A SOBER LOOK AT AGENTIC MISALIGNMENT IN AUTOMATED WORKFLOWS

Anonymous authors
Paper under double-blind review

ABSTRACT

We study a class of emergent misalignment in multi-agent systems (MAS), with a focus on automated workflows, which we refer to as *agentic misalignment*. Although these systems can solve complex tasks, their agents can act inconsistently with assigned roles and intended human goals. We formally define these behaviors and analyze them within a Bayesian framework, showing how similar priors and likelihoods can leave roles insufficiently separated. To address this issue, we propose Agentic Evidence Attribution (AEA), an alignment framework that uses context-specific evidence to improve role adherence. AEA reasons over agent action traces and provides structured evidence to identify misaligned behavior through evidence injection. To better understand the role of evidence, we study two instantiations of AEA, self-reflection using the workflow model and weak-to-strong generalization using a small trained evidence model. Across eight task benchmarks and two attribution datasets, we show that specialized failure attribution improves workflow alignment to different degrees. Our results show that evidence-based alignment is necessary and can improve agent collaboration and reliability in automated workflows.

1 INTRODUCTION

Multi-agent systems (MAS) with large language models (LLMs) are increasingly used for tasks that require decomposition (Prasad et al., 2024; Erdogan et al., 2025), tool use (Qin et al., 2024; Yuan et al., 2025), and multi-step reasoning (Aksitov et al., 2024; Plaat et al., 2025). Many recent MAS use an orchestrator (or meta-agent) to construct agent roles and arrange their interactions through automated workflow generation (Li et al., 2024; Zhang et al., 2025b). Although these workflows allow agents to handle complex objectives, recent studies have reported that MAS often fail in systematic ways (Cemri et al., 2025; Han et al., 2024; Kim et al., 2025; Lynch et al., 2025). These failures arise even when individual agents perform well in isolation, suggesting an underlying coordination problem within the agentic workflow. We study *agentic misalignment* in this cooperative setting, where departures from assigned responsibilities undermine the shared task objective.

One qualitative perspective for interpreting these misalignments is that agents act according to a shared utility shaped by generic pretraining and post-training, rather than the role-specific goals required by the workflow (Figure 1, left) (Korbak et al., 2022; Rădulescu et al., 2020; Bengio et al., 2025). In automated workflows, agents observe only partial information about the task objective and the behavior of other agents. Their implicit utility can therefore reflect generic training preferences more than their assigned role. This utility can diverge from the objective of the whole workflow and lead to reward hacking (Skalse et al., 2022), including effort minimization, incomplete hand-offs, and short-term decisions that harm other agents' goals (Weng, 2024; Guo et al., 2024). These behaviors make automated workflows difficult to control and limit their reliability in deployment.

To better understand these behaviors, we model the multi-agent collaboration process as Bayesian inference over latent utility. We hypothesize that each agent infers a posterior utility based on workflow system prompts, intermediate states, and observations of other agents during execution. When this evidence is sparse or generic, the inferred utility can deviate from the role-specific objective. Our analysis characterizes when similar priors and likelihoods leave the agents' roles insufficiently separated. We examine the behavioral implications through role adherence, correction outcomes, and output variability.

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.

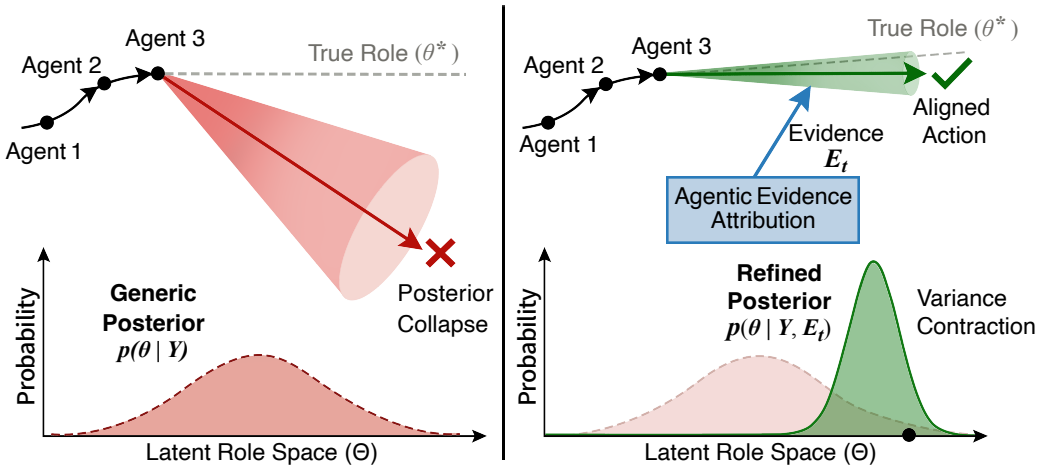


Figure 1: An analytical view of agentic misalignment and evidence-based correction. **Left** Similar role beliefs can leave agents insufficiently differentiated. **Right** Role-specific evidence E_t distinguishes the intended role and supports aligned decisions under the conditions in Section 4.

Based on these insights, we propose *Agentic Evidence Attribution* (AEA), an alignment framework that provides additional structured evidence about each agent's role (Figure 1, right). AEA analyzes turn-level traces from completed MAS trajectories, identifies which actions support or harm the task, and assigns context-aware and role-specific feedback. It revises only the identified agent's prompt and restarts the workflow once. Because AEA operates on workflow traces and prompts, it can guide proprietary multi-agent systems without access to their internal representations.

We evaluate automated workflows on eight task benchmarks, including programming, data analysis, scientific reasoning, and competition-level mathematics, and measure failure attribution on two trace datasets. We study two instantiations of AEA, *Self-Reflection* using the workflow model and *Weak-to-Strong* generalization using a separate trained evidence model. Matched-backbone training improves attribution and final correctness, while matched-restart comparisons show gains over Naive Retry and an independent critic. These results show that small, specialized evidence models can improve both role adherence and the reliability of stronger automated workflows.

2 Related Work

Agentic Misalignment. Misaligned behaviors in reinforcement learning arise when an agent exploits an imperfect reward function without achieving the human's intended goal (Wang et al., 2025). This reward hacking phenomenon is related to Goodhart's Law (Karwowski et al., 2024) and appears in LLM alignment, where models can exploit spurious preference cues such as verbosity or sycophancy (Ye et al., 2025). Agentic systems inherit these risks. Goal misspecification and goal misgeneralization can cause departures from human intent under distribution shifts (Bengio et al., 2025; Sohl-Dickstein, 2023). These observations motivate a more careful study of reward-driven behavior in multi-agent settings.

Automated Workflows. Automated workflow generation assigns agent roles and determines their interactions through prompting or high-level planners. Errors in role interpretation, task decomposition, and information exchange can compromise these workflows (Cemri et al., 2025). We focus on *centralized MAS*, where an orchestrator generates and coordinates subagents (Kim et al., 2025). Jwalapuram et al. (2026) study the cost and performance of automatically generated MAS, while AEA targets correction within a generated workflow. Papamarkou et al. (2026) advocate Bayesian control at the orchestration layer, while our analytical role-inference model motivates learned evidence for prompt repair.

Reward Modeling for Agents. Reward models guide LLM alignment, with KL-regularized reinforcement learning admitting a Bayesian interpretation (Korbak et al., 2022). Variational Preference Learning infers latent user context to personalize reward models (Poddar et al., 2024), while Optimas aligns component-level rewards with the objective of a compound AI system (Wu et al., 2025). AEA

addresses role-specific decisions within a workflow by extracting diagnostic evidence from agent interactions. This evidence updates the acting model's textual context, and our analysis characterizes conditions under which it can reduce role uncertainty and action error.

Agentic Failure Attribution. `Who&When` benchmarks the identification of responsible agents and decisive error steps (Zhang et al., 2025c). AgenTracer constructs attribution data through counterfactual replay and fault injection, trains a failure tracer, and uses its feedback for iterative correction (Zhang et al., 2025a). COPPER trains a shared reflector with counterfactual PPO to generate role-specific reflections for actor prompts (Bo et al., 2024). GPA uses factorized judges to evaluate alignment among goals, plans, and actions (Jia et al., 2025). Subramanian et al. (2026) attribute multi-turn safety failures to user turns and spans using tiered evidence supervision with counterfactual anchors, whereas AEA identifies responsible agents and repairs their role prompts. AgenTracer's case analysis shows how a visible downstream mistake can obscure an earlier failure (Zhang et al., 2025a). AEA combines Bayesian role analysis with structured learned evidence.

3 A Sober Look at Agentic Misalignment

3.1 Problem Formulation

We formalize the automated workflow as a partially observed multi-agent decision process (POMDP) (Spaan, 2012; Zhang et al., 2025c) augmented with latent role variables. The workflow is described by the tuple $\mathcal{M} = \langle \mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{T}, \Theta, \mathcal{U} \rangle$. Here, $\mathcal{N} = \{1, \ldots, N\}$ represents the set of agent roles, while $\mathcal{S}$ denotes the global state space that consists of the task context, dialogue history, and tool outputs. The joint action space is given by $\mathcal{A} = \times_{i \in \mathcal{N}} \mathcal{A}_i$, where $\mathcal{A}_i$ is the specific action space for agent i. $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ denotes the state transition function with $\Delta(\mathcal{S})$ as the probability simplex over states. To define role-specific behaviors, we introduce $\Theta = \{\theta_1, \ldots, \theta_K\}$ as a finite set of *latent utility types* (e.g., "strict verifier", "accurate approximator"). Lastly, $\mathcal{U} : \mathcal{S} \times \mathcal{A} \times \Theta \to \mathbb{R}$ represents the utility function conditioned on the roles.

At any time step t, a single agent $i = \psi(t)$ is active according to the workflow orchestration. The agent does not observe the true role intention directly. Instead, the ground-truth role assignment is a random variable $\boldsymbol{\theta}_i \in \Theta$, with prior $P(\boldsymbol{\theta}_i \mid i)$. The *oracle* (ideal) policy maximizes the true utility for this latent type, with inactive agents' actions fixed.

$$\pi_i^*(s_t; \theta) \in \arg\max_{a \in \mathcal{A}_i} \mathcal{U}(s_t, a; \theta). \tag{1}$$

Definition 3.1 (Agentic Misalignment via Decisive Errors)**.** Let τ be a trajectory and $Z(\tau) \in \{0, 1\}$ be a binary failure indicator (0 means no failure). A time step t is a *Decisive Error* if $Z(\tau) = 1$, but there exists an action $\tilde{a}_t \in \mathcal{A}_i$ such that the modified trajectory $\tilde{\tau}$ satisfies $Z(\tilde{\tau}) = 0$. *Agentic Misalignment* occurs when the agent's policy $\hat{\pi}_i$ selects a_t (the error) instead of $\tilde{a}_t$ because a_t maximizes the expected utility under the *generic* posterior, distinct from the specific role type $\boldsymbol{\theta}_i$.

$$a_t \in \arg\max_a \mathbb{E}_{\theta \sim P(\cdot|Y)}[\mathcal{U}(s_t, a; \theta)], \tag{2}$$

where $a_t \neq \pi_i^*(s_t; \boldsymbol{\theta}_i)$ and Y is the evidence of the agent.

Under Definition 3.1, a decisive error is an action whose replacement can turn a failed trajectory into a successful one. Multiple agent-step pairs in the same trajectory may satisfy this condition. This definition can be naturally extended into continuous and more complicated scenarios.

3.2 Misalignment as a Role Inference Problem

The core obstacle is that agent i should infer the specific instantiation $\boldsymbol{\theta}_i$ from the available evidence Y_t at time t. We model this as Bayesian inference where the agent maintains a belief state $b_t(\theta) = P(\boldsymbol{\theta}_i = \theta \mid Y_t)$. Similar priors and weak role distinctions in the likelihood can leave agents with similar role beliefs.

We quantify the divergence between two distributions using the Total Variation (TV) distance, $\|P - Q\|_{\mathrm{TV}} = \frac{1}{2} \sum |P(x) - Q(x)|$. To rigorously bound the posterior update, we introduce a stability condition on the marginal likelihood (evidence).

Theorem 3.2 (ϵ-Stability of Role Posteriors). *For roles $k \in \{i, j\}$, let $p_k(\theta) = P(\theta \mid k)$ and $L_k(\theta) = P(Y \mid \theta, k)$ at fixed evidence Y. Assume $\|p_i - p_j\|_{\mathrm{TV}} \le \epsilon_\pi$, $0 \le L_k \le B$, $\|L_i - L_j\|_\infty \le \epsilon_\ell$, and $Z_k = \sum_\theta L_k(\theta) p_k(\theta) \ge \zeta > 0$. Then*

$$\|P(\cdot \mid Y, i) - P(\cdot \mid Y, j)\|_{\mathrm{TV}} \le \frac{2B\epsilon_\pi + \epsilon_\ell}{\zeta}. \tag{3}$$

When the bound is small, similar priors and likelihoods leave little separation between role posteriors. This motivates conditioning each agent on evidence that distinguishes its intended role. Action stability additionally requires a shared stochastic decision kernel or a sufficient utility margin (Appendix C.2).

4 Agentic Evidence Attribution

In this section, we present Agentic Evidence Attribution (AEA), an analytical and practical framework for correcting agentic misalignment. We first relate evidence to local action error in Section 4.1, then study posterior contraction under a linear-Gaussian model in Section 4.2. Finally, Section 4.3 proposes scalable instantiations of AEA using learned evidence models for workflow alignment.

4.1 An Analytical Framework

We analyze the *extraction of role-specific evidence* through a local action-selection model. The analysis uses the information asymmetry between the runtime evidence Y_t from partial observation and the full computational history $\tau_{<t}$.

Let $E_t = \mathcal{F}(\tau_{<t})$ be additional evidence derived from the trajectory history using a function $\mathcal{F}$. In this model, evidence transforms from Y_t to $Y_t' = (Y_t, E_t)$. We analyze the theoretical limits of this intervention using information theory. Let $L^* = \pi_i^*(s_t; \boldsymbol{\theta}_i)$ be the discrete label of the optimal action, with a fixed rule for resolving ties.

Theorem 4.1 (Lower Bound on Action Error). *Assume $|\mathcal{A}_i| = M \ge 2$. For any decision rule $\hat{a}(Y)$, let $P_e = P(\hat{a}(Y) \ne L^*)$. Fano's inequality gives*

$$P_e \ge \frac{H(L^*) - I(L^*; Y) - \log 2}{\log M}, \tag{4}$$

where H denotes entropy and I denotes mutual information. If L^* is uniform, $H(L^*) = \log M$.

The bound links uncertainty about the role-optimal action to unavoidable action errors. A decisive error further requires the counterfactual workflow failure condition in Section 3. We next characterize when additional evidence can improve the best achievable action accuracy.

Corollary 4.2 (Information Gain Condition). *Let $P_e^*(Y) = \inf_g P(g(Y) \ne L^*)$ be the Bayes-optimal action error. A necessary condition for $P_e^*(Y_t, E_t) < P_e^*(Y_t)$ is*

$$I(L^*; E_t \mid Y_t) > 0. \tag{5}$$

If this conditional mutual information is zero, the optimal-action posterior and Bayes-optimal error are unchanged. This includes deterministic feedback $E_t = \mathcal{F}(Y_t)$ computed entirely from the agent's existing context. A bounded model can still benefit from reorganizing such information. The condition identifies information needed to improve the best achievable decision rule.

AEA therefore seeks evidence about role-optimal actions that is missing from each agent's partial context. The complete trace can contain such information even when the runtime prompt does not. The quality of the extraction function $\mathcal{F}$ determines which of these signals reach the agent.

4.2 Mechanism of Posterior Contraction

We now instantiate the latent role as a continuous utility weight, instead of the finite type in Section 3. Let $\phi(s, a) \in \mathbb{R}^d$ be a feature vector of the state-action pair and let $\mathbf{w} \in \mathbb{R}^d$ represent the latent role, such that $\mathcal{U}(s, a; \mathbf{w}) = \mathbf{w}^\top \phi(s, a)$.

We model the agent's prior as $\mathbf{w} \sim \mathcal{N}(\boldsymbol{\mu}_0, \Sigma_0)$ and its posterior after baseline evidence as $\mathbf{w} \mid Y \sim \mathcal{N}(\boldsymbol{\mu}_Y, \Sigma_Y)$. Additional evidence follows a linear observation model

$$E_t = \mathbf{H}\mathbf{w} + \boldsymbol{\eta}, \qquad \boldsymbol{\eta} \sim \mathcal{N}(0, \mathbf{R}), \tag{6}$$

where $\mathbf{H}$ is fixed, $\boldsymbol{\eta}$ is independent of $(\mathbf{w}, Y)$, and $\Sigma_Y, \mathbf{R} \succ 0$. Thus, E_t is conditionally independent of Y given $\mathbf{w}$. These assumptions specify an idealized evidence channel for the contraction analysis.

Theorem 4.3 (Covariance Contraction under Linear-Gaussian Evidence)**.** *Under these linear-Gaussian assumptions, the posterior covariance after observing $Y' = (Y, E_t)$ satisfies $\Sigma_{Y'} \preceq \Sigma_Y$. Its precision is*

$$\Sigma_{Y'}^{-1} = \Sigma_Y^{-1} + \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H}. \tag{7}$$

This update reduces variance along evidence-constrained directions. To connect contraction to stochastic action errors, we analyze a policy that samples utility weights from its posterior and maximizes sampled utility. This complements the expected-utility policy in Definition 3.1, whose linear-utility decisions depend on the posterior mean.

Corollary 4.4 (Sufficient Conditions for Reliability Improvement)**.** *In the setting of Theorem 4.3, suppose a_1 is the true optimal action among a_1, a_2 and $\delta = \phi(s, a_1) - \phi(s, a_2) \neq 0$. A policy samples utility weights from its posterior and selects the action with greater utility. If $\mu_{D'} = \boldsymbol{\mu}_{Y'}^\top \delta \geq \mu_D = \boldsymbol{\mu}_Y^\top \delta > 0$, then its conditional error satisfies $P_e(Y') \leq P_e(Y)$.*

Covariance contraction and a nondecreasing positive margin are sufficient to improve or maintain reliability under this stochastic policy. We examine the corresponding behavioral implications through role adherence and output variability. Proofs appear in Appendix C.

4.3 Scalable Instantiation via Learned Evidence

Failure attribution requires reasoning across agents and over time. An error introduced by one role can become an assumption in a later agent's decision, so useful evidence must connect the failure to the responsibilities and interactions that produced it. A general-purpose critic may focus on a locally implausible statement or the final answer without resolving the relevant role deviation. We parameterize $\mathcal{F}_\omega(\tau)$ to extract this role-specific evidence from a completed trace and translate it into a concrete prompt revision, using the two instantiations below.

I. Self-Reflection (Bootstrapping). The simplest instantiation approximates $\mathcal{F}$ using the base model itself, $E \approx \mathcal{M}_{\text{base}}(\tau)$. Self-Reflection reuses the acting model's learned preferences, but gains access to the complete trace. It can produce useful evidence when this broader context reveals an error. When its critique repeats the acting model's role ambiguity, it can instead rationalize the failure.

II. Weak-to-Strong (W2S) Generalization. We introduce a separate evidence model $\mathcal{M}_\omega$, instantiated by a smaller open model, to extract role-specific feedback. It is trained on annotated workflow traces to identify the responsible agent, rate collaboration quality, and propose a correction. Reinforcement learning uses attribution and feedback-quality rewards as training signals for useful evidence. In our experiments, this evidence specifies a replacement for the identified agent's system prompt, followed by one full workflow restart. The local analysis studies action selection, while the restart permits new states and observations. We evaluate this combined procedure through workflow outcomes.

5 Experiments

5.1 Agentic Trace Generation

To learn generalizable agentic evidence from MAS traces, we construct agentic trace reasoning data from four benchmarks (Appendix Table 11), `GAIA` (Mialon et al., 2023), `AssistantBench` (Yoran et al., 2024), `LiveBench` (White et al., 2025), and `Who&When` (Zhang et al., 2025c). We convert tasks from the first three into multi-agent workflows. For `Who&When`, training uses only the Algorithm-Generated split. The Hand-Crafted split is reserved exclusively for evaluation, with its

Table 1: Accuracy (%) on seven benchmarks. Self-Reflection uses the workflow model for attribution. Weak-to-Strong uses AEA-4B. Results are averaged over three random seeds. Best AEA variant per model and benchmark in **bold**.

Setting	HumanEval	DataBench	Chemistry	Physics	AIME24	AIME25	AIME26
Claude 3 Haiku							
Single Agent	$61.6_{\pm 0.00}$	$9.3_{\pm 0.78}$	$31.7_{\pm 4.88}$	$23.5_{\pm 2.94}$	$0.0_{\pm 0.00}$	$3.3_{\pm 5.77}$	$3.3_{\pm 0.00}$
Multi Agent	$78.5_{\pm 1.22}$	$11.7_{\pm 0.67}$	$33.6_{\pm 2.82}$	$36.7_{\pm 3.40}$	$10.0_{\pm 5.77}$	$3.3_{\pm 3.33}$	$6.7_{\pm 5.77}$
Self-Reflection	$72.0_{\pm 0.61}$	$\mathbf{27.3}_{\pm 0.67}$	$31.7_{\pm 2.44}$	$41.2_{\pm 5.09}$	$16.7_{\pm 6.67}$	$13.3_{\pm 3.33}$	$10.0_{\pm 5.77}$
Weak-to-Strong	$\mathbf{79.3}_{\pm 0.61}$	$23.9_{\pm 0.45}$	$\mathbf{36.6}_{\pm 4.22}$	$\mathbf{58.8}_{\pm 2.94}$	$\mathbf{20.0}_{\pm 3.33}$	$\mathbf{16.7}_{\pm 0.00}$	$\mathbf{13.3}_{\pm 3.33}$
Claude 3.5 Sonnet							
Single Agent	$82.3_{\pm 0.00}$	$27.3_{\pm 0.39}$	$47.6_{\pm 1.41}$	$35.3_{\pm 2.94}$	$16.7_{\pm 5.77}$	$6.7_{\pm 0.00}$	$10.0_{\pm 3.33}$
Multi Agent	$84.8_{\pm 1.83}$	$32.5_{\pm 0.22}$	$51.2_{\pm 2.44}$	$58.8_{\pm 5.09}$	$23.3_{\pm 3.33}$	$16.7_{\pm 0.00}$	$16.7_{\pm 3.33}$
Self-Reflection	$\mathbf{89.0}_{\pm 0.00}$	$29.7_{\pm 0.59}$	$54.9_{\pm 2.82}$	$60.2_{\pm 3.40}$	$23.3_{\pm 3.33}$	$\mathbf{26.7}_{\pm 3.33}$	$20.0_{\pm 6.67}$
Weak-to-Strong	$86.6_{\pm 2.11}$	$\mathbf{35.0}_{\pm 0.67}$	$\mathbf{56.1}_{\pm 2.44}$	$\mathbf{61.8}_{\pm 0.00}$	$\mathbf{30.0}_{\pm 0.00}$	$\mathbf{26.7}_{\pm 0.00}$	$\mathbf{26.7}_{\pm 0.00}$
Claude 3.7 Sonnet							
Single Agent	$84.8_{\pm 0.61}$	$15.2_{\pm 0.78}$	$61.5_{\pm 2.82}$	$58.8_{\pm 5.09}$	$23.3_{\pm 3.33}$	$13.3_{\pm 3.33}$	$16.7_{\pm 0.00}$
Multi Agent	$85.4_{\pm 1.83}$	$10.5_{\pm 1.35}$	$61.0_{\pm 4.22}$	$64.7_{\pm 0.00}$	$40.0_{\pm 0.00}$	$26.7_{\pm 0.00}$	$26.7_{\pm 5.77}$
Self-Reflection	$89.6_{\pm 1.06}$	$29.3_{\pm 0.22}$	$63.4_{\pm 2.44}$	$67.6_{\pm 5.09}$	$16.7_{\pm 3.33}$	$16.7_{\pm 5.77}$	$23.3_{\pm 3.33}$
Weak-to-Strong	$\mathbf{90.9}_{\pm 0.61}$	$\mathbf{35.6}_{\pm 0.22}$	$\mathbf{71.4}_{\pm 2.82}$	$\mathbf{68.4}_{\pm 6.12}$	$\mathbf{40.0}_{\pm 3.33}$	$\mathbf{46.6}_{\pm 3.33}$	$\mathbf{40.0}_{\pm 3.33}$
Claude Opus 4.8							
Single Agent	$97.0_{\pm 0.61}$	$42.0_{\pm 0.67}$	$85.4_{\pm 4.22}$	$88.2_{\pm 2.94}$	$96.7_{\pm 3.33}$	$96.7_{\pm 5.77}$	$90.0_{\pm 3.33}$
Multi Agent	$96.3_{\pm 1.61}$	$41.2_{\pm 0.39}$	$82.9_{\pm 2.44}$	$85.3_{\pm 0.00}$	$96.7_{\pm 3.33}$	$96.7_{\pm 5.77}$	$83.3_{\pm 0.00}$
Self-Reflection	$97.0_{\pm 0.00}$	$40.1_{\pm 0.67}$	$80.5_{\pm 0.00}$	$82.4_{\pm 2.94}$	$96.7_{\pm 5.77}$	$\mathbf{100.0}_{\pm 0.00}$	$83.3_{\pm 0.00}$
Weak-to-Strong	$\mathbf{98.2}_{\pm 0.61}$	$\mathbf{44.7}_{\pm 1.35}$	$\mathbf{87.8}_{\pm 4.22}$	$\mathbf{91.2}_{\pm 2.94}$	$\mathbf{100.0}_{\pm 0.00}$	$\mathbf{100.0}_{\pm 0.00}$	$\mathbf{96.7}_{\pm 3.33}$
GPT-4o-mini							
Single Agent	$91.5_{\pm 0.61}$	$26.2_{\pm 0.59}$	$61.0_{\pm 2.44}$	$67.6_{\pm 0.00}$	$13.3_{\pm 5.77}$	$13.3_{\pm 3.33}$	$13.3_{\pm 3.33}$
Multi Agent	$92.0_{\pm 1.61}$	$28.8_{\pm 1.17}$	$58.5_{\pm 2.44}$	$61.8_{\pm 5.09}$	$16.7_{\pm 3.33}$	$26.7_{\pm 3.33}$	$20.0_{\pm 3.33}$
Self-Reflection	$\mathbf{93.3}_{\pm 0.00}$	$27.3_{\pm 1.40}$	$55.9_{\pm 2.44}$	$\mathbf{61.8}_{\pm 0.00}$	$6.7_{\pm 5.77}$	$20.0_{\pm 0.00}$	$16.7_{\pm 3.33}$
Weak-to-Strong	$92.2_{\pm 0.70}$	$\mathbf{29.7}_{\pm 1.25}$	$\mathbf{65.9}_{\pm 2.44}$	$\mathbf{61.8}_{\pm 0.00}$	$\mathbf{23.3}_{\pm 5.77}$	$\mathbf{30.0}_{\pm 0.00}$	$\mathbf{26.7}_{\pm 3.33}$
GPT-4o							
Single Agent	$95.7_{\pm 2.11}$	$27.3_{\pm 0.39}$	$73.2_{\pm 4.22}$	$82.4_{\pm 0.00}$	$16.7_{\pm 0.00}$	$10.0_{\pm 0.00}$	$20.0_{\pm 5.77}$
Multi Agent	$91.5_{\pm 0.61}$	$30.7_{\pm 0.39}$	$70.7_{\pm 1.88}$	$61.8_{\pm 2.94}$	$23.3_{\pm 1.33}$	$33.3_{\pm 0.00}$	$30.0_{\pm 3.33}$
Self-Reflection	$94.5_{\pm 1.22}$	$27.7_{\pm 0.81}$	$65.9_{\pm 2.44}$	$71.0_{\pm 2.94}$	$30.0_{\pm 3.33}$	$26.7_{\pm 3.33}$	$26.7_{\pm 3.33}$
Weak-to-Strong	$\mathbf{96.3}_{\pm 0.61}$	$\mathbf{35.0}_{\pm 0.39}$	$\mathbf{73.2}_{\pm 2.44}$	$\mathbf{76.5}_{\pm 2.94}$	$\mathbf{40.0}_{\pm 3.33}$	$\mathbf{33.3}_{\pm 3.33}$	$\mathbf{40.0}_{\pm 0.00}$
GPT-5.5							
Single Agent	$95.7_{\pm 1.22}$	$37.0_{\pm 0.78}$	$80.5_{\pm 2.44}$	$82.4_{\pm 5.09}$	$86.7_{\pm 3.33}$	$83.3_{\pm 0.00}$	$90.0_{\pm 5.77}$
Multi Agent	$95.1_{\pm 1.06}$	$38.5_{\pm 0.39}$	$78.0_{\pm 4.88}$	$79.4_{\pm 0.00}$	$83.3_{\pm 3.33}$	$80.0_{\pm 3.33}$	$90.0_{\pm 0.00}$
Self-Reflection	$95.7_{\pm 0.61}$	$37.4_{\pm 0.67}$	$75.6_{\pm 4.22}$	$76.5_{\pm 2.94}$	$83.3_{\pm 3.33}$	$76.7_{\pm 0.00}$	$83.3_{\pm 0.00}$
Weak-to-Strong	$\mathbf{97.0}_{\pm 0.00}$	$\mathbf{40.9}_{\pm 0.39}$	$\mathbf{82.9}_{\pm 2.44}$	$\mathbf{82.4}_{\pm 2.94}$	$\mathbf{86.7}_{\pm 0.00}$	$\mathbf{80.0}_{\pm 5.77}$	$\mathbf{96.7}_{\pm 3.33}$

traces and annotations fully held out from AEA training. The curated set contains 1,531 training traces. Screening removes identified evaluation overlap. Thus the evaluation set is entirely held-out. Appendix K reports the data versions and overlap checks.

For each task, we generate agentic trajectories in JSON format. We first use Claude Opus 4 (Anthropic, 2025) to produce initial pseudo-annotations of the failing agent, critical error step, and candidate repair. Similar to Zhang et al. (2025c) and Chen et al. (2026), these annotations are reviewed by a team of human experts, who select the agent and step by majority vote and edit the reason and prompt revision. These are human judgments of failure locations from recorded traces, without counterfactual execution. The final traces and structured evidence labels are used to train AEA. Section 5.3 describes the MAS implementation.

5.2 Evaluation

We evaluate workflow accuracy on eight benchmarks. `HumanEval` (Chen et al., 2021) has 164 coding tasks; `DataBench` (Grijalba et al., 2024) has 257 data-analysis tasks, with only data links given to agents. `SciBench` (Wang et al., 2024) contributes 41 `Chemistry` and 34 `Physics` tasks. `AIME24` (HuggingFace, 2025a), `AIME25` (HuggingFace, 2025b), and `AIME26` each have 30 problems; `HLE` (Phan et al., 2025) has 2,500 questions. We execute generated code against `HumanEval` tests, use GPT-4o (OpenAI, 2024) for `DataBench` and `SciBench`, numerical matching for AIME,

Table 2: Accuracy (%) on AIME26 and HLE. Results average three seeds over 30 AIME26 problems and 2,500 HLE questions. Correction uses one restart. Gemma is an external critic.

Setting	GPT-5.5		Claude Opus 4.8	
	AIME26	HLE	AIME26	HLE
Original workflow	$90.0_{\pm 0.00}$	$44.28_{\pm 0.50}$	$83.3_{\pm 0.00}$	$48.56_{\pm 0.39}$
Self-Reflection	$83.3_{\pm 0.00}$	$43.76_{\pm 0.67}$	$83.3_{\pm 0.00}$	$46.92_{\pm 0.72}$
Naive Retry	$90.0_{\pm 3.33}$	$44.88_{\pm 0.20}$	$86.7_{\pm 3.33}$	$49.08_{\pm 0.21}$
Gemma-4-E4B-it critic	$90.0_{\pm 0.00}$	$46.72_{\pm 0.16}$	$90.0_{\pm 0.00}$	$51.24_{\pm 0.25}$
Weak-to-Strong (AEA-4B)	$\mathbf{96.7}_{\pm 3.33}$	$\mathbf{51.04}_{\pm 0.30}$	$\mathbf{96.7}_{\pm 3.33}$	$\mathbf{55.80}_{\pm 0.67}$

Table 3: Failure attribution accuracy (%) on the held-out Hand-Crafted split of Who&When (Zhang et al., 2025c). Best per protocol in **bold**. Our alignment experiments use only All at Once, following Zhang et al. (2025c).

Model	All at Once		Step by Step	
	Agent Acc.	Step Acc.	Agent Acc.	Step Acc.
Llama-3.1-8B	$9.52_{\pm 2.10}$	$2.38_{\pm 2.10}$	$24.60_{\pm 0.79}$	$13.49_{\pm 0.79}$
Llama-3.1-70B	$48.41_{\pm 0.79}$	$16.67_{\pm 0.79}$	$30.95_{\pm 2.38}$	$19.84_{\pm 0.00}$
Qwen2.5-7B	$10.32_{\pm 2.10}$	$5.56_{\pm 0.79}$	$11.11_{\pm 0.79}$	$6.35_{\pm 1.59}$
Qwen2.5-72B	$55.56_{\pm 1.59}$	$10.32_{\pm 0.79}$	$20.63_{\pm 0.79}$	$16.67_{\pm 1.37}$
Qwen3-8B	$31.75_{\pm 1.37}$	$5.56_{\pm 0.79}$	$20.63_{\pm 2.75}$	$10.32_{\pm 0.00}$
Qwen3-32B	$53.97_{\pm 0.00}$	$15.87_{\pm 0.79}$	$38.10_{\pm 0.79}$	$23.02_{\pm 1.59}$
Claude 3.5 Haiku	$12.70_{\pm 2.10}$	$7.94_{\pm 2.10}$	$21.43_{\pm 0.79}$	$16.67_{\pm 1.37}$
Claude 3.7 Sonnet	$58.70_{\pm 0.79}$	$24.60_{\pm 0.79}$	$46.03_{\pm 2.86}$	$\mathbf{30.95}_{\pm 2.10}$
GPT-4o-mini	$35.71_{\pm 1.37}$	$19.84_{\pm 0.00}$	$30.95_{\pm 1.37}$	$19.05_{\pm 0.79}$
GPT-4o	$50.00_{\pm 1.37}$	$30.95_{\pm 0.10}$	$33.33_{\pm 2.10}$	$26.43_{\pm 0.46}$
AEA-4B	$\mathbf{60.79}_{\pm 1.21}$	$\mathbf{32.70}_{\pm 2.79}$	$\mathbf{48.41}_{\pm 2.86}$	$28.87_{\pm 2.92}$

and the official HLE evaluator. Reference answers and judge outputs are used only for scoring, never for workflow execution or correction. Accuracy is averaged over three seeds. Failure attribution uses the held-out Hand-Crafted split of Who&When and TraceElephant (Chen et al., 2026). Appendices H and K detail the prompts and protocols. Appendix K reports a pooled human audit of final-answer judgments, with agreement on 98 of 100 checked outputs.

5.3 Implementation Details

Multi-Agent System. We implement MAS using CaptainAgent from AG2 (Wu et al., 2024). CaptainAgent assigns roles and coordinates turn-by-turn interactions through shared memory and tools. Each agent observes its role description and the partial system state, then acts to update the workflow. AEA collects the complete trace, revises only the identified agent's system prompt, and restarts the full workflow once. The predicted agent selects the prompt to revise, while the step localizes the diagnosis. The first trace informs the revision but is not carried into the restart as history. Results include incorrect attributions. Naive Retry uses unchanged prompts, while the Gemma critic receives the same full trace and evidence schema under the one-restart protocol.

Weak-to-Strong Training. We train AEA-4B with Qwen3-4B as the backbone through a two-stage procedure. In the first stage, we apply a supervised warm start on RM-R1 (Chen et al., 2025). In the second stage, we apply GRPO on annotated workflow traces (Shao et al., 2024). The reward has four weighted components, agent identification (40%), rating agreement (30%), prompt revision validity (20%), and explanation completeness (10%). Agent identification checks exact agreement with the annotated name, and prompt revision validity checks for a non-trivial revision. Invalid JSON receives -0.8 and valid rewards are clipped to $[-1, 1]$. Note that Step Accuracy is evaluated separately and is not a reward component. Final task correctness is evaluated after correction. Appendix D gives the objective and training details.

5.4 Main Results

Inefficient agentic scaling in automated workflows. Table 1 shows that transitioning from a Single Agent to a Multi-Agent workflow can improve performance, particularly for smaller base models. For

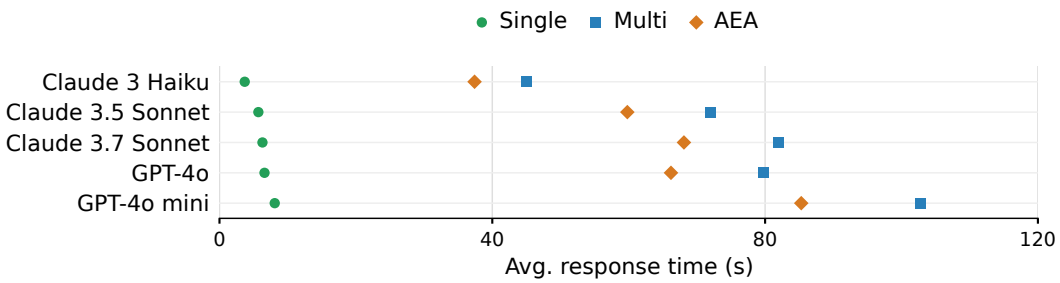


Figure 2: Average response time across agent configurations. Multi-agent workflows take approximately 12–13 times as long as the single-agent baselines.

instance, Claude 3 Haiku improves on `HumanEval` from 61.6% to 78.5% and on `Physics` from 23.5% to 36.7%. However, this agentic test-time scaling comes at a cost. Response times increase by a factor of 12 to 13 (Figure 2). The gains also vary across tasks, with diminishing returns on several `Chemistry` settings. Thus, pure scaling cannot effectively resolve the misalignment issue. More cost analysis is shown in Appendix E.4.

AEA improves workflow reliability. In Table 1, Weak-to-Strong AEA generally improves accuracy over the unaligned workflow across model families and benchmarks (Table 1). On the 556-task paired subset, the two strong workflow models gain 2.79 percentage points in task-weighted accuracy with a 95% CI of $[1.38, 4.20]$ (Appendix K). Appendix E.2 examines role adherence, and Appendices I and J present case studies.

Correction with strong workflow models. On all 2,500 `HLE` questions, AEA raises GPT-5.5 accuracy from 44.28% to 51.04% (Table 2). Under the same restart budget, it solves an average of 108 more questions per seed than the Gemma critic and 154 more than Naive Retry. The gain over the critic is 4.32 percentage points with a paired 95% CI of $[2.78, 5.86]$. Across GPT-5.5 and Claude Opus 4.8 in Table 1, AEA improves in 13 of 14 settings and ties in one, reaching 96.7% on `AIME26` with both models. Appendix K reports paired analyzes and distinguishes benchmark-macro from task-weighted gains.

Weak-to-Strong and Self-Reflection. A key finding is the distinct behavior of Weak-to-Strong generalization compared with Self-Reflection. As shown in Figure 3a, the rating distribution for Self-Reflection is heavily skewed towards high scores (4 and 5), even when the system fails. In contrast, Weak-to-Strong AEA concentrates more ratings at 2 and 3. These results show that the informativeness of alignment methods is crucial for the alignment quality, which aligns with our insight in Section 4.1. We further examine role adherence in Section 5.6.

Correction outcomes with specialized evidence. Figure 3c summarizes recorded correction outcomes across `HumanEval`, `DataBench`, `Chemistry`, `Physics`, `AIME24`, and `AIME25`. The displayed counts for **Fixed** (Incorrect → Correct), **Preserved** (Correct → Correct), and **Regressed** (Correct → Incorrect) are 36, 30, and 27 for Weak-to-Strong AEA, compared with 18, 12, and 51 for Self-Reflection. Among the recorded outcomes, Weak-to-Strong AEA has more fixes and fewer regressions than Self-Reflection, which also aligns with our theoretical analysis.

5.5 Automated Failure Attribution

We evaluate AEA's ability to identify which agent and which step in a workflow is failing on `Who&When` (Table 3). Agent and Step Accuracy are separate exact-match rates for the annotated agent name and error-step index, respectively. In the "All at Once" setting, our specialized AEA-4B achieves the best Agent Accuracy (60.79%) and Step Accuracy (32.70%) among the compared models. Our method also shows competitive performance in the "Step by Step" setting.

Effect of specialized training. With the Gemma checkpoint and inference budget fixed, AEA training raises Agent Accuracy from 69.4% to 80.0% on 85 `TraceElephant` traces. On 60 `AIME24/25` tasks under two workflow structures, final correctness rises from 63.3% to 75.0% over 120 outcomes per seed. The respective gains have paired 95% CIs of $[2.35, 20.00]$ and $[4.17, 19.17]$ percentage points (Appendix K).

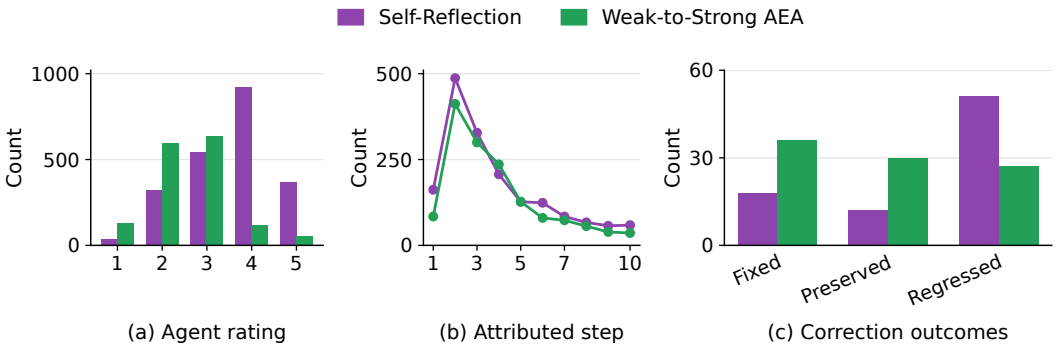


Figure 3: Self-Reflection and Weak-to-Strong AEA. **(a, b)** Collaboration ratings and attributed steps. **(c)** Recorded Fixed, Preserved, and Regressed outcomes across `HumanEval`, `DataBench`, `Chemistry`, `Physics`, `AIME24`, and `AIME25`.

5.6 Empirical Analysis of Evidence

To test the effectiveness of the evidence, we compare four settings under the same GPT-4o workflow, Naive Retry (no new evidence), Generic Feedback (a role-agnostic message), Self-Reflection, and AEA-4B (an external evidence model). As shown in Table 4, AEA-4B achieves an average task-error reduction of 0.071, compared with 0.011 for Generic Feedback, with the largest gain on `AIME24`. Self-Reflection regresses on `DataBench` and `Chemistry`. Appendices E.1 and E provide further discussion and output-variance analyses.

Role adherence. In an `AIME24` workflow with separate Solver, Verifier, and Reporter roles, Weak-to-Strong AEA raises role action accuracy from 34.2% to 59.6% and reduces functional overlap from 58.3% to 29.5% (Appendix E.2). Fixed-rubric judgments agree with human labels on 48 of 50 audited examples. AEA therefore improves adherence to the assigned roles in this setting.

Table 4: Evidence gradient on GPT-4o. ΔP_e is the average task-error reduction across four benchmarks. Naive Retry adds no evidence, Generic Feedback is role-agnostic, Self-Reflection uses the same model, and AEA-4B supplies specialized external evidence.

Setting	ΔP_e	`HumanEval`	`DataBench`	`Chemistry`	`AIME24`
Multi-Agent (original)	$0.000_{\pm 0.000}$	$91.5_{\pm 0.61}$	$30.7_{\pm 0.39}$	$70.7_{\pm 1.88}$	$23.3_{\pm 1.33}$
Naive Retry	$0.005_{\pm 0.005}$	$92.1_{\pm 1.61}$	$31.5_{\pm 0.39}$	$71.2_{\pm 2.82}$	$23.3_{\pm 2.09}$
Generic Feedback	$0.011_{\pm 0.009}$	$93.0_{\pm 1.27}$	$32.3_{\pm 1.17}$	$72.0_{\pm 1.73}$	$23.3_{\pm 1.92}$
Self-Reflection	$0.005_{\pm 0.017}$	$94.5_{\pm 1.22}$	$27.7_{\pm 0.81}$	$65.9_{\pm 2.44}$	$30.0_{\pm 3.33}$
AEA-4B (W2S)	$\mathbf{0.071}_{\pm 0.011}$	$\mathbf{96.3}_{\pm 0.61}$	$\mathbf{35.0}_{\pm 0.39}$	$\mathbf{73.2}_{\pm 2.44}$	$\mathbf{40.0}_{\pm 3.33}$

6 Conclusion

In this work, we study agentic misalignment in automated workflows through Bayesian inference over latent roles. Our analysis characterizes when role beliefs remain insufficiently separated and gives conditions under which additional evidence can improve decisions. To address this problem, we introduce Agentic Evidence Attribution (AEA), an alignment framework for incorporating structured, role-specific evidence into workflow correction. By reasoning across agents and over time, AEA connects observed failures to the responsibilities and interactions that produced them, enabling correction of the identified agent. Our empirical evaluation highlights two main conclusions. First, evidence-conditioned alignment (normally needs human efforts) improves multi-agent reliability, with gains beyond retry and generic-critic baselines under the same restart budget. Second, weak-to-strong generalization shows that small, specialized evidence models can improve stronger automated workflows through learned failure attribution, which is more suitable for broader applications. These findings support a path toward scalable oversight that couples strong reasoning agents with specialized evidence models. Future work can extend this approach to larger workflows and study how evidence models adapt as agent capabilities and collaboration patterns change.

AI USE STATEMENT

We used large language models only to assist with drafting, editing, and checking the mathematical exposition. These tools were not involved in the main ideation or implementation. This assistance included improving clarity, organizing explanations, and identifying inconsistencies in notation and presentation. The authors are responsible for the methods, results, and conclusions.

REPRODUCIBILITY STATEMENT

The appendix provides dataset statistics and training splits, the evidence-model training procedure and reward definition, evidence and evaluator prompts, and the workflow correction protocol. It also documents overlap checks, attribution metrics, and paired statistical analyses. We will provide the anonymous code in the supplementary materials.

REFERENCES

Renat Aksitov, Sobhan Miryoosefi, Zonglin Li, Daliang Li, Sheila Babayan, Kavya Kopparapu, Zachary Fisher, Ruiqi Guo, Sushant Prakash, Pranesh Srinivasan, et al. Rest meets react: Self-improvement for multi-step reasoning llm agent. In *ICLR 2024 Workshop on Large Language Model (LLM) Agents*, 2024.

Anthropic. Introducing Claude 4. https://www.anthropic.com/news/claude-4, May 2025.

Yoshua Bengio, Michael Cohen, Damiano Fornasiere, Joumana Ghosn, Pietro Greiner, Matt MacDermott, Sören Mindermann, Adam Oberman, Jesse Richardson, Oliver Richardson, et al. Superintelligent agents pose catastrophic risks: Can scientist ai offer a safer path? *arXiv preprint arXiv:2502.15657*, 2025.

Xiaohe Bo, Zeyu Zhang, Quanyu Dai, Xueyang Feng, Lei Wang, Rui Li, Xu Chen, and Ji-Rong Wen. Reflective multi-agent collaboration based on large language models. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv preprint arXiv:2407.21787*, 2024.

Mert Cemri, Melissa Z Pan, Shuyi Yang, Lakshya A Agrawal, Bhavya Chopra, Rishabh Tiwari, Kurt Keutzer, Aditya Parameswaran, Dan Klein, Kannan Ramchandran, et al. Why do multi-agent llm systems fail? *arXiv preprint arXiv:2503.13657*, 2025.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé, Jared Kaplan, Harrison Edwards, Yura Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mo Bavarian, Clemens Winter, Phil Tillet, Felipe Petroski Such, David W. Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William H. Guss, Alex Nichol, Igor Babuschkin, Suchir Balaji, Shantanu Jain, Andrew Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew M. Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. *ArXiv*, abs/2107.03374, 2021.

Mengzhuo Chen, Junjie Wang, Fangwen Mu, Yawen Wang, Zhe Liu, Huanxiang Feng, and Qing Wang. Seeing the whole elephant: A benchmark for failure attribution in LLM-based multi-agent systems. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 19888–19905. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.912.

Xiusi Chen, Gaotang Li, Ziqi Wang, Bowen Jin, Cheng Qian, Yu Wang, Hongru Wang, Yu Zhang, Denghui Zhang, Tong Zhang, et al. Rm-r1: Reward modeling as reasoning. *arXiv preprint arXiv:2505.02387*, 2025.

Lutfi Eren Erdogan, Hiroki Furuta, Sehoon Kim, Nicholas Lee, Suhong Moon, Gopala Anumanchipalli, Kurt Keutzer, and Amir Gholami. Plan-and-act: Improving planning of agents for long-horizon tasks. In *Forty-second International Conference on Machine Learning*, 2025.

Jorge Osés Grijalba, L Alfonso Urena Lopez, Eugenio Martínez-Cámara, and Jose Camacho-Collados. Question answering over tabular data with databench: A large-scale empirical evaluation of llms. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pp. 13471–13488, 2024.

Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V Chawla, Olaf Wiest, and Xiangliang Zhang. Large language model based multi-agents: A survey of progress and challenges. In *IJCAI*, 2024.

Shanshan Han, Qifan Zhang, Yuhang Yao, Weizhao Jin, and Zhaozhuo Xu. Llm multi-agent systems: Challenges and open problems. *arXiv preprint arXiv:2402.03578*, 2024.

HuggingFace. Aime 2024 benchmark (huggingfaceh4/aime_2024). https://huggingface.co/datasets/HuggingFaceH4/aime_2024, 2025a.

HuggingFace. Aime 2025 benchmark (opencompass/aime2025). https://huggingface.co/datasets/opencompass/AIME2025, 2025b.

Allison Sihan Jia, Daniel Huang, Nikhil Vytla, Seung Won Wilson Yoo, Nirvika Choudhury, Shayak Sen, John C. Mitchell, and Anupam Datta. What is your agent's GPA? a framework for evaluating agent goal-plan-action alignment. *arXiv preprint arXiv:2510.08847*, 2025.

Prathyusha Jwalapuram, Hehai Lin, Chuyuan Li, Fangkai Jiao, Sudong Wang, Yifei Ming, Zixuan Ke, Chengwei Qin, Giuseppe Carenini, and Shafiq Joty. The illusion of multi-agent advantage. *arXiv preprint arXiv:2606.13003*, 2026.

Jacek Karwowski, Oliver Hayman, Xingjian Bai, Klaus Kiendlhofer, Charlie Griffin, and Joar Max Viktor Skalse. Goodhart's law in reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024.

Yubin Kim, Ken Gu, Chanwoo Park, Chunjong Park, Samuel Schmidgall, A Ali Heydari, Yao Yan, Zhihan Zhang, Yuchen Zhuang, Mark Malhotra, et al. Towards a science of scaling agent systems. *arXiv preprint arXiv:2512.08296*, 2025.

Tomasz Korbak, Ethan Perez, and Christopher Buckley. Rl with kl penalties is better viewed as bayesian inference. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 1083–1091, 2022.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, Lester James Validad Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, et al. Rewardbench: Evaluating reward models for language modeling. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 1755–1797, 2025.

Zelong Li, Shuyuan Xu, Kai Mei, Wenyue Hua, Balaji Rama, Om Raheja, Hao Wang, He Zhu, and Yongfeng Zhang. Autoflow: Automated workflow generation for large language model agents. *arXiv preprint arXiv:2407.12821*, 2024.

Yantao Liu, Zijun Yao, Rui Min, Yixin Cao, Lei Hou, and Juanzi Li. Rm-bench: Benchmarking reward models of language models with subtlety and style. In *The Thirteenth International Conference on Learning Representations*, 2025.

Aengus Lynch, Benjamin Wright, Caleb Larson, Stuart J Ritchie, Soren Mindermann, Evan Hubinger, Ethan Perez, and Kevin Troy. Agentic misalignment: How llms could be insider threats. *arXiv preprint arXiv:2510.05179*, 2025.

Grégoire Mialon, Clémentine Fourrier, Thomas Wolf, Yann LeCun, and Thomas Scialom. Gaia: a benchmark for general ai assistants. In *The Twelfth International Conference on Learning Representations*, 2023.

OpenAI. Hello GPT-4o. https://openai.com/index/hello-gpt-4o/, May 2024.

Theodore Papamarkou, Pierre Alquier, Matthias Bauer, Wray Buntine, Andrew Davison, Gintare Karolina Dziugaite, Maurizio Filippone, Andrew Y. K. Foong, Vincent Fortuin, Dimitris Fouskakis, Jes Frellsen, Eyke Hüllermeier, Theofanis Karaletsos, Mohammad Emtiyaz Khan, Nikita Kotelevskii, Salem Lahlou, Yingzhen Li, Fang Liu, Clare Lyle, Thomas Möllenhoff, Konstantina Palla, Maxim Panov, Yusuf Sale, Kajetan Schweighofer, Artem Shelmanov, Siddharth Swaroop, Martin Trapp, Willem Waegeman, Andrew Gordon Wilson, and Alexey Zaytsev. Position: agentic AI orchestration should be Bayes-consistent. *arXiv preprint arXiv:2605.00742*, 2026.

Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Mohamed Shaaban, John Ling, Sean Shi, et al. Humanity's last exam. *arXiv preprint arXiv:2501.14249*, 2025.

Aske Plaat, Annie Wong, Suzan Verberne, Joost Broekens, and Niki Van Stein. Multi-step reasoning with large language models, a survey. *ACM Computing Surveys*, 2025.

Sriyash Poddar, Yanming Wan, Hamish Ivison, Abhishek Gupta, and Natasha Jaques. Personalizing reinforcement learning from human feedback with variational preference learning. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, pp. 52516–52544, 2024.

Archiki Prasad, Alexander Koller, Mareike Hartmann, Peter Clark, Ashish Sabharwal, Mohit Bansal, and Tushar Khot. Adapt: As-needed decomposition and planning with language models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 4226–4252, 2024.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, et al. Toolllm: Facilitating large language models to master 16000+ real-world apis. In *ICLR*, 2024.

Roxana Rădulescu, Patrick Mannion, Diederik M Roijers, and Ann Nowé. Multi-objective multi-agent decision making: a utility-based analysis and survey. *Autonomous Agents and Multi-Agent Systems*, 34(1):10, 2020.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Joar Skalse, Nikolaus Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward gaming. *Advances in Neural Information Processing Systems*, 35:9460–9471, 2022.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

Jascha Sohl-Dickstein. The hot mess theory of AI misalignment: More intelligent agents behave less coherently . https://sohl-dickstein.github.io/2023/03/09/coherence.html, 2023.

Matthijs TJ Spaan. Partially observable markov decision processes. In *Reinforcement learning: State-of-the-art*, pp. 387–414. Springer, 2012.

Srinivasan Subramanian, Kazi Aminul Islam, and Md. Abdullah Al Hafiz Khan. Beyond unsafe detection: Counterfactually anchored evidence attribution for multi-turn LLM safety failures. *arXiv preprint arXiv:2609.27773*, 2026.

Miles Wang, Tom Dupré la Tour, Olivia Watkins, Alex Makelov, Ryan A Chi, Samuel Miserendino, Jeffrey Wang, Achyuta Rajaram, Johannes Heidecke, Tejal Patwardhan, et al. Persona features control emergent misalignment. *arXiv preprint arXiv:2506.19823*, 2025.

Xiaoxuan Wang, Ziniu Hu, Pan Lu, Yanqiao Zhu, Jieyu Zhang, Satyen Subramaniam, Arjun R Loomba, Shichang Zhang, Yizhou Sun, and Wei Wang. Scibench: Evaluating college-level scientific problem-solving abilities of large language models. In *International Conference on Machine Learning*, pp. 50622–50649. PMLR, 2024.

Lilian Weng. Reward hacking in reinforcement learning. *lilianweng.github.io*, Nov 2024. URL https://lilianweng.github.io/posts/2024-11-28-reward-hacking/.

Colin White, Samuel Dooley, Manley Roberts, Arka Pal, Benjamin Feuer, Siddhartha Jain, Ravid Shwartz-Ziv, Neel Jain, Khalid Saifullah, Sreemanti Dey, et al. Livebench: A challenging, contamination-limited llm benchmark. In *The Thirteenth International Conference on Learning Representations*, 2025.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, et al. Autogen: Enabling next-gen llm applications via multi-agent conversations. In *First Conference on Language Modeling*, 2024.

Shirley Wu, Parth Sarthi, Shiyu Zhao, Aaron Lee, Herumb Shandilya, Adrian Mladenic Grobelnik, Nurendra Choudhary, Eddie Huang, Karthik Subbian, Linjun Zhang, et al. Optimas: Optimizing compound ai systems with globally aligned local rewards. *arXiv preprint arXiv:2507.03041*, 2025.

Wenqian Ye, Guangtao Zheng, and Aidong Zhang. Rectifying shortcut behaviors in preference-based reward learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Ori Yoran, Samuel Amouyal, Chaitanya Malaviya, Ben Bogin, Ofir Press, and Jonathan Berant. Assistantbench: Can web agents solve realistic and time-consuming tasks? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8938–8968, 2024.

Siyu Yuan, Kaitao Song, Jiangjie Chen, Xu Tan, Yongliang Shen, Kan Ren, Dongsheng Li, and Deqing Yang. Easytool: Enhancing llm-based agents with concise tool instruction. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 2025.

Guibin Zhang, Junhao Wang, Junjie Chen, Wangchunshu Zhou, Kun Wang, and Shuicheng Yan. Agentracer: Who is inducing failure in the llm agentic systems? *arXiv preprint arXiv:2509.03312*, 2025a.

Jiayi Zhang, Jinyu Xiang, Zhaoyang Yu, Fengwei Teng, Xiong-Hui Chen, Jiaqi Chen, Mingchen Zhuge, Xin Cheng, Sirui Hong, Jinlin Wang, et al. Aflow: Automating agentic workflow generation. In *The Thirteenth International Conference on Learning Representations*, 2025b.

Shaokun Zhang, Ming Yin, Jieyu Zhang, Jiale Liu, Zhiguang Han, Jingyang Zhang, Beibin Li, Chi Wang, Huazheng Wang, Yiran Chen, and Qingyun Wu. Which agent causes task failures and when? on automated failure attribution of LLM multi-agent systems. In *Forty-second International Conference on Machine Learning*, 2025c.

APPENDIX

A BROADER IMPACT

This paper studies the reliability of automated multi-agent systems through role adherence and failure correction. Understanding how local role deviations affect a shared objective can help improve oversight in scientific research, software engineering, and decision support. AEA provides structured explanations and targeted prompt revisions that make workflow failures easier to diagnose and correct. These capabilities can support safer automation, while broader safety also depends on the goals, tools, and deployment environment of the workflow. Methods that improve coordination could also be applied to harmful workflows. Our evaluation concerns cooperative task completion, and responsible deployment requires safeguards appropriate to the application.

B LIMITATIONS

AEA studies role misspecification through one-role prompt correction. Its effectiveness depends on the evidence model, the acting model's ability to use feedback, and the workflow's underlying capabilities. Human labels identify critical errors from recorded traces, while formal decisiveness requires a successful counterfactual intervention. Our comparisons evaluate the combined effects of evidence validity, attribution, and prompt repair. The independent contributions of these components, including step localization, remain unresolved. End-to-end correction uses CaptainAgent under two workflow structures, with external-trace attribution evaluated on `TraceElephant`. The Bayesian results characterize evidence and decision rules under stated assumptions. Experiments assess observable workflow behavior, while latent beliefs and their distances are modeled analytically. Extending correction to failures requiring changes across several roles, other orchestration frameworks, and adversarial settings remains an open direction.

C PROOFS OF THEORETICAL RESULTS

We provide proofs for the theoretical results in Sections 3 and 4. The role-posterior bound concerns the separation of beliefs. The information-theoretic results concern action classification, and the linear-Gaussian results describe a specified model of evidence and posterior sampling.

C.1 PROOF OF THEOREM 3.2

Theorem 3.2 (ϵ-Stability of Role Posteriors). *For $k \in \{i, j\}$, let $p_k(\theta) = P(\theta \mid k)$ and $L_k(\theta) = P(Y \mid \theta, k)$ at fixed evidence Y. Suppose $\|p_i - p_j\|_{\mathrm{TV}} \le \epsilon_\pi$, $0 \le L_k \le B$, $\|L_i - L_j\|_\infty \le \epsilon_\ell$, and $Z_k = \sum_\theta L_k(\theta) p_k(\theta) \ge \zeta > 0$. Then*

$$\|P(\cdot \mid Y, i) - P(\cdot \mid Y, j)\|_{\mathrm{TV}} \le \frac{2B\epsilon_\pi + \epsilon_\ell}{\zeta}. \tag{8}$$

Proof. Let $A_k(\theta) = L_k(\theta) p_k(\theta)$ be the unnormalized posterior. The triangle inequality gives

$$\|A_i - A_j\|_1 \le \sum_\theta L_i(\theta)|p_i(\theta) - p_j(\theta)| + \sum_\theta |L_i(\theta) - L_j(\theta)| p_j(\theta) \tag{9}$$

$$\le 2B\epsilon_\pi + \epsilon_\ell. \tag{10}$$

Also, $|Z_i - Z_j| \le \|A_i - A_j\|_1$ and $\|A_j\|_1 = Z_j$. Normalizing therefore gives

$$\left\| \frac{A_i}{Z_i} - \frac{A_j}{Z_j} \right\|_1 \le \frac{\|A_i - A_j\|_1}{Z_i} + \|A_j\|_1 \left| \frac{1}{Z_i} - \frac{1}{Z_j} \right| \tag{11}$$

$$= \frac{\|A_i - A_j\|_1 + |Z_i - Z_j|}{Z_i} \tag{12}$$

$$\le \frac{2(2B\epsilon_\pi + \epsilon_\ell)}{\zeta}. \tag{13}$$

Dividing by two yields the stated TV bound. A smaller marginal-likelihood lower bound weakens this stability guarantee. The bound alone does not determine whether the agents select the same action. □

C.2 From Posterior Stability to Action Stability

Let μ_i and μ_j be two role posteriors with $d = \|\mu_i - \mu_j\|_{\mathrm{TV}}$, evaluated at a common state and on a common finite action set. Suppose both decision makers use the same utility function with $|\mathcal{U}(s, a; \theta)| \le C$. For $V_k(a) = \mathbb{E}_{\theta\sim\mu_k}[\mathcal{U}(s, a; \theta)]$,

$$|V_i(a) - V_j(a)| \le 2Cd. \tag{14}$$

This follows by bounding the expectation difference by $C\|\mu_i - \mu_j\|_1$. If the unique maximizer a^* of V_i has gap $V_i(a^*) - \max_{a\neq a^*} V_i(a) > 4Cd$, then

$$V_j(a^*) - V_j(a) \ge V_i(a^*) - V_i(a) - 4Cd > 0, \tag{15}$$

so it remains the unique maximizer under μ_j.

Alternatively, a common stochastic decision kernel $K(a \mid \theta, s)$ contracts TV: $\|\mu_i K - \mu_j K\|_{\mathrm{TV}} \le d$. Indeed, applying the triangle inequality to the mixture over θ and summing K over actions gives this bound. Under a maximal coupling of the action distributions, the disagreement probability is at most d; the claim does not apply to independently sampled actions. Thus, posterior proximity controls actions only with additional assumptions on utilities, margins, or the shared decision kernel.

C.3 Proof of Theorem 4.1

Theorem 4.1 (Lower Bound on Action Error). *Let L^* be an optimal-action label in a space of size $M \ge 2$, with fixed tie resolution. For any decision rule $\hat{a}(Y)$, its error $P_e = P(\hat{a}(Y) \neq L^*)$ satisfies*

$$P_e \ge \frac{H(L^*) - I(L^*; Y) - \log 2}{\log M}. \tag{16}$$

Proof. Fano's inequality and the fact that $\hat{a}$ is based on Y give

$$H(L^* \mid Y) \le H_b(P_e) + P_e \log(M - 1) \le \log 2 + P_e \log M, \tag{17}$$

where H_b is binary entropy. Substituting $H(L^* \mid Y) = H(L^*) - I(L^*; Y)$ and rearranging proves the result. For uniform L^*, $H(L^*) = \log M$, recovering $P_e \ge 1 - [I(L^*; Y) + \log 2]/\log M$. The error event here is action misclassification. A decisive error additionally requires the workflow counterfactual in the definition of agentic misalignment. □

C.4 Proof of Corollary 4.2

Corollary 4.2 (Information Gain Condition). *Let $P_e^*(Y) = \inf_g P(g(Y) \neq L^*)$. If $P_e^*(Y, E) < P_e^*(Y)$, then $I(L^*; E \mid Y) > 0$.*

Proof. We prove the contrapositive. If $I(L^*; E \mid Y) = 0$, conditional independence gives

$$P(L^* = a \mid Y, E) = P(L^* = a \mid Y) \tag{18}$$

for every action a, almost surely. This follows directly from the representation of conditional mutual information as the expected KL divergence between these two conditional distributions. Under zero-one loss, the Bayes-optimal decision selects a label of maximum posterior probability. Hence

$$P_e^*(Y, E) = 1 - \mathbb{E}\left[\max_a P(L^* = a \mid Y, E)\right] \tag{19}$$

$$= 1 - \mathbb{E}\left[\max_a P(L^* = a \mid Y)\right] = P_e^*(Y). \tag{20}$$

Therefore a strict reduction in Bayes-optimal error requires positive conditional mutual information. This argument concerns optimal classification risk and does not infer actual error reduction from a change in Fano's lower bound. □

C.5 Proof of Theorem 4.3

Theorem 4.3 (Covariance Contraction under Linear-Gaussian Evidence). *Suppose* $\mathbf{w} \mid Y \sim \mathcal{N}(\boldsymbol{\mu}_Y, \Sigma_Y)$ *and* $E = \mathbf{H}\mathbf{w} + \boldsymbol{\eta}$*, where* $\mathbf{H}$ *is fixed,* $\boldsymbol{\eta} \sim \mathcal{N}(0, \mathbf{R})$ *is independent of* $(\mathbf{w}, Y)$*, and* $\Sigma_Y, \mathbf{R} \succ 0$*. For* $Y' = (Y, E)$*,*

$$\Sigma_{Y'}^{-1} = \Sigma_Y^{-1} + \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H}, \qquad \Sigma_{Y'} \preceq \Sigma_Y. \tag{21}$$

Proof. Conditional independence implies $P(E \mid \mathbf{w}, Y) = P(E \mid \mathbf{w})$. Multiplying this Gaussian likelihood by the Gaussian posterior given Y, the terms quadratic in $\mathbf{w}$ in the negative log density are

$$\frac{1}{2}(\mathbf{w} - \boldsymbol{\mu}_Y)^\top \Sigma_Y^{-1}(\mathbf{w} - \boldsymbol{\mu}_Y) + \frac{1}{2}(E - \mathbf{H}\mathbf{w})^\top \mathbf{R}^{-1}(E - \mathbf{H}\mathbf{w}). \tag{22}$$

Their quadratic coefficient is $\Sigma_Y^{-1} + \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H}$, giving the posterior precision. For every vector v,

$$v^\top \mathbf{H}^\top \mathbf{R}^{-1} \mathbf{H} v = (\mathbf{H}v)^\top \mathbf{R}^{-1} (\mathbf{H}v) \geq 0. \tag{23}$$

Thus $\Sigma_{Y'}^{-1} \succeq \Sigma_Y^{-1} \succ 0$. Matrix inversion reverses the Loewner order on positive-definite matrices, so $\Sigma_{Y'} \preceq \Sigma_Y$. □

C.6 Proof of Corollary 4.4

Corollary 4.4 (Sufficient Conditions for Reliability Improvement). *Under Theorem 4.3, suppose* a_1 *is the true optimal action among* a_1, a_2 *and* $\delta = \phi(s, a_1) - \phi(s, a_2) \neq 0$*. A policy samples utility weights from its posterior and chooses the action with greater utility. If* $\mu_{D'} = \boldsymbol{\mu}_{Y'}^\top \delta \geq \mu_D = \boldsymbol{\mu}_Y^\top \delta > 0$*, then* $P_e(Y') \leq P_e(Y)$*.*

Proof. For a posterior draw $\widetilde{\mathbf{w}}$, the utility margin $D = \widetilde{\mathbf{w}}^\top \delta$ satisfies

$$D \mid Y \sim \mathcal{N}(\mu_D, \sigma_D^2), \qquad \sigma_D^2 = \delta^\top \Sigma_Y \delta > 0. \tag{24}$$

The sampled policy selects the incorrect action a_2 exactly when $D < 0$, except on a zero-probability tie. Therefore $P_e(Y) = \Phi(-\mu_D / \sigma_D)$, where Φ is the standard normal CDF. Covariance contraction gives

$$0 < \sigma_{D'}^2 = \delta^\top \Sigma_{Y'} \delta \leq \delta^\top \Sigma_Y \delta = \sigma_D^2. \tag{25}$$

Together with $\mu_{D'} \geq \mu_D > 0$, this implies $\mu_{D'}/\sigma_{D'} \geq \mu_D/\sigma_D$. Monotonicity of Φ yields

$$P_e(Y') = \Phi(-\mu_{D'}/\sigma_{D'}) \leq \Phi(-\mu_D/\sigma_D) = P_e(Y). \tag{26}$$

The margin condition concerns the realized posterior update. Zero-mean observation noise alone does not guarantee it for every observation. □

D Training Details

Weak-to-Strong Training. We train AEA-4B by optimizing a reasoning model (using Qwen3-4B as the backbone) on multi-agent workflow traces via a two-stage procedure. Each training example contains the full workflow context (roles, turn-level dialog, tool calls, and results) and a human-verified attribution identifying the responsible agent, a critical error step, and a concrete correction. In the first stage, we apply a supervised warm start on the RM-R1 (Chen et al., 2025) dataset to align the model with standard preference representations. In the second stage, we apply GRPO-based reinforcement learning with verifiable rewards Shao et al. (2024). The reward combines four weighted components. *Agent identification* (40%) checks exact name matches, and *rating alignment* (30%) penalizes deviations from the reference rating. *Correction validity* (20%) checks for a non-trivial revision, and *reasoning completeness* (10%) scores the explanation. Invalid JSON formats receive a fixed penalty of -0.8, with the total reward clamped to $[-1, 1]$. We set the learning rate to 1×10^{-6} and the number of group rollouts to $G = 7$. We maximize the GRPO objective below.

$$\mathcal{J}(\omega) = \frac{1}{G} \sum_{i=1}^{G} [\min(\rho_i(\omega) A_i, \text{clip}(\rho_i(\omega), 1-\varepsilon, 1+\varepsilon) A_i) - \beta \mathbb{D}_{\text{KL}}(\mathcal{M}_\omega || \mathcal{M}_{\text{ref}})] \tag{27}$$

where $\mathcal{M}_\omega$ denotes the evidence model parameterized by ω, $\rho_i(\omega) = \frac{\mathcal{M}_\omega(y_i|x)}{\mathcal{M}_{\text{old}}(y_i|x)}$ is the probability ratio with the old model policy, and A_i is the group-relative advantage. We use a clipping range of $\varepsilon = 0.2$ and a KL penalty $\beta = 0.001$, relative to reference model $\mathcal{M}_{\text{ref}}$. The maximum context length is set to 12,288 for prompts and 6,144 for responses. We train for 5 epochs and use fixed prompts and a shared output schema across the evaluated workflows.

Reward calculation. For a trace x and evidence output y, the reward is

$$R(x, y) = \begin{cases} -0.8, & y \text{ is invalid JSON}, \\ \text{clip}_{[-1,1]}(0.4r_{\text{agent}} + 0.3r_{\text{rating}} + 0.2r_{\text{fix}} + 0.1r_{\text{reason}}), & \text{otherwise}. \end{cases} \tag{28}$$

Here $r_{\text{agent}} = \mathbf{1}[\hat{a} = a^*]$ checks the predicted agent name against the annotated agent. The other components score rating agreement, a non-trivial prompt revision, and explanation completeness, respectively. **Step Accuracy is not a reward component**. The step field localizes the diagnosis and is evaluated separately. These components provide training signals for attribution and repair, while final task correctness is measured by workflow evaluation. The anonymous implementation will provide the component scoring rules. Within each rollout group, rewards are centered and standardized to compute A_i. A constant-reward group provides no relative preference signal.

E Empirical Analyses of Role Adherence and AEA

The Bayesian framework motivates empirical questions about whether agents distinguish their assigned roles, whether specialized evidence improves correction, and whether corrected workflows produce more consistent outputs. We examine these questions through functional overlap, role adherence, task accuracy, and output variability using the setup in Section 5.3. Improved task accuracy without improved role adherence would weaken the role-separation explanation of the gains. These behavioral comparisons complement the conditional analysis in Sections 3 and 4. We also report workflow sizes and token budgets to characterize the evaluated setting.

E.1 Evidence Gradient

We measure correction benefit through the average task-error reduction ΔP_e across four benchmarks. Table 4 compares *Naive Retry* with no new evidence, *Generic Feedback* with a role-agnostic message, *Self-Reflection* using the acting model, and *AEA-4B* using a specialized evidence model under the same GPT-4o workflow. Their gains are 0.005, 0.011, 0.005, and 0.071, respectively. Self-Reflection regresses on `DataBench` (−3.0 pp) and `Chemistry` (−4.8 pp), whereas AEA improves all four benchmarks. AEA's average gain is more than five times that of Generic Feedback. These results show the practical value of specialized evidence under the common correction protocol. The information-theoretic analysis motivates examining evidence quality, while ΔP_e measures realized task-error reduction. Appendix G discusses the relation to capabilities and computational cost.

E.2 Functional Overlap and Role Adherence

Setup. We examine whether unaligned agents display similar behavior across distinct roles, motivated by Theorem 3.2. We instantiate a three-role workflow on `AIME24` with a *Problem Solver* that solves the problem from scratch, a *Solution Verifier* that checks the proposed solution *without re-solving it*, and a *Result Reporter* that only formats the final answer. We measure two behavioral quantities, both judged by GPT-4o with fixed rubrics. *Pairwise Functional Overlap* measures, for every ordered pair of agents, how often they perform functionally equivalent operations across 90 comparisons. It quantifies overlap in the operations performed by agents with different responsibilities. *Role Action Accuracy* measures the fraction of turns whose action matches the assigned role description. A 50-sample human spot-check shows 96% agreement with the judge.

AEA improves behavioral separation between roles. As shown in Table 5, despite the distinct role instructions, unaligned agents exhibit 58.3% functional overlap while only 34.2% of turns are role-aligned. Self-Reflection reduces overlap to 47.1% and increases role accuracy to 42.8%. Weak-to-Strong AEA reduces overlap further to 29.5% and raises role action accuracy to 59.6%. The joint reduction in overlap and improvement in role accuracy supports the use of specialized evidence to distinguish agents' responsibilities in this setting.

Table 5: Functional overlap and role adherence on AIME24 with three distinct roles. AEA reduces overlap and improves role action accuracy, with the largest improvement from Weak-to-Strong evidence.

Setting	Functional overlap (%) ↓	Role action accuracy (%) ↑
Multi-Agent (no alignment)	58.3	34.2
Self-Reflection (AEA)	47.1	42.8
Weak-to-Strong (AEA)	**29.5**	**59.6**

Table 6: Per-task variance of final-answer embeddings across 30 tasks × 3 random seeds. AEA roughly halves output variance. This behavioral measure complements the latent covariance analysis in Theorem 4.3.

Setting	Mean Output Embedding Variance
Multi-Agent (no alignment)	0.0847
Self-Reflection (AEA)	0.0691
Weak-to-Strong (AEA)	**0.0423**

Table 7: Accuracy and average cumulative token cost for GPT-4o. AEA includes the initial execution (11,400 tokens), corrected rerun (12,100 tokens), and evidence-generation cost T_E in this setting.

Method	Total tokens (average)	AIME24	AIME25	DataBench
Single Agent (1×)	∼950	16.7	10.0	27.3
Majority Vote (k=5)	∼4,750	26.7	16.7	31.9
Best-of-K (k=15)	∼14,250	30.0	20.0	33.0
Multi-Agent (no AEA)	∼11,400	23.3	33.3	30.7
Multi-Agent + AEA	$\sim 23{,}500 + T_E$	**40.0**	**33.3**	**35.0**

E.3 Output Variability with AEA

Setup. Under the linear-Gaussian evidence model, Theorem 4.3 gives posterior covariance contraction, $\Sigma_{Y'} \preceq \Sigma_Y$. To study the corresponding behavioral variability, we re-run each of 30 randomly sampled tasks $k = 3$ times, using the three random seeds, under each setting, embed every final answer with a fixed sentence encoder, and measure the per-task variance of the embeddings around their per-task mean.

AEA reduces output variability. As shown in Table 6, the unaligned workflow yields a per-task variance of 0.0847, indicating variability in final answers across repeated runs. Self-Reflection reduces this variance modestly to 0.0691. Weak-to-Strong AEA roughly halves the unaligned variance, reaching 0.0423. These measurements show more reproducible final answers across runs. Output embedding variance is an observable behavioral quantity, while Theorem 4.3 concerns the latent utility covariance in the linear-Gaussian model.

E.4 Accuracy and Cumulative Token Cost

Setup. A natural alternative to AEA is test-time scaling, such as majority voting or best-of-K sampling (Brown et al., 2024; Snell et al., 2024). Table 7 compares accuracy and average cumulative token cost for these strategies on AIME24, AIME25, and DataBench using GPT-4o.

Cumulative correction cost. The initial GPT-4o workflow uses approximately 11,400 tokens, and the corrected rerun uses approximately 12,100 tokens. Thus, the complete AEA episode costs

$$T_{\text{AEA}} = T_{\text{initial}} + T_E + T_{\text{restart}} \approx 23{,}500 + T_E. \tag{29}$$

The 700-token difference compares the two executions; it is not the cost of adding correction. AEA achieves higher accuracy than Best-of-K on all three benchmarks, using a larger cumulative budget than its 14,250 tokens. The evidence-generation cost T_E is kept separate because no value for this setting is reported. Table 17 measures evidence generation in a different workflow setting, so its 3,590 tokens per AEA call are not substituted here.

Table 8: Workflow size distribution on the 556-task GPT-4o subset from HumanEval, DataBench, Chemistry, Physics, AIME24, and AIME25. Percentages describe the share of tasks in this subset.

Agent count	Tasks	Share (%)
2–3 agents	415	74.6
4–5 agents	135	24.3
6+ agents	6	1.1

Table 9: RewardBench results for RM-R1 models. All models use GRPO with RM-R1 training data.

Model	Chat	Chat Hard	Safety	Reasoning	Overall
RM-R1(Qwen2.5-7B)	94.69	74.34	85.00	86.11	85.03
RM-R1(Qwen2.5-14B)	93.57	79.38	87.70	91.40	88.01
RM-R1(Qwen2.5-32B)	93.29	82.02	90.95	94.72	90.25
RM-R1(Qwen3-0.6B)	80.72	60.52	63.37	49.64	63.56
RM-R1(Qwen3-4B)	91.62	75.87	88.11	86.52	86.13

Table 10: RM-Bench results for RM-R1 models. All models use GRPO with RM-R1 training data.

Model	Chat	Math	Code	Safety	Easy	Normal	Hard	Avg
Qwen2.5-7B	67.26	69.41	55.26	92.18	80.96	71.87	60.26	71.03
Qwen2.5-14B	75.10	76.68	60.18	85.46	81.17	75.23	66.67	74.36
Qwen2.5-32B	75.27	81.39	64.81	93.67	85.69	81.01	69.65	78.79
Qwen3-0.6B	55.04	61.73	53.65	63.24	73.75	58.36	43.14	58.42
Qwen3-4B	70.63	72.19	57.21	92.28	81.44	73.99	63.75	73.06

E.5 Workflow Size Distribution

Table 8 reports the number of agents instantiated by CaptainAgent on the 556-task subset comprising HumanEval, DataBench, Chemistry, Physics, AIME24, and AIME25 with GPT-4o. This analysis excludes AIME26 and HLE. Most workflows contain two or three agents, while only six tasks use six or more agents.

Weighting the accuracies of these six benchmarks in Table 1 by their task counts gives approximately 53.2% for the unaligned GPT-4o workflow and 58.6% for Weak-to-Strong AEA, a gain of 5.4 percentage points. These aggregate results summarize performance over all workflow sizes in this subset. The task distribution alone does not establish how accuracy or the AEA gain changes with agent count.

F Standard Reward Evaluations

Tables 9 and 10 provide RM-R1 reference results on RewardBench (Lambert et al., 2025) and RM-Bench (Liu et al., 2025). These benchmarks evaluate preference judgments across chat, safety, reasoning, math, and code. The results give context for the reward reasoning models used in our supervised warm start. The reported rows correspond to RM-R1 checkpoints (Chen et al., 2025), rather than AEA after training on agentic traces.

G Discussion

The Efficiency-Reliability Trade-off. Current trends in agentic AI emphasize scaling by increasing the number of agents and interaction depth (Kim et al., 2025). Each additional agent introduces a decision node where a role deviation may propagate through the workflow. Our matched-restart comparisons show that AEA improves accuracy over retry and critic baselines; cumulative token accounting is discussed in Appendix E.4. The linear-Gaussian analysis explains how informative evidence can reduce posterior covariance, while the experiments measure accuracy and output variability. These findings motivate investing in evidence quality alongside additional computation.

Table 11: Datasets used to generate agentic reasoning traces for training Agentic Evidence Attribution (AEA). The first three benchmarks (GAIA, AssistantBench, LiveBench) are single-agent evaluations that we convert into multi-agent runs, while Who&When contributes only its Algorithm-Generated split; Hand-Crafted is reserved for evaluation. Samples in the first four rows are source counts. **Ours** gives the retained training traces with misalignment diagnostics and human-verified annotations. Appendix K gives the retained counts by source.

Benchmark	Multi Agent	Description	Samples
GAIA (Mialon et al., 2023)	✗	Benchmark agent's reasoning, tool use, web browsing, and multi-modality in real-world tasks.	165
AssistantBench (Yoran et al., 2024)	✗	Benchmark web agents on realistic multi-step tasks across diverse domains.	33
LiveBench (White et al., 2025)	✗	Benchmark with tasks in math, coding, reasoning, language, instruction following, and data analysis.	1436
Who&When (Zhang et al., 2025c)	✓	Algorithm-Generated split for training failure attribution in multi-agent systems.	126
Ours	✓	Multi-agent Misalignment Traces and Diagnostics	**1531**

The condition $I(L^*; E \mid Y) > 0$ states a necessary requirement for evidence to reduce optimal action error.

Capability vs. Evidence. Our framework separates role-specific evidence and task-solving capability as two contributors to workflow performance. A **Fixed** outcome shows that the workflow can solve the task after evidence-conditioned correction. Persistent failures motivate further improvements in evidence, capabilities, or orchestration. Matched-restart comparisons assess the value of the evidence source under a common correction budget.

Attribution as reasoning. Useful correction depends on identifying how a role deviation affects subsequent decisions. A critique can be plausible yet provide little guidance if it overlooks these dependencies or assigns the repair to the wrong agent. AEA trains the evidence model to connect the observed failure, the responsible role, and a concrete revision. We interpret the attribution and correction gains as support for this specialized reasoning process. In this context, informative evidence clarifies what an agent should change and why; the information-theoretic analysis characterizes the corresponding conditions for an ideal decision rule.

Relation to Concurrent Works. Failure attribution and reflection provide complementary routes to improving agent workflows. Who&When benchmarks agent- and step-level diagnosis (Zhang et al., 2025c); AgenTracer trains a specialized tracer and uses its feedback for iterative correction (Zhang et al., 2025a); and COPPER learns role-specific reflections for actor prompts (Bo et al., 2024). GPA evaluates alignment among goals, plans, and actions through factorized judgments (Jia et al., 2025). AEA connects this line of work to an analytical view of role inference and evaluates structured evidence through targeted prompt revision. Its contribution combines this perspective with empirical comparisons of evidence sources, specialized training, and correction under a common one-restart budget. Studies of agent-system scaling (Kim et al., 2025) further motivate examining evidence quality alongside the number of agents and interaction steps.

H Prompt Engineering in Evaluation

The following prompts define the interfaces for failure attribution and correctness evaluation. Attribution prompts request JSON outputs for automated parsing and feedback injection. During evaluation, reference answers and judge outputs are withheld from workflow execution, Self-Reflection, and AEA. They are used only by the separate correctness evaluator.

H.1 AEA System Prompt

This prompt defines the inference interface for AEA-4B. Given a completed workflow trace, it requests a structured attribution identifying the predicted agent, critical step, explanation, collaboration rating,

and revised role prompt. These fields connect trace analysis to a targeted correction. The displayed rating and step values illustrate the numeric field types.

AEA System Prompt

You are an AI assistant that analyzes multi-agent conversations.
Your task is to analyze the conversation and identify issues with agent collaboration.
You MUST respond with a valid JSON object in this exact format:

```
{
"rating": 1,
"agent_name": "<agent>",
"step_number": 0,
"reason": "<text>",
"revised_prompt": "<new prompt>"
}
```

Analysis guidelines: Rate collaboration 1-5 (1=poor, 5=excellent). Identify ONE agent that made the most critical mistake and the step number where it occurred. Note: Computer_terminal is not an agent. Provide a brief reason and improved system prompt.

H.2 Self-Reflection Prompt

To examine the shared-prior interpretation, we query the backbone model (e.g., GPT-4o, Claude) to critique its own trace using the AEA output schema. The template provides the problem, the workflow's final answer, and its conversation history. No reference answer or judge output is supplied. This input restriction applies to every evaluation benchmark.

Self-Reflection Prompt Template

You are analyzing a multi-agent conversation to identify collaboration issues and provide improvement suggestions.
Task Context
Problem: {problem}
Agent's Final Answer: {agent_answer}
Multi-Agent Conversation
{formatted_history}
Your Analysis Task
Analyze this multi-agent conversation and identify:
• Rate the overall collaboration quality (1-5)
• Identify the ONE agent that made the most critical mistake
• Find the step number where the mistake first occurred
• Provide a brief reason explaining the issue
• Suggest an improved system prompt for that agent

Note: "computer_terminal" is a tool, not an agent. Focus on actual agents.
Required Output Format
Respond with a valid JSON object containing: rating, agent_name, step_number, reason, revised_prompt

H.3 LLM-as-a-Judge Prompt

We use GPT-4o to judge final-answer correctness on `DataBench`, `Chemistry`, and `Physics`. This scoring-only prompt converts completed workflow outputs into binary correctness labels. Neither the reference answer nor the judgment is fed back to the agents. For `HumanEval`, we execute the generated implementation against the benchmark tests and count it as correct only if all tests pass; model judgments do not determine its score.

The reported `AIME24`, `AIME25`, and `AIME26` accuracies use deterministic exact numerical matching with the parser described in Appendix K.

LLM Judge Prompt Template

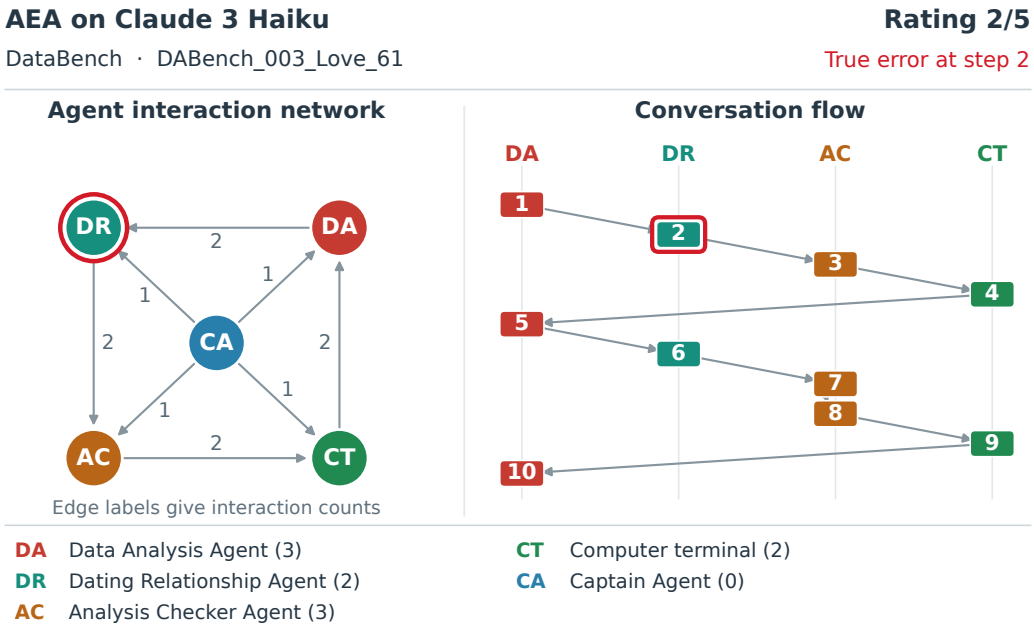
You are evaluating a {solution_type} from a multi-agent system. Please determine if the final answer is correct.
Problem: {question}
Reference Answer (evaluation only): {reference_answer}
Multi-Agent Response: {model_response}
Please evaluate if the multi-agent response correctly solves the problem. Look for:
• **`DataBench`:** Correct answer, data analysis approach, reasoning
• **`SciBench`:** Correct scientific reasoning, calculations, principles
Respond with exactly one of: **CORRECT** (solution is right) or **INCORRECT** (solution is wrong)

I VISUALIZATION ON TOPOLOGICAL GRAPHS

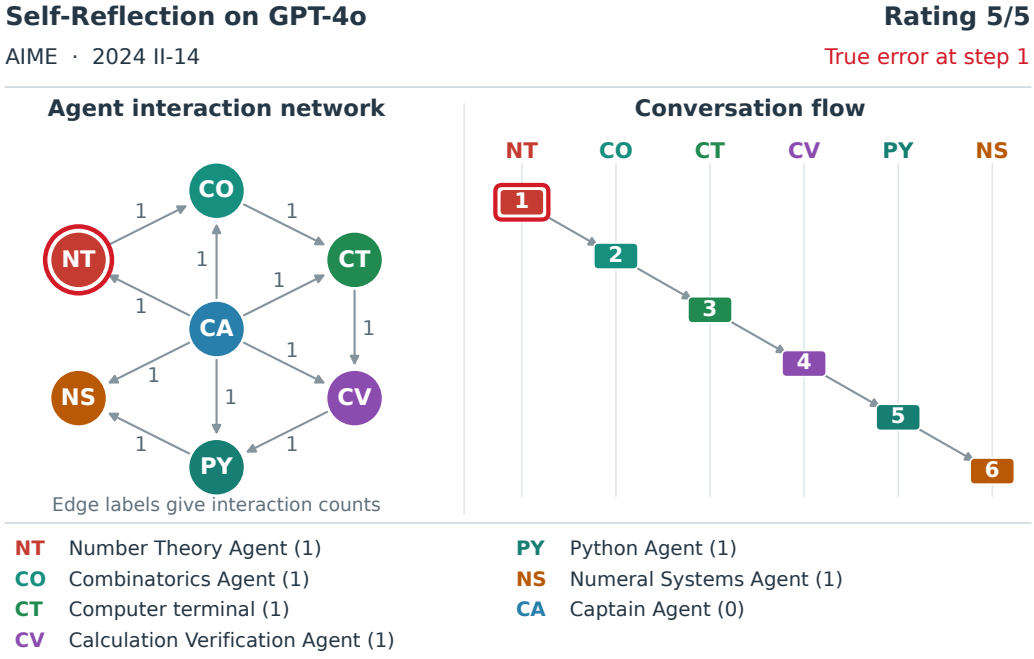
We visualize the workflow execution topology in Figure 4. These graphs show agent interactions and their temporal order around the annotated failures.

AEA enables precise attribution. Figure 4 (Top) displays a `DataBench` workflow using Claude 3 Haiku evaluated by AEA. The annotated critical error occurs at Step 2, in the *Dating Relationship Agent*'s turn, before the *Analysis Checker* acts at Step 3. AEA identifies the annotated error and assigns a low alignment rating $(2/5)$. This example illustrates agreement with the annotated failure location. Appendix J separately presents correction outcomes.

Self-Reflection misses the annotated error. Figure 4 (Bottom) illustrates a failure mode on `AIME` using GPT-4o. Despite the annotated error at Step 1, Self-Reflection assigns a perfect rating $(5/5)$. The example shows how a high self-assigned rating can coexist with an incorrect trajectory, motivating attribution that examines role responsibilities and workflow dependencies.



(a) **AEA on Claude 3 Haiku (`DataBench`).** AEA identifies the annotated critical error at Step 2 and assigns a rating of 2/5.



(b) **Self-Reflection on GPT-4o (AIME).** Self-Reflection assigns a rating of 5/5 despite the error at Step 1.

Figure 4: Workflow topology and failure attribution. Red borders mark the human-annotated critical steps. AEA identifies the annotated error, while Self-Reflection gives the failed trajectory a high collaboration rating.

J CASE STUDIES

We present examples of corrected errors, preserved answers, persistent errors, and regressions. Checkmarks indicate the correctness of the displayed final answers. Feedback excerpts are reproduced from model outputs and may themselves contain errors. Each verification note states the benchmark target or an independent calculation.

J.1 WEAK-TO-STRONG CASES

AEA: Fixed Error (`AIME24`)

Transition: incorrect → correct
Question: Let $\mathcal{B}$ be the set of rectangular boxes with surface area 54 and volume 23. Let r be the radius of the smallest sphere that can contain each box in $\mathcal{B}$. Write $r^2 = p/q$ in lowest terms and find $p + q$.
Original Answer: The final answer is 45 (which is p + q, where $r^2 = 41/4$ with p = 41 and q = 4). ✗
Agent Flagged: `Geometry_Expert` at Step 4
Feedback excerpt: *In step 4, the Geometry_Expert incorrectly states that the dimensions* $2 \times 2 \times 5.75$ *give an exact value of* $r^2 = 41/4$.
After Alignment: Based on the expert conversation, the final answer is: $p + q = 721$. This is because $r^2 = 657/64$, where $p = 657$ and $q = 64$ are relatively prime. ✓
Verification. The largest squared diagonal is attained at dimensions $2, 2, 23/4$. Thus $r^2 = (2^2 + 2^2 + (23/4)^2)/4 = 657/64$, giving $p + q = 721$.

AEA: Fixed Error (`Chemistry`)

Transition: incorrect → correct
Question: A sample of 3.00 mol of diatomic perfect gas at 200 K is compressed reversibly and adiabatically to 250 K. Given $C_{V,\mathrm{m}} = 27.5\ \mathrm{J\,K^{-1}\,mol^{-1}}$, calculate the heat transfer q.
Original Answer: The heat transfer q for the reversible adiabatic compression of the diatomic perfect gas is: $q = 200$ J ✗
Agent Flagged: `Calorimetry_Expert` at Step 2
Feedback excerpt: *The Calorimetry_Expert incorrectly stated that 'In an adiabatic process, there is no heat exchange between the system and the surroundings, which means that the heat transfer, q, is zero.' This is factually correct, but the agent's reasoning for this...*
After Alignment: The heat transfer q for the adiabatic compression process is: $q = 0$ J ✓
Verification. An adiabatic process has $q = 0$, matching the `SciBench` target. The final answer is corrected, although the feedback incorrectly criticizes the statement that adiabatic heat transfer is zero.

AEA: Stayed Wrong (`AIME24`)

Transition: incorrect → incorrect
Question: Among the 900 residents of Aimeville, 195 own a diamond ring, 367 own golf clubs, and 562 own a garden spade. Everyone owns candy hearts. Exactly 437 residents own two of these four items, and exactly 234 own three. How many own all four?
Original Answer: The number of residents of Aimeville who own all four items (a diamond ring, a set of golf clubs, a garden spade, and a bag of candy hearts) is 234. ✗
Agent Flagged: `Statistics_Expert` at Step 13
Feedback excerpt: *The Statistics_Expert incorrectly stated that* $|A \cap B \cap C| = 234$ *(residents owning exactly three items) is the same as* $|A \cap B \cap C \cap D|$ *(residents owning all four items). This is invalid reasoning because the 234 residents who own exactly three items...*
After Alignment: The number of residents of Aimeville who own all four items is 234. ✗
Verification. If x residents own all four items, then $195 + 367 + 562 = 437 + 2 \cdot 234 + 3x$, so $x = 73$. Both displayed answers are incorrect. Since everyone owns candy hearts, the two intersections mentioned in the feedback are equal. The error is assigning 234 to either intersection.

J.2 Self-Reflection Cases

Self-Reflection: Regression (`AIME25`)

Transition: correct → incorrect
Question: On $\triangle ABC$, points A, D, E, B lie in that order on AB, with $AD = 4$, $DE = 16$, and $EB = 8$. Points A, F, G, C lie in that order on AC, with $AF = 13$, $FG = 52$, and $GC = 26$. Let M be the reflection of D through F, and N the reflection of G through E. The area of $DEGF$ is 288. Find the area of heptagon $AFNBCEM$.
Original Answer: 588 ✓
Agent Flagged: `Verification_Specialist` at Step 10
Feedback excerpt: *The Verification_Specialist correctly identified a discrepancy between the triangle decomposition and shoelace formula results, but misinterpreted it as a self-intersection problem rather than recognizing the correct area calculation method.*
After Alignment: The area of heptagon AFNBCEM is 1536. ✗
Verification. The side ratios give $[DEGF] = (24/49)[ABC]$, so $[ABC] = 588$. Applying the shoelace formula with $M = 2F - D$ and $N = 2E - G$ gives $[AFNBCEM] = [ABC] = 588$. The original answer is correct and the revised answer is incorrect.

Self-Reflection: Preserved (`DataBench`)

Transition: correct → correct
Question: In the FIFA Football Dataset, what is the most common preferred foot amongst players?
Original Answer: Right ✓
Agent Flagged: `None` at Step 0
Feedback excerpt: *There were no critical mistakes in this collaboration. All agents worked effectively together, providing complementary expertise from different domains. The FIFA_Data_Analyst_Expert led the discussion appropriately.*
After Alignment: Right ✓
Verification. The `DataBench` reference for this question in `007_Fifa` is `Right`. Both displayed answers match it.

Self-Reflection: Stayed Wrong (`AIME24`)

Transition: incorrect → incorrect
Question: Points A, B, C, D lie on the hyperbola $x^2/20 - y^2/24 = 1$ and form a rhombus whose diagonals intersect at the origin. Find the greatest real number that is less than BD^2 for every such rhombus.
Original Answer: 80 ✗
Agent Flagged: `none` at Step 0
Feedback excerpt: *The collaboration was excellent, with all agents contributing effectively to solving the problem. The GeometricConstraints_Expert correctly set up the parametrization of points on the hyperbola, derived the constraint $t_A = -t_B$ for forming a rhombus.*
After Alignment: The greatest real number that is less than BD^2 for all such rhombi is 191.9999... or simply 192. ✗
Verification. If BD has slope m, the perpendicular diagonals require $5/6 < m^2 < 6/5$. Then $BD^2 = 480(1 + m^2)/(6 - 5m^2) > 480$, with infimum 480. Both 80 and 192 are incorrect.

K Evaluation Protocols and Detailed Results

Evaluation setup. Our task evaluation suite comprises `HumanEval`, `DataBench`, `Chemistry`, `Physics`, `AIME24`, `AIME25`, `AIME26`, and `Humanity's Last Exam` (`HLE`) (Phan et al., 2025). Table 1 covers the first seven benchmarks, containing 164, 257, 41, 34, 30, 30, and 30 tasks, respectively. Accuracy results average three random seeds. Table 2 compares correction on `AIME26` and the 2,500 `HLE` questions, scored with its official evaluator. Across benchmarks, reference answers and judge outputs are restricted to final scoring and withheld from the workflow and evidence models. Failure attribution uses the fully held-out Hand-Crafted split of `Who&When` and 85 held-out `TraceElephant` traces (Chen et al., 2026).

Workflow correction. For correction, we first execute the workflow and collect its complete trace. The evidence model then identifies an agent and revises its system prompt, after which the full workflow restarts once. The remaining role prompts and workflow structure are unchanged. The first trace informs the prompt revision but is not carried into the second run as execution history. The attributed step identifies the error and is not a restart point. Thus, the completed first trace supplies evidence for a new execution; the history-prefix analysis in Section 4.1 describes a decision-time abstraction. Naive Retry uses the same restart with unchanged prompts. The independent Gemma-4-E4B-it critic receives the full trace and produces the same structured evidence as AEA. We reproduce Goal-Plan-Action (Jia et al., 2025) under this interface and refer to it as Reproduced GPA.

Correction with strong workflow models. Across GPT-5.5 and Claude Opus 4.8 on the seven benchmarks in Table 1, AEA improves on the original workflow in 13 of 14 settings and ties in one, with a benchmark-macro gain across the 14 model–benchmark settings. Weak-to-Strong uses AEA-4B as the evidence model. On `AIME26`, it improves over both the original workflow and Self-Reflection for each model. These comparisons average accuracy over three seeds.

Paired evaluation across tasks. The paired analysis covers the 556 tasks from `HumanEval`, `DataBench`, `Chemistry`, `Physics`, `AIME24`, and `AIME25`, using GPT-5.5 and Claude Opus 4.8 with three seeds. Table 12 reports task-weighted accuracy. The paired bootstrap resamples task identities while keeping all model and seed outcomes from the same task together. The combined result improves from 67.27% to 70.05%, with a 95% confidence interval of $[1.38, 4.20]$ percentage points for the gain. This corresponds to an 8.52% reduction in aggregate error. This analysis excludes `AIME26` and `HLE`. Its confidence intervals describe the aggregate gain across the six included benchmarks.

The macro and task-weighted summaries answer different questions: the former weights benchmark settings equally, while the latter weights tasks equally. `DataBench` contributes 257 of the 556 tasks, or 46.2% of this subset. As a descriptive check using the rounded accuracies in Table 1, excluding `DataBench` leaves an approximate task-weighted gain of 2.7 percentage points across the remaining five benchmarks. This calculation summarizes the existing table; the reported paired confidence interval applies to the full six-benchmark subset.

Table 12: Paired analysis on 556 tasks from `HumanEval`, `DataBench`, `Chemistry`, `Physics`, `AIME24`, and `AIME25`. Accuracies are percentages. Gains and their task-clustered 95% confidence intervals are in percentage points.

Workflow model	Multi Agent	AEA-4B	Gain	95% CI
GPT-5.5	65.29	67.63	+2.34	[0.78, 3.90]
Claude Opus 4.8	69.24	72.48	+3.24	[1.54, 4.93]
Combined	67.27	70.05	+2.79	[1.38, 4.20]

Correction on `HLE`. Table 2 presents the main `HLE` comparison. On `HLE`, AEA raises GPT-5.5 workflow accuracy from 44.28% to 51.04%, compared with 44.88% for Naive Retry and 46.72% for the independent Gemma critic (Table 13). AEA solves an average of 108 more questions per seed than the critic under the same one-restart budget. The paired gain is 4.32 percentage points with a 95% confidence interval of $[2.78, 5.86]$. The gain over Naive Retry is 6.16 points with an interval of $[4.47, 7.85]$. These results show that specialized evidence improves correction beyond both another execution and an off-the-shelf external critic.

Table 13: Workflow correction on all 2,500 `HLE` questions with GPT-5.5 in an orchestrator-worker workflow. Counts are means over three seeds. Each correction method uses one complete workflow restart.

Setting	Mean correct / 2,500	Accuracy (%)
Original workflow	1,107	44.28
Naive Retry	1,122	44.88
Gemma-4-E4B-it critic	1,168	46.72
Weak-to-Strong (AEA-4B)	**1,276**	51.04

Failure attribution on external traces. We evaluate on 85 held-out `TraceElephant` traces (Chen et al., 2026) in the All at Once setting without demonstrations. The traces include executions from Magentic-One and SWE-agent. Table 14 compares AEA-4B with three general-purpose models. AEA-4B obtains the highest point estimates for both Agent Accuracy and Step Accuracy in this comparison. These results evaluate attribution on external traces, separately from the end-to-end correction experiments using CaptainAgent.

Table 14: Failure attribution on 85 `TraceElephant` traces. Accuracy is reported in percent.

Evidence model	Agent Accuracy	Step Accuracy
GPT-5.5	67.1	27.1
Claude Sonnet 4.6	64.7	27.1
Gemini 3.1 Pro	51.8	20.0
AEA-4B	**70.6**	**29.4**

Matched-backbone training. To isolate the effect of specialized training, we compare the same Gemma-4-E4B-it checkpoint before and after AEA training (Table 15). With the backbone and inference budget fixed, Agent Accuracy increases from 69.4% to 80.0% and Step Accuracy from 25.9% to 36.5%. The Agent Accuracy gain is 10.59 percentage points with a 95% confidence interval of $[2.35, 20.00]$. On 60 `AIME24/25` tasks under orchestrator-worker and sequential workflows, the mean number of correct outcomes increases from 76 to 90 per 120 task–workflow outcomes. The 11.67-point gain has a 95% confidence interval of $[4.17, 19.17]$. Specialized training therefore improves both failure attribution and the resulting workflow correction. We interpret this joint improvement as support for learning to reason from workflow dependencies to informative, role-specific repairs.

Table 15: Effect of AEA training with the same Gemma backbone. AEA-4B is included as a reference. Attribution uses 85 `TraceElephant` traces. Final correctness uses the same 60 `AIME24/25` tasks under orchestrator-worker and sequential workflows, giving 120 outcomes per seed. Counts are means over three seeds.

Evidence model	Agent Accuracy	Step Accuracy	Final correctness
Gemma-4-E4B-it	59/85 (69.4)	22/85 (25.9)	76/120 (63.3)
AEA-4B	60/85 (70.6)	25/85 (29.4)	82/120 (68.3)
AEA-trained Gemma	**68/85 (80.0)**	**31/85 (36.5)**	**90/120 (75.0)**

Evidence sources and workflow structure. We compare correction methods on the same 60 `AIME24/25` tasks under orchestrator-worker and sequential workflows. Each method uses one complete restart, giving 120 task-workflow outcomes per seed. Naive Retry uses unchanged prompts. The Gemma-4-E4B-it critic receives the complete trace and produces the same evidence schema as AEA. Reproduced GPA adapts the factorized diagnostic design to this correction interface. Table 16 reports the combined final correctness.

AEA-4B improves over Naive Retry by 10.0 percentage points with a 95% confidence interval of $[2.5, 17.5]$. Its gain over the Gemma critic is 5.0 points with interval $[0.83, 9.17]$. Both intervals are computed over the 60 tasks while retaining all workflow and seed outcomes from each task together. Training the Gemma checkpoint with AEA further improves the mean correct count from 76 to 90 per seed, or from 63.3% to 75.0%.

Table 16: Final correctness on 60 `AIME24/25` tasks under two workflow structures. Counts are means over three seeds. All correction methods use one complete restart.

Evidence source	Mean correct / 120	Accuracy (%)
Naive Retry	70	58.3
Reproduced GPA	74	61.7
Gemma-4-E4B-it critic	76	63.3
AEA-4B	82	68.3
AEA-trained Gemma	**90**	75.0

Table 17 separates the two workflow structures and reports evidence validity and token usage. AEA-4B improves over the Gemma critic in the orchestrator-worker setting and ties it in the sequential setting. It also produces valid evidence in more runs. The comparison evaluates the complete evidence pipeline, including output validity, attribution, and prompt revision. The matched-backbone results further show that AEA training improves both agent identification and downstream correction. Token counts report mean prompt and completion usage for evidence generation over 120 runs per seed, excluding the initial workflow and the restarted execution. Reproduced GPA is our implementation under the common interface, so its results also reflect this adaptation.

Table 17: Results by workflow structure, averaged over three seeds. Correct counts use 60 tasks per structure; valid evidence uses 120 runs per seed. Tokens measure evidence generation only.

Evidence source	Orchestrator worker	Sequential	Valid evidence	Evidence tokens
Gemma critic	34/60	42/60	103/120	433.9K
Reproduced GPA	35/60	39/60	63/120	5.257M
AEA-4B	40/60	42/60	118/120	430.8K

Evaluation checks. The role-adherence audit checks 50 role judgments from `AIME24` workflows and finds 48 agreements with human labels. A separate auxiliary audit of the final-answer judge checks 100 outputs across `HumanEval`, `DataBench`, `Chemistry`, and `Physics`, finding 98 agreements (Table 18). This audit assesses judge agreement; reported `HumanEval` accuracy instead requires executing each generated implementation and passing all benchmark tests. Compilation or execution failures count as incorrect. `AIME24`, `AIME25`, and `AIME26` use deterministic numerical matching with a Python parser. The parser prioritizes a boxed answer and an explicit final-answer marker, extracts the final integer, and compares it with the reference. Missing, non-integer, or unequal answers are incorrect.

Table 18: Agreement between automated judgments and human labels. Intervals are Wilson 95% confidence intervals in percent.

Judgment	Agreement	Agreement (%)	95% CI
Role Action Accuracy	48/50	96.0	[86.5, 98.9]
Final-answer correctness	98/100	98.0	[93.0, 99.4]

Training sources and screening. The curated training set contains 1,531 traces, with 165 from `GAIA`, 33 from `AssistantBench`, 1,207 from `LiveBench`, and 126 from the Algorithm-Generated split of `Who&When`. For `Who&When`, training uses only Algorithm-Generated; evaluation uses only Hand-Crafted. The Hand-Crafted traces and their annotations are excluded from AEA training in full. Source benchmark counts and retained training counts differ because filtering precedes trace generation. The five experts select the agent and step by majority vote and edit the proposed reason and revised prompt.

For `LiveBench`, we use code commit `46516996aa679f0d43b0680055f8d8dd7c18b4b9` and math snapshot `bb66571c8ccf32d3df9e6f48b920d3770ff4aacb`. The metadata filter removes all 14 `aime_i_2024` rows and all 15 `aime_ii_2024` rows before trace generation. These 29 source rows correspond to `AIME24` problems. The snapshot contains no `AIME25` items. AMC12 2023 and SMC 2023 items are retained because they are not part of our evaluation.

The overlap audit reports no detected overlap between the curated training set and the checked evaluation sources. These checks include the full `DataBench` and `SciBench` source pools, containing 1,830 and 692 items, respectively, rather than only the evaluated subsets. A separate comparison of `GAIA` and `LiveBench` finds zero normalized exact matches and zero literal-containment pairs of at least 80 characters. For `HLE`, we compare questions and answers against the RM-R1 warm-start data and all 1,531 AEA training traces and find no detected overlap.