learners. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17):19632–19642, March 2024. ISSN 2159-5399. doi: 10.1609/aaai.v38i17.29936. URL http://dx.doi.org/10.1609/aaai.v38i17.29936.

Mingming Zhao, Jiqian Dong, Kangping Xu, Zadid Hasan, Chengrui Fan, Shan Jiang, Shuai Mao, Yating Ling, Linyi Zou, Tailin Zhou, Yun Hin Chan, Wenkai Zhang, Zhanhong Zhou, Guowei Huang, Hongliang Li, Wenjing Cun, Zhitang Chen, Mingxuan Yuan, and Yanhui Geng. ScienceFlow: A long-horizon agent for ML research, scientific discovery and beyond, 2026. URL https://arxiv.org/abs/2608.14354.

Andy Zhou, Kai Yan, Michal Shlapentokh-Rothman, Haohan Wang, and Yu-Xiong Wang. Language agent tree search unifies reasoning, acting, and planning in language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 62138–62160. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/zhou24r.html.

Xinyu Zhu, Yuzhu Cai, Zexi Liu, Bingyang Zheng, Cheng Wang, Rui Ye, Yuzhi Zhang, Linfeng Zhang, Weinan E, Siheng Chen, and Yanfeng Wang. Toward ultra-long-horizon agentic science: Cognitive accumulation for machine learning engineering, 2026. URL https://arxiv.org/abs/2601.10402.

APPENDIX CONTENTS

A Theoretical Details and Proofs **15**

A.1 Model Details and Exact Performance 15

A.2 Alternation Guarantee and Experiment Budget 15

A.3 Interruption Guarantee and Sufficient Caps 17

B Prefix Retention and Empirical Cap Selection **19**

B.1 Retention Statistics and Confidence Bounds 19

B.2 Configurations and Observed Prefix Results 19

C Experimental Setup and Reproducibility **21**

C.1 Humanize and Execution Environment 21

C.2 Agent and Baseline Configurations 21

C.3 Evaluation Protocol and Repeats 21

D Additional Evidence for the Motivation **23**

D.1 Declining Response Length in Single-Agent Runs 23

D.2 Harness Staleness across Model Versions 23

E Additional Main-Comparison Analyses **25**

F Ablation Studies **26**

F.1 Starting-Order Effects 26

F.2 Effects of the Fixed Interruption Cap 27

F.3 Natural Termination before the Cap 28

F.4 Final Review and Historical Any-Medal Retention 28

G Case Studies of Heterogeneous Alternation **30**

G.1 INGV Volcanic Eruption Prediction 30

G.2 Google QUEST Q&A Labeling: GPT–DeepSeek 30

G.3 PetFinder Pawpularity: GLM–Kimi 31

G.4 Google Smartphone Decimeter: DeepSeek–Kimi 32

A THEORETICAL DETAILS AND PROOFS

We organize the formal supplement in the same order as the Method section: Appendix A.1 defines the finite-budget CR-SMDP and exact performance curves; Appendix A.2 proves the cooperative alternation guarantee and budget bound; Appendix A.3 handles NTA, fixed-k interruption, and sufficient-cap conditions.

A.1 MODEL DETAILS AND EXACT PERFORMANCE

We supplement Section 3.1 with finite-budget definitions, the general option recursion, and exact performance expressions used in the alternation proof.

Definition A.1 (State and finite budget)**.** The state from Section 3.1 implicitly includes agent identity, option age, and any scheduling history needed for the Markov property (Dietterich, 2000, Section 3.2). For session $o_i^{(u_i)}$ at x_i, let n_i be the completed experiment count and t_i the remaining budget. Then

$$n_i := \sum_{h=0}^{i-1} d_h, \qquad t_i := N - n_i, \qquad d_i = \min\{T_i^{(u_i)}(x_i), k, t_i\}. \tag{7}$$

Here $n_0 = 0$, and $T_i^{(u)}(x_i)$ is the positive, almost-surely finite natural duration, which may depend on the entry state. Thus early natural termination and a final partial option are included; $k = \infty$ gives NTA. Values track the remaining budget, vanish when it is exhausted, and no option starts at $t_i = 0$.

Definition A.2 (Option rewards and transitions)**.** For $\gamma \in (0, 1]$ and the experiment rewards in Section 3.1, define

$$\bar{r}^{(u)}(x) := \mathbb{E}\left[\sum_{\ell=0}^{d_i-1} \gamma^\ell r_{n_i+\ell+1} \,\middle|\, x_i = x,\ u_i = u\right]. \tag{8}$$

Let $\Phi_{k,t}^{(u)}(x', d \mid x)$ be the joint law of duration and next option-entry state, including context reset, and $P_{k,t}^{(u)}(x, x') := \sum_d \Phi_{k,t}^{(u)}(x', d \mid x)$. The complete-option kernel is $P_k^{(u)} := P_{k,\infty}^{(u)}$; finite t truncates the same option. Suppressing fixed k, t, the general semi-MDP recursion (Sutton et al., 1999; Bradtke & Duff, 1994) is

$$Q^\mu(x, o_i^{(u)}) = \bar{r}^{(u)}(x) + \sum_{d,x'} \gamma^d \Phi^{(u)}(x', d \mid x) V^\mu(x'). \tag{9}$$

At $\gamma = 1$, this reduces to equation 1, and total reward is $q(E_N) - q(E_0)$.

Proposition A.3 (Exact performance expressions)**.** *Let $\mathcal{P}_\mu$ be the one-experiment kernel under μ on the augmented state, including option boundaries, and $S_\tau := \{x : q(x) \geq \tau\}$. After workflow termination, retain the terminal state. Then*

$$p_\mu(N; \tau) = \nu_0 \mathcal{P}_\mu^N \mathbf{1}_{S_\tau}, \qquad \mathrm{Perf}_\mu(N) = \nu_0 \mathcal{P}_\mu^N q, \tag{10}$$

where $\mathbf{1}_{S_\tau}$ indicates target attainment. These expressions give the probability of reaching S_τ and the expected final quality, respectively, with $\mathrm{Perf}_\mu(N) \geq \tau p_\mu(N; \tau)$.

Proof of Proposition A.3. The endpoint law is $\nu_0 \mathcal{P}_\mu^N$; applying it to $\mathbf{1}_{S_\tau}$ and q gives equation 10. Taking expectations of $q \geq \tau \mathbf{1}_{S_\tau}$ yields the inequality. □

A.2 ALTERNATION GUARANTEE AND EXPERIMENT BUDGET

We follow Section 3.2: cooperative access yields a failure recursion, which proves the alternation guarantee and the experiment budget; we then discuss startup effects.

Definition A.4 (Reference ceiling and cycle budget)**.** For the reference family in Section 3.2, bounded quality and Assumption 3.1 give

$$U_* := \max_{b \in \mathfrak{B}_{A,B}} U_b, \qquad U_b := \lim_{N\to\infty} \mathrm{Perf}_b(N) = \mathbb{E}_{b,\nu_0}\Big[\lim_{N\to\infty} q(E_N)\Big]. \tag{11}$$

Fix μ_k and $\tau > U_*$. Write u for the first agent in each cycle: cycle m executes $o_{2m}^{(u)}$ followed by $o_{2m+1}^{(\bar{u})}$, with kernel M_{μ_k}. Before total-budget truncation, the experiments used by m cycles are $C_{\mu_k,m} := \sum_{i=0}^{2m-1} \min\{T_i^{(u_i)}(x_i), k\}$, so $C_{\mu_k,m} \le 2km$ for finite k. Write $F_m := \Pr_{\mu_k}(q(x_{2m}) < \tau)$, where $\Pr_{\mu_k}$ and $\mathbb{E}_{\mu_k}$ include the initial draw from ν_0.

Lemma A.5 (Cooperative success and failure recursion)**.** *Under Assumptions 3.1 and 3.2, each cycle has conditional success probability at least $a := \alpha\beta$ whenever $F_m > 0$, and $F_{m+1} \le (1-a)F_m$ for all $m \ge 0$.*

Proof of Lemma A.5. The u-option enters G with conditional probability at least α, and the $\bar{u}$-option reaches $\mathcal{T} \subseteq S_\tau$ from each $x \in G$ with probability at least β. By the Markov property, for $F_m > 0$,

$$\Pr_{\mu_k}(x_{2m+2} \in S_\tau \mid x_{2m} \notin S_\tau) \ge \beta \Pr_{\mu_k}(x_{2m+1} \in G \mid x_{2m} \notin S_\tau) \ge \alpha\beta = a. \tag{12}$$

Quality retention makes S_τ absorbing, so $F_{m+1} \le F_m(1-a)$. If $F_m = 0$, it also gives $F_{m+1} = 0$. □

Theorem A.6 (Joint alternation exceeds the reference limits; Theorem 3.3 restated)**.** *Under Assumptions 3.1 and 3.2, take $a = \alpha\beta$ as a lower bound on conditional success per cycle, and $F_0 = \Pr_{\mu_k}(q(x_0) < \tau)$. For finite k,*

$$\mathrm{Perf}_{\mu_k}(2km) \ge \tau[1 - F_0(1-a)^m], \qquad U_{\mu_k} := \lim_{N\to\infty} \mathrm{Perf}_{\mu_k}(N) \ge \tau > U_*. \tag{13}$$

Proof of Theorem A.6. Iterating Lemma A.5 gives $F_m \le F_0(1-a)^m$. Since $C_{\mu_k,m} \le 2km$, quality retention and Proposition A.3 yield

$$\mathrm{Perf}_{\mu_k}(2km) \ge \tau p_{\mu_k}(2km;\tau) \ge \tau(1-F_m) \ge \tau[1 - F_0(1-a)^m]. \tag{14}$$

Bounded monotone convergence ensures that U_{μ_k} exists; letting $m \to \infty$ gives $U_{\mu_k} \ge \tau > U_*$. The same proof permits early natural termination when Assumption 3.2 holds for the resulting kernels, since $C_{\mu_k,m} \le 2km$ still holds. □

Corollary A.7 (Experiment budget; Corollary 3.4 restated)**.** *Under the theorem's conditions, let $F_0 = 1$ and fix $p_0 \in (0,1)$. For $0 < a \ll 1$, a sufficient budget for $p_{\mu_k}(N;\tau) \ge p_0$ has the leading form*

$$N_{\mathrm{suff}}(k, p_0, \tau) = \frac{2k}{a} \log \frac{1}{1-p_0} \big[1 + O(a)\big]. \tag{15}$$

Proof of Corollary A.7. For $0 < a < 1$, the bound $p_{\mu_k}(2km;\tau) \ge 1 - (1-a)^m$ reaches p_0 once $m \ge \log(1/(1-p_0))/[-\log(1-a)]$. Hence the exact rounded sufficient budget is

$$N_{\mathrm{suff}}(k, p_0, \tau) = 2k \left\lceil \frac{\log(1/(1-p_0))}{-\log(1-a)} \right\rceil. \tag{16}$$

Using $-\log(1-a) = a + O(a^2)$ and a rounding error below $2k$ gives $(2k/a)\log(1/(1-p_0)) + O(k)$, equivalent to equation 15 for fixed p_0. Finally, $p_0 = (U_* + \eta)/\tau$, $0 < \eta < \tau - U_*$, gives $\mathrm{Perf}_{\mu_k}(N) \ge U_* + \eta$ by Proposition A.3. □

Startup and initial-agent effects. Startup denotes the early cycles before cooperative progress stabilizes. The initial agent can affect access to G and finite-budget quality. If the two starts later have similar conditional success rates, their subsequent budget scales $2k/a$ are similar, while their startup progress can still differ.

A.3 Interruption Guarantee and Sufficient Caps

We expand Section 3.3. Corollary A.8 extends the alternation guarantee to NTA. We define interruption advantages in Definition A.9 and compare the workflows through option-level Bellman recursions. Theorem A.10 compares fixed-k interruption with NTA; Corollaries A.11–A.12 give a finite candidate condition and the explicit cap bound underlying Corollary 3.6.

Corollary A.8 (Natural termination and quality limits)**.** *For Natural-Termination Alternation (***NTA***), use $\mu_\infty = (A, B, u_0, \infty)$. Under Assumptions 3.1 and 3.2 and almost-surely finite option durations, let $M_\tau := \inf\{m \geq 0 : q(x_{2m}) \geq \tau\}$. Then $C_{\mu_\infty, M_\tau} < \infty$ almost surely and $U_{\mu_\infty} \geq \tau > U_*$. If* NTA *and a finite cap satisfy access for every $\tau \in (U_*, U^\dagger)$, where $U^\dagger \in (U_*, 1]$ bounds both reachable quality spaces, then $U_{\mu_k} = U_{\mu_\infty} = U^\dagger$.*

Proof of Corollary A.8. Applying equation 12 to NTA *gives $\Pr_{\mu_\infty}(M_\tau > m) \leq F_0(1-a)^m \to 0$, with $a = \alpha\beta$. Thus M_τ and the sum of its option durations are almost surely finite. Monotone quality yields $U_{\mu_\infty} \geq \tau$. Combining this argument with Theorem A.6 for every $\tau \in (U_*, U^\dagger)$ and using the common upper bound gives equality for both limits.* □

Definition A.9 (Interruption advantage and reference margin)**.** Fix $\mu_k = (A, B, u_0, k)$ and N. The workflows μ_k and $\mu_\infty = (A, B, u_0, \infty)$ share internal policies and boundary updates and differ only in option termination. Following Section 3.3, $x_{i,0} = x_i$ is option i's entry, $x_{i,-1} = x_{i, d_i^{(u_i)}}$ its endpoint before reset, and $x_{i+1,0} = f(x_{i,-1})$ the next entry, with $q(x_{i+1,0}) = q(x_{i,-1})$.

At an added cap interruption, NTA would continue from $x_{i,-1}$ with the current context. Ending the option instead gives advantage $A^{\mu_\infty}(x_{i,-1}) := V^{\mu_\infty}(x_{i+1,0}) - Q^{\mu_\infty}(x_{i,-1}, o_i^{(u_i)})$. Both values use NTA with the same remaining experiment budget; the next entry activates the other agent with empty context. The interruption contribution is zero at natural termination or budget exhaustion.

The added interruptions are indexed by

$$\mathcal{J}_{\mu_k}(N) := \left\{ i \geq 0 : \sum_{j=0}^{i} d_j^{(u_j)} < N,\ d_i^{(u_i)} = k < T_i^{(u_i)}(x_{i,0}) \right\}.$$

Thus natural termination takes precedence when it coincides with the cap. The expected net gain is $\mathcal{A}_{\mu_k}(N) := \mathbb{E}_{\mu_k} \sum_{i \in \mathcal{J}_{\mu_k}(N)} A^{\mu_\infty}(x_{i,-1})$, and the NTA margin is $\Delta_{\mu_\infty}(N) := \mathrm{Perf}_{\mu_\infty}(N) - U_*$.

Theorem A.10 (Expected advantage under fixed-k interruption; Theorem 3.5 restated)**.** *For the selected initial agent u_0, a sufficient condition to exceed U_* is*

$$\Delta_{\mu_\infty}(N) + \mathcal{A}_{\mu_k}(N) > 0 \quad \Longrightarrow \quad \mathrm{Perf}_{\mu_k}(N) > U_*. \tag{17}$$

Proof of Theorem A.10. Let M be the number of options executed under μ_k within budget N, with $x_{M,0}$ terminal, and let $\iota_i := \mathbf{1}_{\{i \in \mathcal{J}_{\mu_k}(N)\}}$. For each $i < M$, the option-level Bellman recursion (Sutton et al., 1999, proof of Theorem 2) gives

$$\begin{aligned} &\mathbb{E}_{\mu_k}[q(x_{i+1,0}) - q(x_{i,0}) + V^{\mu_\infty}(x_{i+1,0}) \mid x_{i,0}] - V^{\mu_\infty}(x_{i,0}) \\ &\quad = \mathbb{E}_{\mu_k}[\iota_i A^{\mu_\infty}(x_{i,-1}) \mid x_{i,0}]\,. \end{aligned} \tag{18}$$

Since $V^{\mu_\infty}(x_{M,0}) = 0$ and $x_{0,0} = x_0 \sim \nu_0$, summing telescopes:

$$\begin{aligned} \mathrm{Perf}_{\mu_k}(N) - \mathrm{Perf}_{\mu_\infty}(N) &= \mathbb{E}_{\mu_k}[q(x_{M,0}) - q(x_{0,0}) - V^{\mu_\infty}(x_{0,0})] \\ &= \mathbb{E}_{\mu_k} \sum_{i \in \mathcal{J}_{\mu_k}(N)} A^{\mu_\infty}(x_{i,-1}) = \mathcal{A}_{\mu_k}(N). \end{aligned} \tag{19}$$

Thus $\mathrm{Perf}_{\mu_k}(N) - U_* = \Delta_{\mu_\infty}(N) + \mathcal{A}_{\mu_k}(N)$, which proves equation 17. □

Initial-agent dependence of interruption gains. The kernels $P_k^{(A)}P_k^{(B)}$ and $P_k^{(B)}P_k^{(A)}$ can induce different early task states, changing both $\Delta_{\mu_\infty}(N)$ and $\mathcal{A}_{\mu_k}(N)$. Consequently, the preferred initial agent can depend on k and N; the associated startup budget effects are discussed in Appendix A.2.

Corollary A.11 (A sufficient cap in a finite candidate range)**.** *Fix A, B, u_0, N with $\Delta_{\mu_\infty}(N) > 0$ and integer $K \geq 1$. For each candidate cap $1 \leq k \leq K$, let $\mu_k = (A, B, u_0, k)$ and suppose there is $\varepsilon_{\mu_k} \geq 0$ with $\mathcal{A}_{\mu_k}(N) \geq -\varepsilon_{\mu_k}\mathbb{E}_{\mu_k}|\mathcal{J}_{\mu_k}(N)|$. Then*

$$\min_{1\leq k\leq K} \lfloor N/k \rfloor \varepsilon_{\mu_k} < \Delta_{\mu_\infty}(N) \quad \Longrightarrow \quad \exists\, 1 \leq k \leq K : \mathrm{Perf}_{\mu_k}(N) > U_*. \tag{20}$$

Proof of Corollary A.11. Each added interruption uses a distinct block of k experiments, so $|\mathcal{J}_{\mu_k}(N)| \leq \lfloor N/k \rfloor$ and $\mathcal{A}_{\mu_k}(N) \geq -\lfloor N/k \rfloor \varepsilon_{\mu_k}$. Choose a minimizing cap in equation 20 and apply Theorem A.10. □

Corollary A.12 (Explicit cap under geometric loss decay; quantitative form of Corollary 3.6)**.** *Fix A, B, u_0, N with $\Delta_{\mu_\infty}(N) > 0$. Under the expected loss condition of Corollary A.11, assume $\varepsilon_{\mu_k} \leq C\rho^k$ for every $\mu_k = (A, B, u_0, k)$, $k \geq 1$, with constants $C > 0$ and $0 < \rho < 1$. The policy $\mu_{k_0} := (A, B, u_0, k_0)$ with the following cap preserves at least half the* NTA *margin:*

$$\begin{aligned} k_0 &:= \max\left\{1, \left\lceil \frac{\log(2NC/\Delta_{\mu_\infty}(N))}{-\log\rho} \right\rceil \right\}, \\ \mathrm{Perf}_{\mu_{k_0}}(N) &\geq U_* + \tfrac{1}{2}\Delta_{\mu_\infty}(N) > U_*. \end{aligned} \tag{21}$$

Proof of Corollary A.12. The expected loss is at most $\lfloor N/k \rfloor C\rho^k \leq NC\rho^k$. The choice of k_0 makes this at most $\Delta_{\mu_\infty}(N)/2$; substitution in equation 19 proves the claim. Taking $k = k_0$ for the selected initial agent gives a sufficient cap in $[1, k_0]$, proving Corollary 3.6. Uniform C, ρ and a positive margin lower bound across budgets give $k_0 = O(\log N)$. □

B Prefix Retention and Empirical Cap Selection

We quantify the observed gains retained by short prefixes and provide the simultaneous confidence construction used to motivate candidate caps in Section 3.3. Executed fixed-k outcomes are analyzed separately in Appendix F.

B.1 Retention Statistics and Confidence Bounds

We establish the statistical guarantee supporting Section 3.3. Definition B.1 specifies task-level prefix retention, Assumption B.2 states the sampling condition, and Proposition B.3 gives simultaneous $1-\delta$ lower confidence bounds.

Definition B.1 (Observed prefix gain)**.** For policy μ and task j, define

$$\mathcal{O}_{\mu,j} := \{i : o_i^{(u_i)} \text{ is recorded on task } j \text{ under } \mu\}.$$

For each session index $i \in \mathcal{O}_{\mu,j}$, let l_i be its recorded experiment count and $q_i(d)$ its best quality after d experiments, with entry quality at $d=0$. For integer $k \geq 0$, define

$$g_i(k) = q_i(\min\{k, l_i\}) - q_i(0), \qquad \zeta_{\mu,j}(k) = \frac{\sum_{i\in\mathcal{O}_{\mu,j}} g_i(k)}{\sum_{i\in\mathcal{O}_{\mu,j}} g_i(l_i)}. \tag{22}$$

The ratio is defined for positive total gain; n_μ counts these tasks.

Assumption B.2 (Independent task observations)**.** For each prespecified policy μ, cap k, and statistic, let $Y_j \in [0,1]$ be $\zeta_{\mu,j}(k)$ or $\mathbf{1}_{\{\zeta_{\mu,j}(k)\geq 0.9\}}$. Conditional on defined ratios, assume independence across the n_μ tasks. Dependence across configurations and caps is allowed.

Proposition B.3 (Simultaneous lower confidence bounds)**.** *Under Assumption B.2, set* $\bar{Y} = n_\mu^{-1}\sum_j Y_j$ *and* $\mu_Y = n_\mu^{-1}\sum_j \mathbb{E}Y_j$*. For* M_{cmp} *prespecified comparisons and* $\delta \in (0,1)$*, define* $\underline{\mu}_Y \leq \bar{Y}$ *by*

$$n_\mu d_{\mathrm{KL}}(\bar{Y} \| \underline{\mu}_Y) = \log(M_{\mathrm{cmp}}/\delta), \tag{23}$$

with $\underline{\mu}_Y = 0$ *when* $\bar{Y} = 0$*. Then* $\Pr(\underline{\mu}_Y \leq \mu_Y \text{ for all comparisons}) \geq 1-\delta$*.*

Proof of Proposition B.3. The exponential-moment bound (Garivier & Cappé, 2011, Lemma 9), Jensen's inequality, and Chernoff optimization give $\Pr(\bar{Y} \geq z) \leq e^{-n_\mu d_{\mathrm{KL}}(z\|\mu_Y)}$ *for* $\mu_Y < z \leq 1$*. Inverting at failure level* δ/M_{cmp} *gives equation 23; a union bound over the comparisons proves the claim.* □

B.2 Configurations and Observed Prefix Results

This subsection documents the MLE-bench configurations and results supporting empirical cap selection in Section 3.3 and Figure 3.

Table 3 lists six goal and five NTA configurations, all using `max` reasoning effort. A goal task run is one continuous session; NTA alternates at natural termination, starting with the first listed model. Harness documentation is provided by Bolin (2026); Anthropic (2026b); Moonshot AI (2026); DeepSeek-AI (2026).

For MLE-bench, we use $q = 1 - (\text{position} - 1)/(\text{leaderboard size})$ on 75 tasks, including 75 measured Opus goal runs. Each goal task run is one option; NTA follows recorded boundaries, giving 1,065 options.

At each k, Figure 3 plots the mean of $\min\{k, l_i\}/l_i$ across options horizontally and gain retention averaged across tasks within each configuration, then equally across configurations vertically, both as percentages. Individual curves use each configuration's data, starting at $(0,0)$ and ending at $(100,100)$ at its maximum option length.

Table 3: Model and harness composition of the eleven MLE-bench prefix-analysis configurations. NTA arrows specify the starting order of the repeating pair.

Family	**Model(s), in starting order**	**Harness(es), same order**
Goal	GPT-5.6-sol	Codex
Goal	Kimi K3	Kimi Code
Goal	GLM-5.3	Claude Code
Goal	DeepSeek-V4-Flash	DeepSeek Harness
Goal	DeepSeek-V4.1-Flash	DeepSeek Harness
Goal	Claude Opus 5	Claude Code
NTA	GPT-5.6-sol → Kimi K3	Codex → Kimi Code
NTA	Kimi K3 → GPT-5.6-sol	Kimi Code → Codex
NTA	GLM-5.3 → DeepSeek-V4-Flash	Claude Code → DeepSeek Harness
NTA	DeepSeek-V4.1-Flash → GLM-5.3	DeepSeek Harness → Claude Code
NTA	Claude Opus 5 → GPT-5.6-sol	Claude Code → Codex

At $k = 1, 3, 5$, mean gain retention is 63.15%, 93.35%, and 97.83%, with experiment fractions 24.76%, 54.88%, and 73.77%. At $k = 5$, individual configurations retain 96.11%–99.20% (Opus: 98.74%), motivating candidates $1 \leq k \leq 5$.

For eleven configurations, fifty caps ($k = 1, \ldots, 50$), and two statistics, $M_{\text{cmp}} = 1100$. At $\delta = 0.05$ and $k = 5$, 87.84%–98.67% of tasks retain at least 90% of gains; Appendix B.1 gives simultaneous lower bounds of 65.08%–83.30%.

C Experimental Setup and Reproducibility

We collect the execution environment, agent configurations, baseline settings, evaluation rules, and repeat protocol used throughout the experiments. Empirical findings are separated into Appendices D–G.

C.1 Humanize and Execution Environment

Terminology. A configuration specifies the agent or agent pair, starting order, cap, and budget. A task run is one complete execution on one task; a repeat evaluates the full task suite. Each agent session is modeled as an option, with experiments counted within it; a model response is one recorded output from a model call.

Framework. We implement the workflow on Humanize (Liu et al., 2026), which coordinates agent execution in a shared workspace, fixed-k interruption, and final review as specified in Algorithm 1. The model harness supplies the internal agent policy; the workflow controls option boundaries and preserves shared task workspace. HMA uses $k = 5$ and $t_{\text{total}} = 21600$ seconds per task, including $t_{\text{rev}} = 900$ seconds reserved for the other agent's final review. Thus exploration ends at $t_{\text{s}} = 20700$ seconds.

Hardware. Our execution environment exposes 30 vCPUs backed by an Intel Xeon Platinum 8358 processor at 2.60 GHz, 220 GiB RAM, and one NVIDIA A10 GPU. The GPU reports 23028 MiB of available device memory. The installed NVIDIA driver is version 570.148.08 and reports CUDA 12.8 support. These resources are shared by the two alternating agents within a single task run.

C.2 Agent and Baseline Configurations

Agents. Table 4 lists the full model names and native harnesses. In analyses and figures, GPT, Kimi, GLM, and Opus abbreviate GPT-5.6-sol, Kimi K3, GLM-5.3, and Claude Opus 5; DeepSeek-V4 and DeepSeek-V4.1 abbreviate the corresponding Flash models. DeepSeek alone denotes the version specified for the configuration. All agents in our runs use `max` reasoning effort. Codex, Claude Code, Kimi Code, and DeepSeek Harness provide the model execution interfaces (DeepSeek-AI, 2026; Bolin, 2026; Anthropic, 2026b; Moonshot AI, 2026). Appendix B.2 separately lists the goal and NTA configurations used for prefix analysis.

Table 4: Model–harness mappings and reasoning effort for our runs.

Model	Harness	Reasoning effort
GPT-5.6-sol	Codex	`max`
Kimi K3	Kimi Code	`max`
GLM-5.3	Claude Code	`max`
DeepSeek-V4-Flash	DeepSeek Harness	`max`
DeepSeek-V4.1-Flash	DeepSeek Harness	`max`
Claude Opus 5	Claude Code	`max`

Comparison baselines. We include ScienceFlow, MLEvolve, and ML-Master 2.0 using their reported results (Zhao et al., 2026; Du et al., 2026; Zhu et al., 2026). Table 5 records their original configurations; the `max` reasoning setting applies to our runs, while these published results retain their reported inference settings.

C.3 Evaluation Protocol and Repeats

Evaluation metrics. Lite, Medium, High, and All denote the 22-, 38-, 15-, and 75-task sets, respectively. Lite, Medium, and High report medal percentages within each complexity group; All reports the medal percentage over the full 75-task suite. For repeated evaluations, rates are calculated from per-repeat medal counts before averaging. Table 2 reports all performance metrics to one decimal place. Within a model pair, *Avg. of Single* averages the two constituent goal means with equal weight; HMA gains are computed against this mean before rounding. Gold is the percentage

Table 5: Configurations of the three comparison baselines, as reported in their original papers. Resource allocations and time budgets differ across methods.

Baseline	Model and time budget	Execution resources	Key configuration
ScienceFlow	DeepSeek-V4-Flash-Preview; 24 h	16–32 logical CPU cores, 256 GB RAM, at most 2 GPUs	Recoverable workspaces, ESTRA re-anchoring, and resource-aware execution control.
MLEvolve	Gemini-3.1-Pro-preview; 12 h	21 vCPUs, 234 GB RAM, 1 NVIDIA H200	Temperature 1.0; 500 expansion steps; 3 parallel branches and 3 initial drafts.
ML-Master 2.0	DeepSeek-V3.2-Speciale; DeepSeek-V3.2 (thinking) for context promotion; 24 h	36 AMD EPYC vCPUs and 2 RTX 4090 GPUs per agent; 1008 GB RAM and 1 TB SSD shared across 4 tasks	Hierarchical cognitive caching; prior wisdom initialized from 407 Kaggle competitions excluding MLE-bench.

of tasks attaining a gold medal. Med+ is the percentage outperforming the human leaderboard median. Mean percentile averages the task-level leaderboard percentile over the evaluated suite, using the rank convention in Appendix B.2. All metrics use the workflow's returned candidate; for HMA this is the final review selection or its fallback in Algorithm 1. Test-set scores are computed offline.

Repeated evaluations. The evaluation protocol specifies three independent repeats per configuration, each covering 75 task runs. The current goal estimates in Table 2 use three repeats per model. For a per-repeat suite metric z_r on the fixed task suite, we report its mean $\bar{z}$ and standard error $\mathrm{SE}(\bar{z}) = \sqrt{\sum_{r=1}^{n_{\mathrm{rep}}}(z_r - \bar{z})^2/[n_{\mathrm{rep}}(n_{\mathrm{rep}}-1)]}$. For *Avg. of Single*, independent per-model means give $\mathrm{SE}((\bar{z}_A + \bar{z}_B)/2) = \frac{1}{2}\sqrt{\mathrm{SE}(\bar{z}_A)^2 + \mathrm{SE}(\bar{z}_B)^2}$. Split and overall medal-rate errors are computed from per-repeat counts; Gold, Med+, and mean-percentile propagation uses the adopted source-reported goal errors before display rounding. Published baseline uncertainties retain their original definitions.

D Additional Evidence for the Motivation

We provide the detailed evidence behind the two motivation claims in Section 2: response-length decay in uninterrupted runs and harness staleness across model versions.

D.1 Declining Response Length in Single-Agent Runs

Configurations. We document Figure 2 using the six configurations in Table 4. The `/goal` protocol runs one agent until natural termination or exhaustion of six hours of active runtime, without model alternation or fixed-cap interruption. Each of the six configurations is evaluated on 75 tasks with three independent repeats and `max` reasoning effort. Recorded cohorts and prompt policies are retained.

Response-level moving average. Figure 2 reports mean output tokens per recorded model response. At one-minute active-runtime checkpoints, we divide the sum of provider-reported output tokens by the number of responses in the preceding 30 minutes, pooling the 75 tasks and repeats within each configuration. Near the origin, the window uses the elapsed prefix; windows without responses are undefined. This response-weighted statistic excludes tool-return text, input tokens, and coordinator output, while retaining model responses that request tools.

Early versus late responses. Table 6 compares responses in hours 0–2 and 3–5 using raw event totals divided by response counts in each interval. All six configurations show a reduction of 29.8%–45.0%; these interval statistics are computed before temporal smoothing.

Table 6: **Mean output tokens per recorded response.** Change is relative to hours 0–2. The same task suite and repeats are used as in Figure 2.

Model	Hours 0–2 tokens/response	Hours 3–5 tokens/response	Change (%)
GPT-5.6-sol	336.5	213.6	−36.5
Claude Opus 5	1008.3	554.8	−45.0
GLM-5.3	658.4	388.5	−41.0
DeepSeek-V4.1-Flash	1137.5	696.1	−38.8
Kimi K3	585.7	390.2	−33.4
DeepSeek-V4-Flash	682.2	479.2	−29.8

D.2 Harness Staleness across Model Versions

Comparison setup. We compare ML-Master 2.0 (Zhu et al., 2026), MLEvolve (Du et al., 2026), and ScienceFlow (Zhao et al., 2026) with the official DeepSeek Harness on a 16-task MLE-bench subset (Chan et al., 2025), using DeepSeek-V4-Flash and DeepSeek-V4.1-Flash. We retain the research harnesses' original workflow designs and experimental settings while disabling task-specific cold-start knowledge wherever present. Execution budgets are reported in Table 1. DeepSeek Harness runs each model under `/goal` with `max` reasoning effort for six hours; its reported medal rate is averaged over three repeats.

Result accounting. Medal rates use all 16 tasks as the denominator. The disqualified ScienceFlow run on Chaii with DeepSeek-V4.1-Flash is counted as no medal because its checkpoint had seen the test answers. Table 1 reports medal rates and percentage-point differences from the corresponding DeepSeek Harness baseline. Table 7 provides the task-level outcomes. These 16-task DeepSeek Harness references are distinct from the full 75-task results in Table 2.

Table 7: **Task-level outcomes for the harness comparison.** DeepSeek Harness entries report the number of runs earning a medal out of three; its medal total is the mean count. The last row divides each medal total by 16, matching Table 1. Other columns report medal classes from the retained run: G, S, and B denote gold, silver, and bronze; — denotes no medal, × denotes the disqualified ScienceFlow V4.1 Chaii run. V4 and V4.1 denote DeepSeek-V4-Flash and DeepSeek-V4.1-Flash. When a run did not nominate a final artifact, the recorded score-independent rule selects its latest eligible submission.

Workflow	**DeepSeek Harness**		**ScienceFlow**		**ML-Master 2.0**		**MLEvolve**	
Task	V4	V4.1	V4	V4.1	V4	V4.1	V4	V4.1
3D Object Detection	1/3	2/3	S	**G**	—	S	—	—
APTOS Blindness Detection	0/3	3/3	S	B	—	**G**	—	S
Toxic Comment Classification	0/3	3/3	—	**G**	—	—	B	S
Leaf Classification	1/3	3/3	S	S	—	—	B	**G**
Billion Word Imputation	0/3	3/3	—	**G**	—	—	—	—
Cdiscount Image Classification	0/3	1/3	—	—	—	—	—	—
Chaii Question Answering	0/3	1/3	—	×	—	—	—	—
Freesound Audio Tagging	1/3	1/3	—	—	—	—	—	S
iMet	0/3	1/3	—	S	—	—	—	—
iWildCam	2/3	3/3	—	**G**	—	S	**G**	S
Kuzushiji Recognition	2/3	3/3	B	—	—	**G**	—	—
Learning Agency Essay Scoring	0/3	3/3	—	**G**	—	**G**	—	—
LMSYS Chatbot Arena	0/3	2/3	—	S	—	—	—	—
PetFinder Pawpularity	0/3	3/3	—	**G**	—	S	—	S
Tweet Sentiment Extraction	0/3	2/3	—	S	—	—	—	S
US Patent Phrase Matching	1/3	3/3	—	—	S	—	—	S
Medals	**2.7**	**12.3**	**4**	**11**	**1**	**6**	**3**	**8**
Medal rate (%)	16.7	77.1	25.0	68.8	6.3	37.5	18.8	50.0

E Additional Main-Comparison Analyses

We supplement the main comparison with the aggregation rules, response and performance trajectories, matched constituent weighting, and task-level comparisons used for Figure 4.

Analysis configurations. We analyze the five participating goal models and six HMA configurations in Table 2, using the protocol in Appendix C. The trajectory aggregation and matched constituent construction follow the definitions below. The separate scheduling ablations appear in Appendix F.

Performance and response trajectories. At each active-runtime checkpoint, we aggregate the medal status of the latest accepted candidate on each task, assigning zero before a valid submission. Final points use the returned candidate under the recorded selection rule. Scores use the corrected answer keys. Receipt processing at the nominal deadline is included in the final point. Figure 4 (a) uses the response-level statistic in Appendix D.1: at each one-minute checkpoint, total actor output tokens in the preceding 30-minute trailing window are divided by the number of recorded responses, pooling tasks and repeats within each configuration and including HMA's final review. Near the origin, the window uses the elapsed prefix; empty windows are undefined. Figure 4 (b) retains the unsmoothed medal-rate trajectory. Token counts measure output volume across different tokenizers, not a common monetary or computational cost. Neither panel extrapolates trajectories or uses best-so-far score selection.

Matched group-average curves. Figure 4 assigns weight $1/6$ to each HMA configuration. For configuration j with constituent models A_j, B_j, its goal reference is $(Y^{\mathrm{goal}}_{A_j}(t) + Y^{\mathrm{goal}}_{B_j}(t))/2$, where Y denotes mean output tokens per response or medal rate, computed within each configuration as above. We average these six references at each common runtime checkpoint. Equivalently, the goal weights are $5/12$ for GPT, $1/6$ each for Opus, DeepSeek-V4.1, and Kimi, and $1/12$ for GLM. DeepSeek-V4-Flash is excluded because it appears in none of the six HMA configurations. This matches the mean-of-constituents reference used for the gains in Table 2; the weights describe configuration composition, not measured execution-time shares. Both panels use active runtime horizontally, with mean output tokens per response in (a) computed over a 30-minute trailing window and mean medal rate in (b). Faint curves retain the participating configurations. For the early/late comparison, we divide raw output-token totals by response counts separately in hours 0–2 and 3–5 within each configuration, then apply the same group weights. The resulting means are 610.9 and 527.6 tokens per response for HMA, and 650.3 and 394.9 for goal, retaining 86.4% and 60.7%, respectively; these interval statistics are computed before temporal smoothing.

Early progress. At two hours, the six HMA configurations attain 57.3%, 58.7%, 54.7%, 60.0%, 50.7%, and 42.7% medals; their respective final rates are 78.2%, 73.3%, 72.0%, 70.7%, 69.3%, and 62.7%.

Task-level gains and regressions. Across matched GPT goal comparisons, GPT–Kimi HMA gains medals on 5, 4, and 6 tasks and loses them on 4, 3, and 2, yielding net gains of 1, 1, and 4. Its successful tasks remain within the evaluated goal-task union, characterizing gains within the six-hour budget.

F Ablation Studies

We supplement Section 4.3 with starting-order effects, executed fixed-k comparisons, natural-termination statistics, and a selection-only final review audit. We discuss order before cap size because model exposure within a finite time budget depends on both.

F.1 Starting-Order Effects

We compare GPT-first and Kimi-first alternation under NTA and HMA with $k = 5$, then relate the endpoint differences to model-session time and partner participation.

Configurations. Each configuration covers the same 75 tasks with a six-hour budget and `max` reasoning effort. The additional Kimi-first HMA run records 283 exploration options: among the 75 initial options, 52 reach the cap, 22 end naturally, and one reaches the deadline. The GPT-first $k = 5$ run records 277 exploration options, with 66 initial cap closures and nine natural exits. Corrected answer keys give 42/75 medals for Kimi-first NTA, including three corrected outcomes. For the Kimi-first HMA archive, the terminal window is 21,700 seconds; all accepted candidates arrive within 21,600 seconds and its six-hour medal count matches the returned result.

Final quality and finite-budget exposure. Figure 6(a) compares both GPT–Kimi starting orders under the six-hour budget. NTA reaches 77.3% and 56.0% medals with GPT and Kimi first, respectively; HMA with $k = 5$ reaches 70.7% and 66.7%, narrowing the observed gap from 21.3 to 4.0 percentage points. The six-hour horizon limits sustained participation by both agents under NTA: by 5.75 hours, the second agent has produced a recorded response on only 19/75 GPT-first tasks and 47/75 Kimi-first tasks. Under NTA, the nominal first model occupies about 81–83% of the budget. Because the GPT goal baseline is stronger than the Kimi baseline (68.0% versus 54.2%), the longer uninterrupted GPT exposure in the GPT-first archive is associated with the higher NTA result. GPT-first HMA instead allocates 51.1% to GPT and 46.3% to Kimi, whereas Kimi-first HMA still allocates 81.3% to Kimi and only 16.2% to GPT. These finite-budget differences accord with the startup effects in Appendix A.2; improvement from interruption remains conditional on continuation advantage (Theorem 3.5).

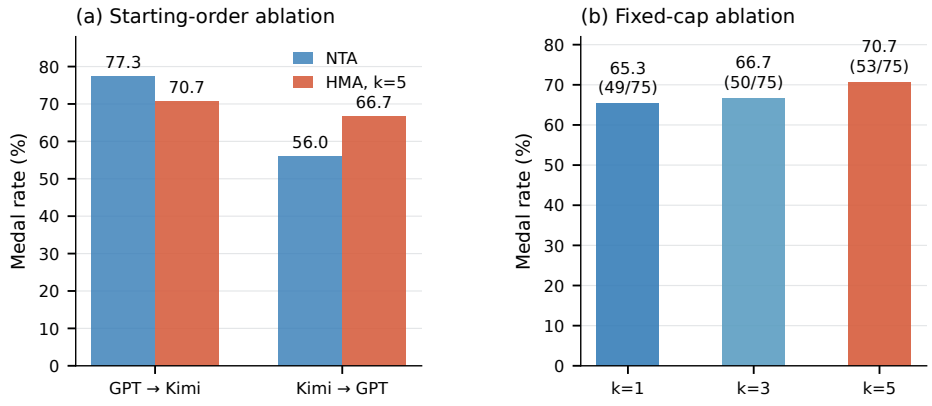

Figure 6: **Fixed-cap interruption: starting-order and cap ablations.** Both panels use the same 75-task, six-hour GPT-5.6-sol–Kimi K3 suite. (a) Starting-order comparison: NTA and HMA with $k = 5$; the cap narrows the observed order gap. (b) GPT-first HMA with $k \in \{1, 3, 5\}$; the new A10 runs produce 49/75 and 50/75 medals at $k = 1$ and $k = 3$, while the existing $k = 5$ run produces 53/75. Values are final medal rates, with counts shown in the cap panel.

Model time allocation. We reconstruct occupied model-session intervals from native session and controller logs over the first six hours. Native creation and completion records determine session endpoints. For an interrupted session without a completion record, we use the next session or option start, or the active attempt's end, whichever occurs first. Intervals are intersected with active-runtime windows; pause gaps and auxiliary subagent sessions are excluded, and final review is attributed to

its executing model. If $w_j^{(u)}$ is the resulting model-u duration on task j, its mean budget share is $100 \sum_{j=1}^{75} w_j^{(u)} / (75 \cdot 21600)$. All returns to a model are accumulated. Unallocated time averages 0.07% and 0.15% for GPT-first and Kimi-first NTA, and 2.63% and 2.49% for the corresponding HMA runs. These shares include tool execution and waiting within sessions.

Recorded starting-order exceptions. Workflow labels retain the archived nominal configurations. In the GPT-first NTA archive, seven of 75 tasks have Kimi as the earliest recorded native session; the other NTA configuration and both HMA runs match their nominal initial models on all 75 tasks. Runtime attribution follows the recorded model, and all tasks remain included. Thus the historical NTA groups are configuration comparisons rather than perfectly controlled starting-order interventions.

Designated-partner participation. We measure participation from the first recorded response by the nominal second agent (Kimi in the GPT-first configuration and GPT in the Kimi-first configuration), excluding `turn-review` sessions. This measures observable access to search, rather than an exact controller launch time. Tasks with no such response are right-censored at six hours; counts retain all 75 tasks in the denominator. Table 8 reports these response-based timing statistics separately from the session-duration estimates above. With GPT first, the second agent's mean within-task share of output tokens before hour 5.75 rises from 13.0% under NTA to 36.1% under HMA. A cap limits experiments per option; it does not impose a fixed wall-clock duration on an experiment.

Table 8: **The cap advances access to the second agent.** All counts are out of 75 tasks and exclude final review. By the six-hour deadline, 22 GPT-first NTA tasks and 18 Kimi-first NTA tasks still have no recorded response from the second agent.

	Tasks with a second-agent response			Median first response (h)	Final medals (%)
Workflow	By 3 h	By 5 h	By 5.75 h		
GPT → Kimi NTA	17	18	19	5.97	77.3
Kimi → GPT NTA	15	27	47	5.56	56.0
GPT → Kimi HMA	55	69	75	1.89	70.7

F.2 Effects of the Fixed Interruption Cap

We hold GPT as the initial agent and compare executed HMA policies with $k \in \{1, 3, 5\}$, complementing the observed-prefix analysis in Appendix B.

Configurations and results. Each cap contributes one repeat over 75 tasks with the same model pair, `max` reasoning effort, and nominal six-hour budget including final review. The $k = 1$ and $k = 3$ archives use a 21,700-second terminal window. Table 9 reports the final medal counts and rates, also plotted in Figure 6(b). These cap configurations and the reverse-order configuration are used for scheduling ablations; the main comparison retains its six configurations.

Table 9: $k = 5$ **gives the highest medal rate among the tested finite caps.** GPT-first HMA alternates GPT-5.6-sol and Kimi K3 on the same 75 tasks under the nominal six-hour budget.

Cap k	**Medals / 75**	**Medal rate (%)**
1	49	65.3
3	50	66.7
5	**53**	**70.7**

Empirical selection within the candidate range. The prefix statistics in Appendix B motivate candidates $1 \le k \le 5$. Among the executed GPT-first runs, $k = 1, 3, 5$ yield 65.3%, 66.7%, and 70.7% medals, respectively, against the GPT goal mean of 68.0%. These results support selecting

$k = 5$ within the tested finite caps for this order and horizon. Theorem 3.5 gives the corresponding conditional advantage criterion; the prefix statistics guide candidate selection without directly estimating its continuation values.

Interaction with time and starting order. A larger cap permits longer uninterrupted experiment blocks before handoff, while the six-hour deadline limits the partner's subsequent search. Given GPT's stronger standalone performance, the GPT-first cap trend and higher GPT-first NTA result are consistent with a benefit from preserving more continuous GPT search. Conversely, $k = 5$ improves the Kimi-first result and reduces sensitivity to the initial agent. The limited horizon can therefore leave the pair's complementary capabilities incompletely exploited, making cap size and initial-agent choice jointly relevant. The recorded starting-order exceptions in Appendix F.1 delimit this interpretation.

F.3 Natural Termination before the Cap

We report option-closure outcomes from terminal controller receipts and evaluator ledgers for the six main-table HMA configurations. One complete repeat per configuration, using the first Opus-first GPT repeat, yields 450 task runs with the same answer keys and final-candidate rule as Table 2.

Option closure. Table 10 separates the first option from later options and distinguishes reaching five accepted experiments, natural termination before five, and closure at the exploration deadline. Final review sessions are excluded. Initial options contain 57 natural exits among 446 closures before the deadline (12.8%); later options contain 598 among 992 (60.3%). Across all options, 655/1,438 closures before the deadline are natural (45.5%). Of the 655 natural exits, 248 produce no accepted experiment. These are retained in the audit. One GPT–DeepSeek deadline receipt omits the cap field following a controller timeout; its accepted count is zero, and the recorded standing submission agrees with the adopted fallback. All 1,888 recorded option counts are at most five.

Table 10: **Observed option-closure reasons at $k = 5$.** Each configuration has 75 initial options. Natural-exit percentages exclude deadline closures because the latter censor whether an option would reach the cap or terminate naturally. The first listed model starts each workflow.

	Initial option			Later options		
Configuration	Cap	Natural	Deadline	Cap	Natural	Deadline
Opus → GPT	61	13	1	56	121	74
GPT → Opus	72	2	1	89	69	74
GPT → DeepSeek-V4.1	71	4	0	88	132	75
GPT → Kimi	66	9	0	56	71	75
GPT → GLM	54	19	2	71	36	73
DeepSeek-V4.1 → Kimi	65	10	0	34	169	75
All configurations	389	57	4	394	598	446

F.4 Final Review and Historical Any-Medal Retention

We evaluate retention of historically attained any-medal outcomes on the same 450 task runs as Appendix F.3, supplementing the mechanism in Section 3.4.

Selection-only comparison. We hold each recorded exploration trajectory fixed and compare the returned candidate with the candidate standing immediately before final review. The reserved final review time is not reallocated to search, so this measures the effect of final selection rather than a rerun with a longer exploration budget.

Historical any-medal retention. For each task run, historical any-medal status is one if any accepted exploration candidate earns bronze or better, and zero otherwise. We compute this status from offline evaluator records before final review; repeated receipts for a reviewed candidate add no new historical outcome. Historical any-medal retention is the fraction of historically medal-positive task runs whose selected candidate earns a medal. Table 11 compares this fraction before and after

final review. Of 450 task runs, 330 produce a medal-winning candidate; final review raises retained medals from 311/330 (94.2%) to 319/330 (96.7%), recovering eight medals without losing any held immediately before final review.

Table 11: **Final review improves retention of historical any-medal outcomes.** Each configuration contains 75 task runs. Historical counts runs with at least one medal-winning exploration candidate; retention divides selected medal counts by Historical. Before and after refer to the standing candidate and final review's returned candidate. Missed counts historical medal outcomes not retained after final review.

Configuration	Changed	Historical any medal	Selected medals before → after	Retention (%) before → after	Missed after
Opus → GPT	4	59	$58 \rightarrow 58$	$98.3 \rightarrow 98.3$	1
GPT → Opus	6	57	$55 \rightarrow 55$	$96.5 \rightarrow 96.5$	2
GPT → DeepSeek-V4.1	11	56	$51 \rightarrow 54$	$91.1 \rightarrow 96.4$	2
GPT → Kimi	6	54	$52 \rightarrow 53$	$96.3 \rightarrow 98.1$	1
GPT → GLM	9	54	$48 \rightarrow 52$	$88.9 \rightarrow 96.3$	2
DeepSeek-V4.1 → Kimi	4	50	$47 \rightarrow 47$	$94.0 \rightarrow 94.0$	3
All configurations	40	330	$311 \rightarrow 319$	$94.2 \rightarrow 96.7$	11

Across all 450 runs, the returned candidate matches the historical any-medal status in 439 (97.6%), including 120 runs that never produce a medal. The conditional retention rate of 96.7% excludes these 120 runs. Final review changes 40 selections (8.9%); 311/319 final medals (97.5%) were already held before final review. Thirty-five GLM-pair final reviews are silent and one DeepSeek-pair final review fails; their recorded fallback candidates remain included. This audit measures historical any-medal retention at final selection.

Medal recovery and remaining misses. On MBM-01 with GPT–DeepSeek-V4.1, the first GPT option reaches 0.84671 without a medal, and DeepSeek reaches bronze at 0.85445. The returning GPT replaces the standing result with 0.75795; final review restores the archived DeepSeek candidate and recovers the medal. On MBM-20 with GPT–GLM, GPT produces a gold-medal candidate at 0.84835, but the final GLM candidate scores 0.25403; final review restores 0.84835. In aggregate, final review reduces missed historical medals from 19 to 11, supporting its role in retaining search outcomes at final selection.

G CASE STUDIES OF HETEROGENEOUS ALTERNATION

We examine four model pairs across five model families: GPT–Opus, GPT–DeepSeek, GLM–Kimi, and DeepSeek–Kimi. The INGV case in Section 4.4 is followed by examples from text prediction, image regression, and satellite positioning.

Selection and measurement. We selected task runs with inspectable artifacts, controller logs, evaluator receipts, and native response records. GPT–Opus, GPT–DeepSeek, and DeepSeek–Kimi occur in the main comparison; GLM–Kimi is an additional combination using the same $k = 5$ workflow. Following the trajectory analysis of Zhao et al. (2026), we align each accepted candidate with its producing option using controller counts and receipt times. Test scores are evaluated offline and were hidden from the agents; all accepted scores, including regressions, are plotted. Goal references use three runs per constituent. The appendix figures compare disjoint first- and last-20-response windows; options with fewer than 40 responses show their counts instead. Final review is excluded from token statistics and preserves the standing candidate in all four cases. Response renewal is compared within the same model, since tokenizers and policies differ. These cases illustrate observed mechanisms rather than estimating their average causal effect.

G.1 INGV VOLCANIC ERUPTION PREDICTION

Repeated refinement. On INGV (MBH-07), GPT and Opus complete seven exploration options with 5, 5, 5, 2, 5, 1, and 4 accepted experiments and 77, 263, 251, 59, 110, 49, and 48 responses, respectively. Options 1, 2, 3, and 5 reach the cap; options 4 and 6 end naturally, and option 7 reaches the deadline. Their best MAEs are 1,123,915, 41,214, 39,061, 39,878, 37,422, 36,934, and 34,234: five of the six post-handoff options improve the historical best, while the fourth option regresses. Shared features and cached predictions support successive refinements. The trace records GPT reusing and pruning Opus's retrieval features, Opus adding consistency constraints for paired windows, GPT introducing an uncertainty-based residual gate, Opus refining its calibration, and GPT enforcing joint temporal feasibility.

Score comparisons and plot conventions. After the first Opus session reaches 41,214 MAE, subsequent alternation lowers it to 34,234, a further 16.9% improvement and 46.5% below the best of six constituent goal runs (63,991). Native MAE resolves progress beyond the already-gold goal results. Figure 5(a) retains all 27 accepted scores, including regressions, without smoothing; it uses a logarithmic axis with a linear inset for late options. Horizontal lines mark the gold threshold (3,971,366), the minimum MAE in MLE-bench's 621-entry reference leaderboard (3,438,191), and the best constituent goal endpoint. Blue and purple regions identify GPT and Opus, and dotted vertical lines mark handoffs.

Response measurement and renewal. In Figure 5(b), each point averages output tokens per response from the preceding accepted submission or option start to the current acceptance. The 27 intervals contain 826 responses, excluding 31 trailing responses; final review is excluded. Comparing the first experiment after each return with the same model's preceding last experiment, mean response length increases on four of five returns by a factor of 1.6–3.6; the final Opus return has a ratio of 0.94.

G.2 GOOGLE QUEST Q&A LABELING: GPT–DEEPSEEK

Repeated reuse of text-model predictions. On Google QUEST Q&A Labeling (MBM-10), GPT-5.6-sol and DeepSeek-V4.1-Flash complete seven options, five of which produce 23 accepted candidates. GPT initializes sparse-text and encoder models; DeepSeek completes additional DeBERTa fits and combines their saved predictions. Subsequent sessions reuse these artifacts for target-specific blending, calibration, and larger-encoder refinement. The first five options achieve best Spearman correlations of 0.45664, 0.47124, 0.48741, 0.50381, and 0.50760. The returned candidate scores 0.50747, versus 0.47252 for the best of six constituent goal runs (Figure 7). At the first returns, GPT's mean response length rises from 168.9 to 529.1 tokens and DeepSeek's from 647.2 to 1060.9; GPT's next return instead falls from 616.7 to 339.2. The final two options produce no accepted experiment, so neither response renewal nor useful progress occurs at every handoff.

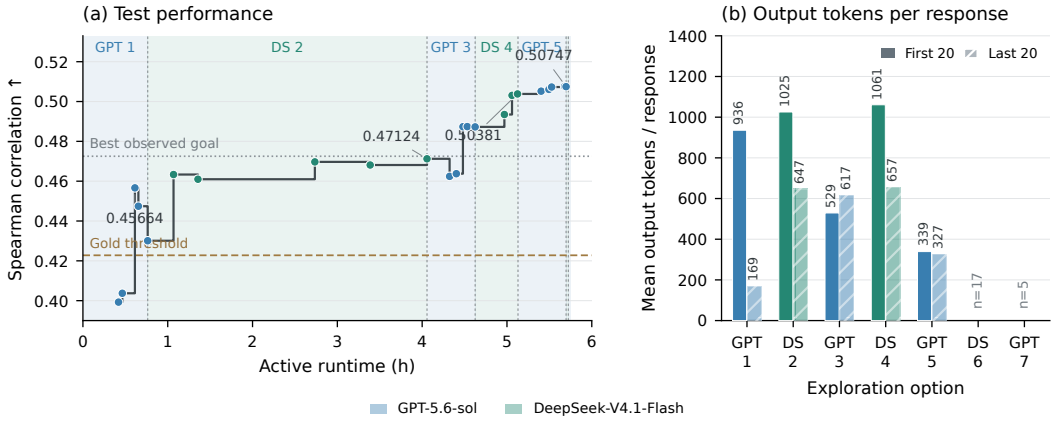

Figure 7: **GPT–DeepSeek refinement on Google QUEST.** (a) All 23 accepted scores, including regressions. Shading identifies the active model; reference lines mark the best constituent goal endpoint and gold threshold. (b) First- and last-20-response means per option. Options 6 and 7 have only 17 and 5 responses and no accepted experiments; their response counts replace overlapping windows. Final review retains the standing candidate and is excluded from token statistics.

G.3 PetFinder Pawpularity: GLM–Kimi

Extending a shared prediction pool. On PetFinder Pawpularity (MBM-27), GLM-5.3 builds CLIP and DINOv2 feature caches, regressors, and calibrated predictions before Kimi K3 takes over at 2.40 hours. Kimi reuses the out-of-fold predictions in a constrained stack and adds concatenated-feature ridge and MLP members, improving the best test RMSE from 16.87437 to 16.78333. Returning GLM explores further combinations and reaches 16.77953; the returned candidate scores 16.85639 and earns gold (Figure 8). Neither constituent earns a medal in its three goal runs. The three productive options contain 5, 5, and 3 experiments; the final Kimi option adds none. GLM's returning first 20 responses average 724.5 tokens, compared with 503.8 in its preceding option's last 20. The final score regression remains visible: artifact reuse improves the search trajectory without ensuring that the returned candidate is its best.

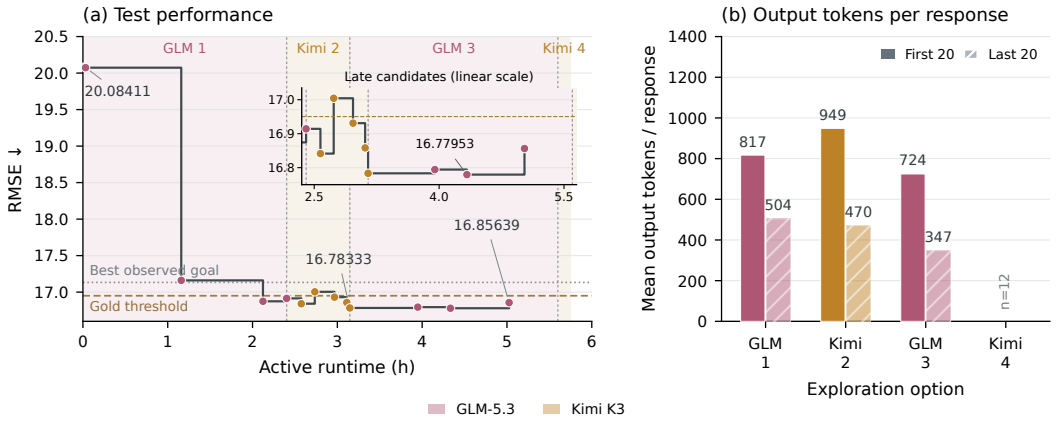

Figure 8: **GLM–Kimi cooperation on Pawpularity.** (a) Every accepted test RMSE, with a linear inset resolving the late candidates. The dotted and dashed lines show the best of six constituent goal endpoints and the gold threshold. (b) Response-length windows for the three productive options; the last option has only 12 responses. The final candidate remains below the gold threshold despite regressing from the best observed RMSE.

G.4 GOOGLE SMARTPHONE DECIMETER: DEEPSEEK–KIMI

Late correction after an unsuccessful refinement. On the Google Smartphone Decimeter Challenge 2022 (MBH-12), DeepSeek-V4.1-Flash builds a GNSS fusion pipeline with Doppler constraints and robust optimization, reaching 4.33091 mean haversine error. Kimi K3 improves it to 4.28914, but DeepSeek's next option leaves a worse candidate at 5.35529. Returning at 5.08 hours, Kimi retains the shared solver, checks the mean-error objective, and replaces straight-line extrapolation through an end-of-log outage with a direction-matched route from public training trajectories. Its accepted candidate reaches 2.77310 and earns bronze, a 42.1% reduction from the best of six constituent goal endpoints (4.78536), none of which earns a medal (Figure 9). Kimi's first 20 returning responses average 847.9 tokens, below the preceding Kimi option's last-20 mean of 1010.0. This late quality gain therefore does not require a response-length rebound.

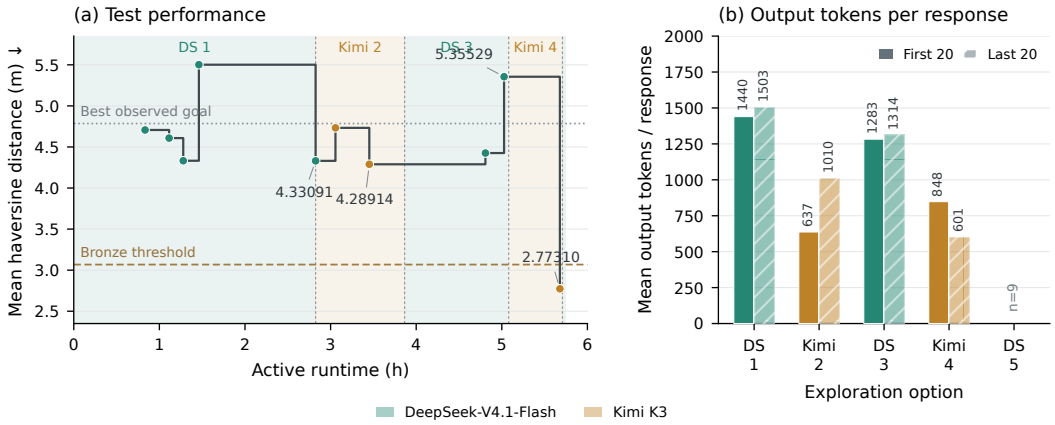

Figure 9: **DeepSeek–Kimi correction in smartphone positioning.** (a) All ten accepted mean-haversine scores. The fourth option repairs the late outage and crosses the bronze threshold after the third option regresses. The dotted line is the best constituent goal endpoint. (b) First- and last-20-response means; the final DeepSeek option contains nine responses and no accepted experiment. Final review retains Kimi's repaired candidate.