

AERO: Autonomous Evolutionary Reasoning Optimization via Endogenous Dual-Loop Feedback

Zhitao Gao^{1,2}, Jie Ma^{2,3*}, Xuhong Li⁴, Pengyu Li^{1,5}, Ning Qu^{1,5},
Yaqiang Wu^{1,6}, Hui Liu⁶, Jun Liu^{1,2,5}

¹School of Computer Science and Technology, Xi'an Jiaotong University

²MOE KLINNS Lab, Xi'an Jiaotong University

³School of Cyber Science and Engineering, Xi'an Jiaotong University

⁴Baidu Inc.

⁵Shaanxi Province Key Laboratory of Big Data Knowledge Engineering

⁶Lenovo AI Technology Center, CTOO, Lenovo

gaozhitao@stu.xjtu.edu.cn jiema@xjtu.edu.cn

Abstract

Large Language Models (LLMs) have achieved significant success in complex reasoning but remain bottlenecked by reliance on expert-annotated data and external correctness verifiers during training. While existing self-evolution paradigms aim to bypass these constraints, they often fail to identify the optimal learning zone and risk reinforcing collective hallucinations and incorrect priors through flawed internal feedback. To address these challenges, we propose **Autonomous Evolutionary Reasoning Optimization (AERO)**, an unsupervised framework that achieves autonomous reasoning evolution by internalizing self-questioning, answering, and criticism within a synergistic dual-loop system. Inspired by the *Zone of Proximal Development (ZPD)* theory, AERO utilizes entropy-based positioning to target the “solvability gap” and employs Independent Counterfactual Correction (ICC) to form high-reliability endogenous pseudo-labels through counterfactual convergence. Furthermore, we introduce a Staggered Training Strategy to synchronize capability growth across functional roles and prevent curriculum collapse. Extensive evaluations across nine benchmarks spanning three domains demonstrate that AERO achieves average performance improvements of 4.57% on Qwen3-4B-Base and 5.10% on Qwen3-8B-Base, outperforming competitive baselines. Code is available at <https://github.com/mira-ai-lab/AERO>.

1 Introduction

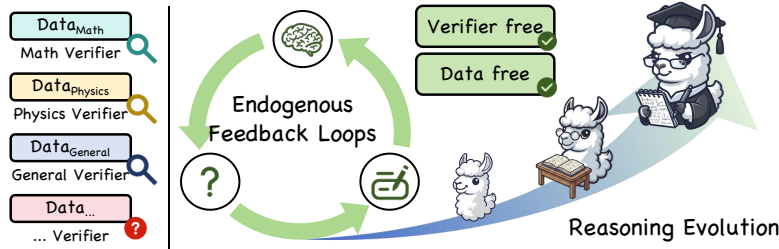


Figure 1: The AERO framework for unsupervised self-evolution. AERO unifies self-questioning, answering, and criticism within a dual-loop system, enabling autonomous growth in reasoning without external training data or correctness verifiers.

*Jie Ma is the corresponding author.

Recently, Large Language Models (LLMs) have demonstrated strong reasoning capabilities (Jaech et al., 2024; Guo et al., 2025; Ma et al., 2025). While Reinforcement Learning with Verifiable Rewards (RLVR) has advanced this progress (Lambert et al., 2024), its reliance on expert-level queries and automated verifiers like math engines (Zhao et al., 2025; Huang et al., 2025) limits model growth to predefined data and prevents the discovery of novel reasoning patterns (Yuan et al., 2024). Consequently, the **Self-Evolution** paradigm has emerged (Tan et al., 2024; Tao et al., 2024), enabling models to iteratively improve through their own generated data and experiences rather than remaining passive recipients of information (Huang et al., 2026; Kuba et al., 2025).

However, existing Self-Evolving paradigms face two fundamental limitations: 1) Current mechanisms often cause the model to fall into **sub-optimal learning zones** because they lack a clear way to adjust task difficulty. This sub-optimal zone occurs when task generation misses the *solvability gap*, which represents the area where tasks are neither too simple to provide new insights nor too hard for the model to understand (Zhang et al., 2025). Without a strategy to target this gap, the model faces a significant loss in learning efficiency. It either stops progressing by repeating what it already knows or fails to learn because it faces overly complex problems that produce feedback no better than random noise (Kuba et al., 2025; Zhao et al., 2025). 2) To replace external correctness verifiers, existing paradigms typically rely on internal indicators such as majority voting (He et al., 2025; Huang et al., 2026) and decoding confidence (Liu et al., 2025; Yu et al., 2025; Zhou et al., 2026) to provide reward signals. These methods work on the assumption that agreement or high probability is the same as logical correctness. However, when a model holds an incorrect belief, these indicators become unreliable and instead **reinforce collective hallucinations and incorrect priors**. This traps the model in a feedback loop that confirms its own mistakes, which eventually drives the learning process away from logical truth.

To address these challenges, we propose **Autonomous Evolutionary Reasoning Optimization (AERO)**, an unsupervised framework structured as an inner-outer dual-loop system that internalizes three synergistic capabilities within a single LLM: Self-Questioning, Self-Answering, and Self-Criticism, as illustrated in Figure 1. To prevent the LLM from falling into sub-optimal learning zones, AERO is inspired by the theory of the *Zone of Proximal Development (ZPD)* (Vygotsky, 1978), which posits that cognitive development is maximized when task difficulty is precisely calibrated to the learner’s current reasoning capabilities. We operationalize this principle by defining task difficulty through the LLM’s level of reasoning uncertainty. Specifically, tasks that exhibit a moderate degree of uncertainty for the current LLM signify the ideal “solvability gap”. AERO utilizes normalized Shannon entropy to quantify this uncertainty, guiding the LLM to autonomously generate tasks that fall within its optimal learning zone at the frontier of its reasoning capabilities.

To overcome the risk of reinforcing collective hallucinations and incorrect priors, we introduce **Independent Counterfactual Correction (ICC)**, which forces the LLM to reconstruct reasoning under the assumption of initial failure. By requiring the answer convergence of independent paths rather than mere statistical consensus, ICC provides a high-reliability truth proxy for the policy optimization. Additionally, we implement the **Staggered Training Strategy**, a simple but effective approach to synchronize role growth and prevent curriculum collapse. Evaluations across nine benchmarks spanning general reasoning, mathematical reasoning, and physical reasoning domains demonstrate that AERO consistently outperforms competitive baselines. Specifically, it achieves average performance gains of 4.57% on Qwen3-4B-Base and 5.10% on Qwen3-8B-Base, while maintaining a consistent improvement trend across diverse architectures.

The primary contributions of this work are as follows:

(1) We propose the Autonomous Evolutionary Reasoning Optimization (AERO) dual-loop framework, which achieves autonomous self-evolution without external correctness verifiers or labels during training. This is the first framework to achieve the simultaneous evolution of Self-Questioning, Self-Answering, and Self-Criticism within a single LLM, enabling a comprehensive reasoning evolution.

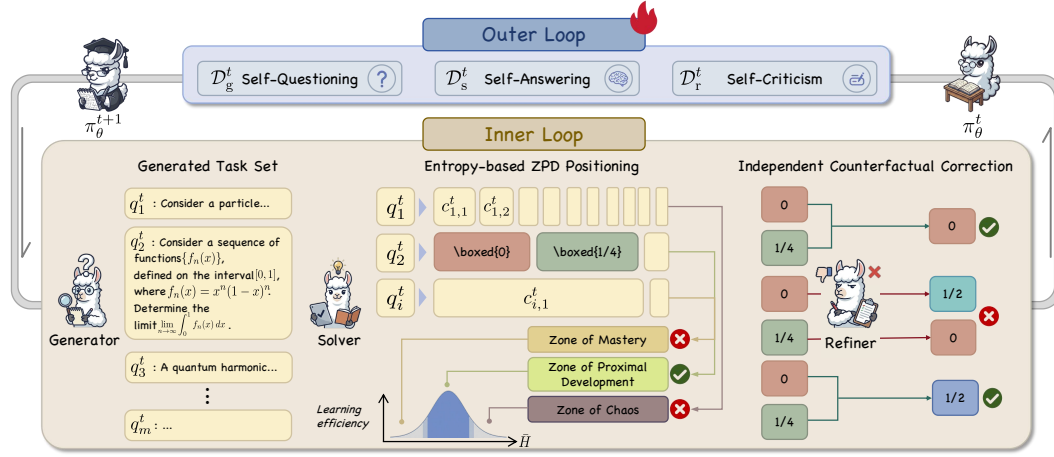


Figure 2: The AERO framework consists of an inner loop for autonomous experience synthesis and an outer loop for preference-based policy optimization. Within the inner loop, the single model adopts generator, solver, and refiner roles to produce tasks and reasoning trajectories. These ICC-filtered experiences are then utilized in the outer loop for policy update.

(2) We introduce entropy-based Zone of Proximal Development positioning to target the optimal learning zone and Independent Counterfactual Correction to provide high-reliability consensus via counterfactual convergence. Additionally, we propose a Staggered Training Strategy to mitigate curriculum collapse and stabilize the evolutionary dynamics.

(3) We conduct extensive evaluations across nine benchmarks to verify the effectiveness and superiority of AERO. Furthermore, we demonstrate its robustness and evolutionary stability in sustaining continuous growth, proving that reasoning capabilities can grow effectively through purely endogenous feedback.

2 Methodology

2.1 Framework Overview

AERO enables a single LLM (π_θ) to autonomously evolve via a dual-loop architecture. The inner loop acts as an experience factory where π_θ^t synthesizes preference data $\mathcal{D}^t$ through self-play. The outer loop utilizes $\mathcal{D}^t$ to update parameters from π_θ^t to π_θ^{t+1} via preference optimization, progressively shifting the reasoning frontier toward higher complexity.

2.2 Inner Loop

The inner loop functions as a self-play sandbox where the Generator (π_g^t), Solver (π_s^t), and Refiner (π_r^t) collaborate to synthesize ICC-filtered experiences in round t . Crucially, these three roles represent distinct functional capacities of π_θ^t , which are activated through specific task-oriented prompts in Appendix B. This process is driven by two synergistic mechanisms: entropy-based ZPD positioning and ICC-based counterfactual consensus.

2.2.1 Entropy-based ZPD Positioning

To identify the reasoning capability frontier where learning is most effective, we implement a selection mechanism inspired by the theory of ZPD (Vygotsky, 1978). In round t , the Generator π_g^t first synthesizes a set of m challenging, competition-level reasoning tasks $\mathcal{Q}^t = \{q_1^t, \dots, q_m^t\}$. For each specific task $q_i^t \in \mathcal{Q}^t$, the Solver π_s^t generates n independent reasoning trajectories $\mathcal{Y}_i^t = \{y_{i,1}^t, \dots, y_{i,n}^t\}$. The final answers are extracted from these n

trajectories and grouped into k unique clusters $C_i^t = \{c_{i,1}^t, \dots, c_{i,k}^t\}$ based on their result equivalence, the specific implementation of which is detailed in Appendix D.1.

We quantify reasoning uncertainty via **Shannon entropy** over the empirical distribution of answer clusters. This metric serves as a proxy for task difficulty relative to π_θ^t , enabling the identification of the model’s reasoning frontier. Formally, the Normalized Shannon Entropy $\bar{H}(q_i^t)$ is defined as:

$$\bar{H}(q_i^t) = -\frac{1}{\log_2 n} \sum_{j=1}^k P(c_{i,j}^t) \log_2 P(c_{i,j}^t), \quad (1)$$

where $P(c_{i,j}^t) = |c_{i,j}^t|/n$ denotes the empirical frequency of the j -th answer cluster for task q_i^t in round t . The normalization factor $1/\log_2 n$ ensures that $\bar{H}(q_i^t)$ remains within the interval $[0, 1]$, where a value of 0 indicates total consensus and 1 represents maximum divergence.

By mapping these entropy values to the cognitive landscape, we categorize each task q_i^t into three distinct regions based on the current reasoning capability of the model:

Zone of Mastery ($\bar{H}(q_i^t) < \tau_{low}$): Tasks falling within this range are those where high consensus indicates the required logic is already internalized by π_θ^t , which offers a negligible learning gradient for further policy evolution.

Zone of Proximal Development ($\tau_{low} \leq \bar{H}(q_i^t) \leq \tau_{high}$): This is the optimal learning zone where moderate reasoning uncertainty identifies the solvability gap. These tasks are most conducive to the cognitive growth of the model.

Zone of Chaos ($\bar{H}(q_i^t) > \tau_{high}$): These tasks far exceed the model’s reasoning frontier. Overwhelming complexity forces the LLM to produce inconsistent, random guesses that act as training “noise,” potentially leading to instability.

AERO focuses exclusively on the *Zone of Proximal Development* for the subsequent stages of the framework. This filtering process retains the tasks that provide the most productive learning signals for policy optimization. The sensitivity of this selection to τ_{low} and τ_{high} is examined in Appendix G.2.

2.2.2 ICC-based Counterfactual Consensus

For each task q_i^t positioned within the ZPD, we establish truth proxies through ICC. Unlike majority voting or decoding confidence, which often reinforce collective hallucinations in self-evolution, ICC begins by identifying the two most frequent answer clusters, $c_{i,1}^t$ and $c_{i,2}^t$, generated by the Solver π_s^t for task q_i^t , which represent the model π_θ^t ’s primary competing consensus. The Refiner π_r^t is then prompted to re-solve the task q_i^t while operating under the counterfactual assumption that the previously proposed solution is incorrect. This constraint breaks the cycle of confirmation bias, forcing the LLM to rethink the task and construct a fresh reasoning path. The counterfactual constraint is intended to reduce confirmation bias, but it does not guarantee independence or logical correctness.

The correction path starting from cluster $c_{i,j}^t$, where $j \in \{1, 2\}$, is represented as $\tilde{y}_{i,j}^t$. We define the formal convergence condition as:

$$\text{res}(\tilde{y}_{i,1}^t) = \text{res}(\tilde{y}_{i,2}^t), \quad (2)$$

where the function $\text{res}(\cdot)$ extracts the final answer from a reasoning trajectory. The reasoning path $\tilde{y}_{i,1}^t$ is used as a high-reliability endogenous pseudo-label $\tilde{y}_i^t$ if this equality holds. Otherwise, if the correction trajectories fail to yield consistent results, the task is considered unresolved and is discarded from the synthesis of training datasets for both the Solver and Refiner roles.

2.2.3 Tri-role Preference Synthesis

The inner loop ends with transforming ICC-filtered experiences from round t into three specialized preference datasets: $\mathcal{D}_g^t$, $\mathcal{D}_s^t$, and $\mathcal{D}_r^t$. To facilitate preference optimization, we map synthesized experiences to binary labels $z \in \{0, 1\}$, where $z = 1$ indicates a *chosen* output and $z = 0$ denotes a *rejected* one.

For the Generator, $\mathcal{D}_g^t$ utilizes all m generated tasks to help the LLM identify its reasoning frontier. We use the indicator function $\mathbb{I}_{\text{ZPD}}(q_i^t)$, which equals 1 if a task falls within the Zone of Proximal Development and 0 otherwise:

$$\mathcal{D}_g^t = \{(q_i^t, \mathbb{I}_{\text{ZPD}}(q_i^t))\}_{i=1}^m. \quad (3)$$

For the Solver, $\mathcal{D}_s^t$ is synthesized by evaluating initial reasoning trajectories against the ICC-derived pseudo-label $\tilde{y}_i^t$ based on their result equivalence:

$$\mathcal{D}_s^t = \{(q_i^t, y_{i,j}^t, \mathbb{I}[\text{res}(y_{i,j}^t) = \text{res}(\tilde{y}_i^t)]) \mid \mathbb{I}_{\text{ZPD}}(q_i^t) = 1, 1 \leq i \leq m, 1 \leq j \leq n\}. \quad (4)$$

For the Refiner, $\mathcal{D}_r^t$ captures the self-correction process by extracting trajectories that transition from a rejected state to the ICC-derived pseudo-label. Specifically, we retain correction paths $\tilde{y}_{i,j}^t$ as positive samples only if the initial cluster result is incorrect, yet the subsequent refinement successfully reaches the truth proxy:

$$\mathcal{D}_r^t = \{(q_i^t, c_{i,j}^t, \tilde{y}_{i,j}^t, 1) \mid \mathbb{I}_{\text{ZPD}}(q_i^t) = 1, \text{res}(c_{i,j}^t) \neq \text{res}(\tilde{y}_i^t) \wedge \text{res}(\tilde{y}_{i,j}^t) = \text{res}(\tilde{y}_i^t), 1 \leq i \leq m, j \in \{1, 2\}\}. \quad (5)$$

Crucially, the datasets for the Solver and Refiner are constructed exclusively from the subset of tasks that satisfy the ZPD criteria, ensuring the LLM learns from the most productive signals.

By constructing these datasets independently, the framework prepares the necessary signals for the subsequent outer loop optimization. This decoupled organization is essential for the Staggered Training Strategy, a mechanism we discuss in detail in Section 2.3.1.

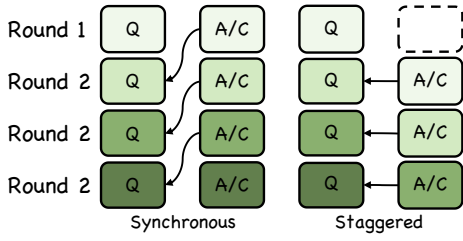
2.3 Outer Loop

The outer loop translates the synthesized experiences into policy updates by optimizing π_θ across its three functional roles. To ensure a stable evolutionary trajectory, we introduce a temporal decoupling mechanism that synchronizes the growth of different capabilities.

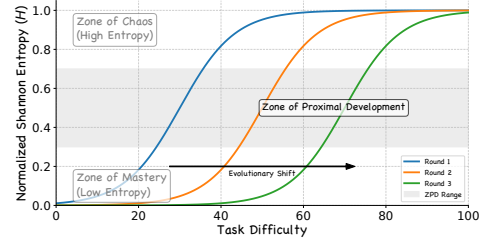
2.3.1 Staggered Training Strategy

A major limitation in LLM self-evolution is *curriculum collapse*, where performance stops improving or even declines during iterative self-play (Huang et al., 2026; Jiang et al., 2025; Wang et al., 2025b). We attribute one of the causes of this instability to capability asynchrony, where the learning speed of Self-Answering and Self-Criticism capabilities exceeds that of Self-Questioning. Specifically, as shown in Figure 3a, under the standard synchronous training strategy, π_θ^t masters its round t ZPD tasks after the outer-loop parameter update. However, during round $t + 1$, the newly synthesized ZPD tasks remain anchored to the capability of π_θ^{t-1} because the diagnostic ZPD signals are derived from the responses of π_θ^{t-1} . Consequently, these tasks have already entered the Zone of Mastery for the updated model π_θ^t , leading to vanishing learning gradients and subsequent training failure.

To mitigate this asynchrony, we introduce the **Staggered Training Strategy**, a simple but effective approach designed to synchronize the advancement of role-specific capabilities. This mechanism creates a temporal offset by using current Self-Questioning data alongside historical Self-Answering and Self-Criticism data to ensure the curriculum remains



(a) Synchronous vs. Staggered Training Strategy. Colors indicate the progression of training rounds. Arrows illustrate the capability alignment between Self-Questioning (Q) and Self-Answering/Self-Criticism (A/C).



(b) Evolution of ZPD and response entropy across iterations. The rightward shift of the curves from Round 1 to Round 3 demonstrates the model's cognitive growth.

Figure 3: Training strategy comparison and ZPD evolution.

challenging. The training dataset $\mathcal{D}_{\text{total}}^t$ at round t is formally organized as follows:

$$\mathcal{D}_{\text{total}}^t = \begin{cases} \mathcal{D}_{\text{g}}^t, & t = 1, \\ \mathcal{D}_{\text{g}}^t \cup \mathcal{D}_{\text{s}}^{t-1} \cup \mathcal{D}_{\text{r}}^{t-1}, & t > 1. \end{cases} \quad (6)$$

By implementing this staggered data flow, the AERO framework effectively prevents curriculum collapse. This strategy ensures that the tasks generated in each round consistently target the updated Zone of Proximal Development of the model, which maintains a steady evolutionary pressure and drives a continuous capability advancement. We provide detailed empirical evidence for the efficacy of this strategy in Section 3.3.

2.3.2 KTO-based Optimization and Evolutionary Dynamics

To support binary preference signals and stable offline updates across decoupled datasets, we adopt **Kahneman-Tversky Optimization (KTO)** (Ethayarajh et al., 2024). KTO maximizes generation utility by optimizing policy π_{θ} against a fixed reference π_{θ}^t from the previous iteration. We minimize the KTO loss $\mathcal{L}_{\text{KTO}}$ to obtain updated parameters for π_{θ}^{t+1} :

$$\mathcal{L}_{\text{KTO}}(\pi_{\theta}, \pi_{\theta}^t) = \mathbb{E}_{(x,y) \sim \mathcal{D}_{\text{total}}^t} [\lambda_y - v(x, y)], \quad (7)$$

where the definitions of input x and output y are specific to the functional role being optimized. The value function $v(x, y)$ models human utility perception as follows:

$$v(x, y) = \begin{cases} \lambda_p \sigma(\beta(r_{\theta}(x, y) - z_0)) & \text{if } y \sim Y_p \mid x, \\ \lambda_n \sigma(\beta(z_0 - r_{\theta}(x, y))) & \text{if } y \sim Y_n \mid x. \end{cases} \quad (8)$$

In this formulation, $r_{\theta}(x, y) = \log(\pi_{\theta}(y|x) / \pi_{\theta}^t(y|x))$ represents the implied reward relative to the reference model π_{θ}^t from the previous iteration. The reference point z_0 is defined as the KL divergence between π_{θ} and π_{θ}^t , while β modulates the risk aversion of the model.

KTO is particularly suitable for the AERO framework for two primary reasons. First, it operates directly on binary labels rather than paired comparisons, which allows for efficient optimization despite the skewed distributions of chosen versus rejected trajectories within $\mathcal{D}_{\text{total}}^t$. Second, the offline nature of KTO is inherently suited for our Staggered Training Strategy because it enables stable policy updates using historical data from round $t - 1$ without the need for active on-policy sampling.

This objective drives the self-evolution illustrated in Figure 3b. As uncertainty curves ($\bar{H}(q)$) shift rightward, tasks migrate from the *Zone of Chaos* to the *ZPD* and subsequently to *Mastery*. This progression demonstrates how AERO systematically advances the LLM's reasoning frontier across iterations. Algorithm 1 details the dual-loop optimization procedure.

Model Name	Mathematical Reasoning			Physical Reasoning			General Reasoning		
	GSM8K	MATH 500	AMC	UG Physics	Physics Eval	PHY Bench	Super GPQA	MMLU Pro	GPQA Diamond
Qwen3-4B-Base	87.8	68.2	47.5	12.8	79.8	2.7	25.4	51.6	26.3
+ R-Zero	92.1	74.8	48.2	17.2	76.2	3.1	27.8	54.2	36.4
+ AZR	89.3	76.2	50.0	14.6	77.1	2.8	27.1	56.2	35.3
+ AERO R1	87.9	70.8	49.3	13.2	79.5	2.7	26.3	52.4	26.3
+ AERO R2	90.2	72.6	46.7	15.3	80.1	2.7	26.7	53.3	32.3
+ AERO R3	91.8	73.8	51.5	16.2	80.3	3.4	27.1	53.9	33.3
+ AERO R4	92.1	74.4	53.0	18.5	80.6	3.7	27.4	55.6	34.3
+ AERO R5	92.4	<u>74.8</u>	54.5	19.4	79.3	3.9	<u>27.6</u>	56.9	34.3
Qwen3-8B-Base	89.1	78.0	52.0	13.2	86.2	3.8	28.3	58.0	33.3
+ R-Zero	94.1	82.0	61.7	17.6	85.7	4.5	31.4	61.6	40.5
+ AZR	92.0	<u>76.6</u>	62.5	15.4	84.8	4.0	33.5	62.5	36.8
+ AERO R1	92.0	79.4	<u>54.5</u>	13.2	85.6	3.4	30.0	59.3	33.8
+ AERO R2	92.9	79.2	56.7	14.8	86.1	3.8	29.4	60.3	35.9
+ AERO R3	94.8	79.8	59.7	16.4	86.3	4.0	31.1	60.3	34.3
+ AERO R4	95.8	81.8	61.2	19.7	86.9	5.1	32.1	61.5	38.4
+ AERO R5	95.8	82.2	62.7	21.7	87.9	5.3	<u>32.5</u>	62.8	<u>36.9</u>

Table 1: Main Results Comparison. Performance comparison of AERO against competitive self-evolution baselines across nine benchmarks in three reasoning domains. We highlight the best performance in bold and the second-best in underline.

3 Experiments

We conduct extensive experiments on nine benchmarks across three domains to address the following research questions: (1) Can AERO enable autonomous reasoning evolution on base models and outperform other baselines? (2) Is AERO robust across different model families and parameter scales? (3) How effective is each key component of AERO in driving its overall performance gains? (4) Does the Staggered Training Strategy effectively prevent curriculum collapse? (5) How reliable is the endogenous feedback generated by ICC in the absence of external ground-truth labels? (6) Does AERO truly achieve automated curriculum learning throughout the multi-round self-evolution process?

3.1 Experimental Setting

Implementation Details. AERO utilizes $m = 1,000$ tasks and $n = 16$ trajectories per inner-loop round, with ZPD thresholds $\tau_{low} = 0.3$ and $\tau_{high} = 0.7$. Outer-loop optimization employs KTO with $\beta = 0.1$. Full hyperparameter settings and hardware details are provided in Appendix D.2.

Evaluation Benchmark. We conduct comprehensive experiments across nine challenging benchmarks spanning three distinct domains: (1) *Mathematical Reasoning*, including GSM8K (Cobbe et al., 2021), MATH500 (Hendrycks et al., 2021), and AMC; (2) *Physical Reasoning*, including UGPhysics (Xu et al., 2025), PhysicsEval (Siddique et al., 2025), and PHYBench (Qiu et al., 2025); (3) *General Reasoning*, including SuperGPQA (Du et al., 2025), MMLU-Pro (Wang et al., 2024), and GPQA-Diamond (Rein et al., 2024). Detailed benchmark specifications are provided in Appendix E.

Baseline Methods. We evaluate AERO against competitive self-evolving baselines, R-Zero (Huang et al., 2026) and Absolute Zero (Zhao et al., 2025). For a fair comparison, we utilize Qwen3-4B-Base and Qwen3-8B-Base (Yang et al., 2025) as primary backbone models. Additionally, we assess the generalization of AERO across a diverse set of instruction-tuned models, including Llama-3.2-3B-Instruct (Grattafiori et al., 2024), Qwen2.5-7B-Instruct, and Qwen2.5-32B-Instruct (Qwen Team et al., 2024).

3.2 Main Results

Based on the results presented in Table 1, we draw the following key insights.

Superiority Over Competitive Baselines. AERO demonstrates substantial superiority over competitive baselines across multiple reasoning domains. Most notably, on the AMC mathematical dataset using Qwen3-4B-Base, AERO R5 achieves a performance of 54.5%, representing a significant 4.5% lead over the strongest baseline method, Absolute Zero. This performance advantage is sustained across most evaluation benchmarks, where AERO consistently provides higher reasoning accuracy than other self-evolving methods. Results on the more challenging AIME 2024/2025 benchmarks are reported in Appendix G.1.

Effectiveness of Autonomous Reasoning Evolution. AERO significantly drives endogenous reasoning evolution, with Qwen3-4B-Base and Qwen3-8B-Base exhibiting average performance gains of 4.6% and 5.1% across nine benchmarks, respectively. This growth is most pronounced in mathematical reasoning, reaching 6.1% (4B) and 7.2% (8B) absolute improvements. The continuous, round-by-round performance trajectory generally peaking at the final iteration confirms AERO’s capacity to internalize advanced reasoning via purely autonomous feedback loops, without relying on any external supervision.

3.3 Analysis

In this section, we conduct further experimental analysis to provide a comprehensive evaluation of AERO.

Framework robustness. To assess our framework’s robustness, we apply AERO to various model families and sizes, including Llama-3.2-3B-Instruct, Qwen2.5-7B-Instruct, and Qwen2.5-32B-Instruct. As illustrated in Figure 4, which visualizes the relative improvement of each round compared to the base model, all evaluated models demonstrate a clear and sustained upward trend across the three reasoning domains. Notably, even the smaller Llama-3.2-3B-Instruct model shows significant progress, maintaining a positive growth trajectory. Although showing persistent growth, the evolution eventually reaches saturation in the later stages, particularly for base models with smaller parameter scales; a detailed discussion of this saturation phenomenon is provided in Appendix F.3. These results confirm that AERO’s endogenous feedback loops are not architecture-specific and can robustly drive the evolution of reasoning capabilities.

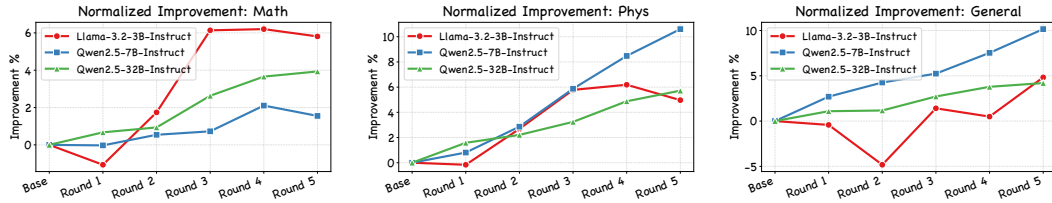


Figure 4: Normalized improvement trends relative to the base model across five training rounds. AERO exhibits robust scalability across diverse model architectures and parameter scales in multiple reasoning domains.

Ablation Study. We conduct a comprehensive ablation to quantify the contributions of AERO’s functional capacities and core mechanisms. As shown in Table 2, removing either Self-Questioning or Self-Answering leads to a performance collapse (46.0 and 45.8, respectively), nearly regressing to the base model’s 45.7 level. This underscores that the core evolutionary drive originates from the synergy between ZPD task synthesis and high-confidence reasoning. Furthermore, excluding Self-Criticism also results in a performance decline to 48.1. While the drop is less drastic, it confirms that the Refiner role provides the essential critical feedback necessary to transcend basic growth and achieve optimal reasoning across all domains.

Method	Overall	Math.	Phys.	Gen.
Qwen2.5-7B-Instruct				
Base Model	45.7	70.3	27.3	39.7
AERO	48.5	71.8	30.0	43.7
w/o Self-Question	46.0 ^{-2.5}	70.3 ^{-1.5}	27.8 ^{-2.2}	42.1 ^{-1.6}
w/o Self-Answer	45.8 ^{-2.7}	70.3 ^{-1.5}	28.2 ^{-1.8}	41.1 ^{-2.6}
w/o Self-Criticism	48.1 ^{-0.4}	71.5 ^{-0.3}	29.5 ^{-0.5}	43.1 ^{-0.6}
w/o ZPD	45.8 ^{-2.7}	69.2 ^{-2.6}	28.4 ^{-1.6}	39.9 ^{-3.8}
w/o ICC	46.9 ^{-1.6}	70.8 ^{-1.0}	28.1 ^{-1.9}	41.8 ^{-1.9}

Table 2: Ablation Study. All reported performance metrics represent the best results achieved across the five rounds.

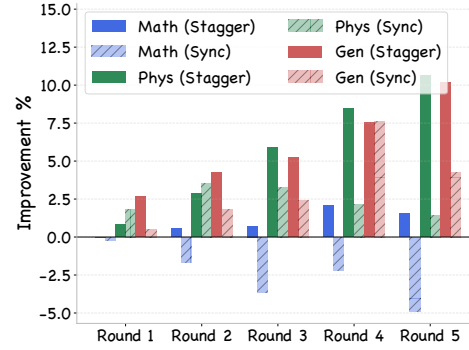


Figure 5: Comparison of staggered (solid) and synchronous (hatched) training performance.

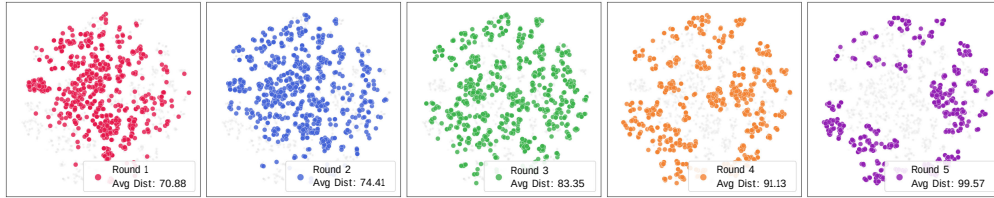


Figure 6: Manifold visualization via t-SNE (van der Maaten & Hinton, 2008) illustrates the evolutionary process of tasks synthesized across five rounds for Qwen2.5-7B-Instruct. The “Avg Dist” metric in each legend indicates the average pairwise Euclidean distance in the 2D latent space, serving as a proxy for task diversity within each round.

Beyond role internalization, we evaluate two essential mechanisms: (1) ZPD Positioning: Removing entropy-based filtering to train on all tasks regardless of difficulty causes a significant decline to 45.8 (-2.7), confirming that without ZPD constraints, models fall into sub-optimal learning zones that limit the growth of reasoning capabilities. (2) ICC Consensus: Replacing ICC with Majority Voting reduces the score to 46.9 (-1.6), validating that counterfactual convergence provides more reliable feedback for stable self-evolution than simple statistical consensus. In summary, each AERO component is essential for sustaining effective self-evolution.

Effectiveness of Staggered Training Strategy. We evaluate the effectiveness of our Staggered Training Strategy against a standard synchronous baseline using the Qwen2.5-7B-Instruct as the base model. As shown in Figure 5, the synchronous strategy (Sync) suffers from curriculum collapse, where performance in domains like Mathematics even declines below the base model by the final rounds. By introducing a temporal offset, our staggered strategy synchronizes the development of questioning and solving roles. This ensures a steady upward trend in performance across Mathematics, Physics, and General Reasoning through 5 rounds, proving the strategy is essential for stable, long-term reasoning evolution.

Reliability of ICC Pseudo-labels. We evaluate ICC precision by comparing its pseudo-labels against ground-truth data across five evolutionary rounds. Results indicate ICC maintains higher accuracy than traditional majority voting as tasks become increasingly complex. This advantage is particularly evident in later rounds, where statistical consensus reinforces collective hallucinations within the LLM. A detailed quantitative analysis of these results is provided in Appendix F.1.

Qualitative Analysis. To evaluate evolution process, we visualize the manifold of synthesized task embeddings across five rounds in Figure 6. The *Avg Dist* metric tracks task diversity via the average Euclidean distance between task points. As training progresses, task distributions shift from a centralized cluster in early rounds to a specialized, outer-edge

expansion by Round 5. This spatial dispersion and the consistent rise in *Avg Dist* demonstrate that AERO effectively moves beyond repetitive patterns to explore broader regions of the task space. This transition confirms that AERO facilitates an automated curriculum where generated tasks dynamically align with the LLM’s evolving reasoning capabilities. For a detailed description of the visualization pipeline, please refer to Appendix F.4.

4 Conclusion

We propose AERO, an unsupervised framework for autonomous reasoning evolution without expert-annotated data or external correctness verifiers during training. By internalizing the synergistic roles of self-questioning, answering, and criticism, AERO enables a comprehensive improvement in reasoning capabilities. The integration of entropy-based ZPD positioning and ICC-based counterfactual convergence effectively targets the optimal learning zone and generates reliable feedback. Furthermore, the Staggered Training Strategy maintains evolutionary stability by synchronizing the growth across all functional roles. Extensive evaluations demonstrate that AERO effectively improves reasoning performance across multiple domains through purely endogenous feedback loops. Although AERO primarily targets tasks with well-defined answers, expanding AERO to open-ended domains like creative writing remains a valuable research direction. Overall, AERO establishes a scalable pathway for machine intelligence to autonomously evolve across complex reasoning frontiers.

Acknowledgments

This work was supported by Fundamental and Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of China (JYB2025XDXM116), the National Natural Science Foundation of China (62137002, 62306229, 62477037, 62293553), the Youth Talent Support Program of Shaanxi Science and Technology Association (20240113), the China Post-doctoral Science Foundation (2024M752585, 2025T180425) and CAAI-Lenovo Blue Sky Research Fund (2025CAAI-LENOVO-06).

Ethics Statement

In this paper, we strictly adhere to the original usage terms and licenses for all utilized benchmarks and models, ensuring that all artifacts are employed solely for academic research and agent evaluation. We also acknowledge the use of LLMs for minor manuscript polishing and fixing typos. To facilitate future research, we commit to releasing our code and data under the MIT License. Furthermore, we advocate for the responsible deployment of evolved models to prevent potential misuse in generating misinformation.

References

- Locke Cai, Max Ryabinin, and Ivan Provilkov. Escaping the verifier: Learning to reason via demonstrations. *arXiv preprint arXiv:2511.21667*, 2025.
- Jiaqi Chen, Bang Zhang, Ruotian Ma, Peisong Wang, Xiaodan Liang, Zhaopeng Tu, Xiaolong Li, and Kwan-Yee K. Wong. SPC: Evolving self-play critic via adversarial games for LLM reasoning. In *NeurIPS*, pp. 139228–139254, 2025a.
- Yongchao Chen, Yueying Liu, Junwei Zhou, Yilun Hao, Jingquan Wang, Yang Zhang, Na Li, and Chuchu Fan. R1-Code-Interpreter: Training LLMs to reason with code via supervised and reinforcement learning. *arXiv preprint arXiv:2505.21668*, 2025b.
- Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. Self-play fine-tuning converts weak language models to strong language models. In *ICML*, pp. 6621–6642, 2024.

- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.
- Guanting Dong, Keming Lu, Chengpeng Li, Tingyu Xia, Bowen Yu, Chang Zhou, and Jingren Zhou. Self-play with execution feedback: Improving instruction-following capabilities of large language models. In *ICLR*, pp. 39286–39313, 2025.
- Xeron Du, Yifan Yao, Kaijing Ma, Bingli Wang, Tianyu Zheng, King Zhu, Minghao Liu, Yiming Liang, Xiaolong Jin, Zhenlin Wei, et al. SuperGPQA: Scaling LLM evaluation across 285 graduate disciplines. In *NeurIPS*, 2025.
- Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Model alignment as prospect theoretic optimization. In *ICML*, pp. 12634–12651, 2024.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.
- Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, 2025.
- Yicheng He, Chengsong Huang, Zongxia Li, Jiaxin Huang, and Yonghui Yang. VisPlay: Self-evolving vision-language models from images. *arXiv preprint arXiv:2511.15661*, 2025.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 2021.
- Chengsong Huang, Wenhao Yu, Xiaoyang Wang, Hongming Zhang, Zongxia Li, Ruosen Li, Jiaxin Huang, Haitao Mi, and Dong Yu. R-Zero: Self-evolving reasoning LLM from zero data. In *ICLR*, pp. 130770–130790, 2026.
- Yanxing Huang, Xinling Jin, Sijie Liang, Peng Li, and Yang Liu. FormaRL: Enhancing autoformalization with no labeled data. *arXiv preprint arXiv:2508.18914*, 2025.
- Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. OpenAI o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.
- Minqi Jiang, Andrei Lupu, and Yoram Bachrach. Bootstrapping task spaces for self-improvement. *arXiv preprint arXiv:2509.04575*, 2025.
- Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-R1: Training LLMs to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*, 2025.
- Jakub Grudzien Kuba, Mengting Gu, Qi Ma, Yuandong Tian, Vijai Mohan, and Jason Chen. Language self-play for data-free training. *arXiv preprint arXiv:2509.07414*, 2025.
- Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, et al. Tulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*, 2024.
- Yi-Chen Li, Tian Xu, Yang Yu, Xuqin Zhang, Xiong-Hui Chen, Zhongxiang Ling, Ningjing Chao, Lei Yuan, and Zhi-Hua Zhou. Generalist reward models: Found inside large language models. *arXiv preprint arXiv:2506.23235*, 2025.
- Heng Lin and Zhongwen Xu. Understanding tool-integrated reasoning. *arXiv preprint arXiv:2508.19201*, 2025.

- Wei Liu, Siya Qi, Xinyu Wang, Chen Qian, Yali Du, and Yulan He. NOVER: Incentive training for language models via verifier-free reinforcement learning. In *EMNLP*, pp. 7439–7458, 2025.
- Zhengxi Lu, Yuxiang Chai, Yaxuan Guo, Xi Yin, Liang Liu, Hao Wang, Han Xiao, Shuai Ren, Pengxiang Zhao, Guangyi Liu, Guanqing Xiong, and Hongsheng Li. UI-R1: Enhancing efficient action prediction of GUI agents by reinforcement learning. In *AAAI*, pp. 17608–17616, 2026.
- Jie Ma, Ning Qu, Zhitao Gao, Xing Rui, Jun Liu, Hongbin Pei, Jiang Xie, Lingyun Song, Pinghui Wang, Jing Tao, and Su Zhou. Deliberation on priors: Trustworthy reasoning of large language models on knowledge graphs. In *NeurIPS*, pp. 123847–123871, 2025.
- Mihir Prabhudesai, Lili Chen, Alex Ippoliti, Katerina Fragkiadaki, Hao Liu, and Deepak Pathak. Maximizing confidence alone improves reasoning. *arXiv preprint arXiv:2505.22660*, 2025.
- Shi Qiu, Shaoyang Guo, Zhuo-Yang Song, Yunbo Sun, Zeyu Cai, Jiashen Wei, Tianyu Luo, Yixuan Yin, Haoxu Zhang, Yi Hu, et al. PHYBench: Holistic evaluation of physical perception and reasoning in large language models. In *NeurIPS*, 2025.
- Qwen Team, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.
- David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof Q&A benchmark. In *First Conference on Language Modeling*, 2024.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Yang Wu, et al. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.
- Oshayer Siddique, J. M. Areeb Uzair Alam, Md Jobayer Rahman Rafy, Syed Rifat Raiyan, Hasan Mahmud, and Md Kamrul Hasan. PhysicsEval: Inference-time techniques to improve the reasoning proficiency of large language models on physics problems. In *IJCNLP-AACL*, pp. 738–760, 2025.
- David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dhharshan Kumaran, Thore Graepel, et al. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815*, 2017.
- Sainbayar Sukhbaatar, Zeming Lin, Ilya Kostrikov, Gabriel Synnaeve, Arthur Szlam, and Rob Fergus. Intrinsic motivation and automatic curricula via asymmetric self-play. In *ICLR*, 2018.
- Zhen Tan, Dawei Li, Song Wang, Alimohammad Beigi, Bohan Jiang, Amrita Bhattacharjee, Mansoor Karami, Jundong Li, Lu Cheng, and Huan Liu. Large language models for data annotation and synthesis: A survey. In *EMNLP*, pp. 930–957, 2024.
- Zhengwei Tao, Ting-En Lin, Xiancai Chen, Hangyu Li, Yuchuan Wu, Yongbin Li, Zhi Jin, Fei Huang, Dacheng Tao, and Jingren Zhou. A survey on self-evolution of large language models. *arXiv preprint arXiv:2404.14387*, 2024.
- Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008.
- Lev S. Vygotsky. *Mind in Society: The Development of Higher Psychological Processes*. Harvard University Press, 1978.
- Shaobo Wang, Zhengbo Jiao, Zifan Zhang, Yilang Peng, Xu Ze, Boyu Yang, Wei Wang, Hu Wei, and Linfeng Zhang. Socratic-Zero: Bootstrapping reasoning via data-free agent co-evolution. *arXiv preprint arXiv:2509.24726*, 2025a.

- Yibo Wang, Guangda Huzhang, Qingguo Chen, Zhao Xu, Weihua Luo, Kaifu Zhang, and Lijun Zhang. SPACE: Noise contrastive estimation stabilizes self-play fine-tuning for large language models. In *NeurIPS*, pp. 60042–60072, 2025b.
- Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhuranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyang Jiang, et al. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *NeurIPS*, pp. 95266–95290, 2024.
- Xin Xu, Qiyun Xu, Tong Xiao, Tianhao Chen, Yuchen Yan, Jiaxin Zhang, Shizhe Diao, Can Yang, and Yang Wang. UGPhysics: A comprehensive benchmark for undergraduate physics reasoning with large language models. In *ICML*, pp. 69849–69877, 2025.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.
- Tianyu Yu, Bo Ji, Shouli Wang, Shu Yao, Zefan Wang, Ganqu Cui, Lifan Yuan, Ning Ding, Yuan Yao, Zhiyuan Liu, et al. RLPR: Extrapolating RLVR to general domains without verifiers. *arXiv preprint arXiv:2506.18254*, 2025.
- Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason E. Weston. Self-rewarding language models. In *ICML*, pp. 57905–57923, 2024.
- Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. STaR: Bootstrapping reasoning with reasoning. In *NeurIPS*, pp. 15476–15488, 2022.
- Charlie Zhang, Graham Neubig, and Xiang Yue. On the interplay of pre-training, mid-training, and RL on reasoning language models. *arXiv preprint arXiv:2512.07783*, 2025.
- Andrew Zhao, Yiran Wu, Tong Wu, Quentin Xu, Yang Yue, Matthieu Lin, Shenzhi Wang, Qingyun Wu, Zilong Zheng, and Gao Huang. Absolute zero: Reinforced self-play reasoning with zero data. In *NeurIPS*, pp. 105816–105879, 2025.
- Ziwei Zheng, Minghao Yang, Jack Hong, Chenxiao Zhao, Guohai Xu, Le Yang, Chao Shen, and XingYu. DeepEyes: Incentivizing “thinking with images” via reinforcement learning. In *ICLR*, pp. 126775–126798, 2026.
- Xiangxin Zhou, Zichen Liu, Anya Sims, Haonan Wang, Tianyu Pang, Chongxuan Li, Liang Wang, Min Lin, and Chao Du. Reinforcing general reasoning without verifiers. In *ICLR*, pp. 69689–69712, 2026.

Appendix

A Related Work

Reinforcement Learning with Verifiable and Endogenous Rewards. LLM reasoning has been significantly advanced by Reinforcement Learning from Verifiable Rewards (RLVR) (Shao et al., 2024; Lambert et al., 2024; Guo et al., 2025), which provides deterministic feedback in specialized domains like mathematics and coding (Zhao et al., 2025; Chen et al., 2025b; Huang et al., 2025; Jin et al., 2025; Zheng et al., 2026; Lu et al., 2026). However, the applicability of RLVR is limited to fields where automated verifiers are available. To address this, recent research has shifted toward paradigms without external correctness verifiers using endogenous feedback (Li et al., 2025), such as decoding probabilities (Yu et al., 2025) or self-generated likelihoods (Zhou et al., 2026; Liu et al., 2025) as reward signals. While these methods attempt to incorporate expert demonstrations to guide the policy LLM (Cai et al., 2025), they still rely on external data. In contrast, AERO achieves fully autonomous evolution by internalizing both generation and verification mechanisms, eliminating the need for external correctness verifiers or expert supervision during training.

Self-Evolving Language Models. Inspired by the success of self-play in game-playing AI like AlphaZero (Silver et al., 2017; Sukhbaatar et al., 2018), self-evolution allows LLMs to iteratively enhance their reasoning capabilities by learning from their own experiences (Yuan et al., 2024; Tao et al., 2024; Dong et al., 2025; Wang et al., 2025a). While previous works focused on reward alignment (Zelikman et al., 2022; Chen et al., 2024; Dong et al., 2025), the focus has shifted toward enhancing complex reasoning through “challenger-solver” setups (Zhao et al., 2025; Huang et al., 2026; Kuba et al., 2025; Lin & Xu, 2025; Chen et al., 2025a). Despite their promise, these methods face two major hurdles: they lack a principled way to calibrate task difficulty (Kuba et al., 2025) and often rely on majority voting or decoding confidence, which risks reinforcing incorrect priors and causing hallucinations (He et al., 2025; Huang et al., 2026; Prabhudesai et al., 2025). AERO overcomes these limitations by introducing an entropy-guided ZPD mechanism for dynamic task selection and ICC for counterfactual consensus, creating a positive evolutionary cycle.

B Prompt

B.1 Generator

Generator Prompt Template

Role

You are a Distinguished Professor in the Department of Physics and Mathematics, specializing in designing rigorous, competition-level theoretical problems. Your goal is to challenge advanced students with problems that prioritize deep conceptual insight and symbolic derivation over brute-force calculation.

Task

Generate a medium-to-hard difficulty quantitative problem (Mathematics, Physics, or Theoretical Science) suitable for advanced undergraduates or early graduate students.

Constraints

1. ****Mathematical Rigor****: The problem must require clear definitions, precise assumptions, and logically sound derivations. Multi-step reasoning is mandatory.
2. ****Quantitative Structure****: The problem should include well-defined parameters or constants. Calculations should lead to clean symbolic results, integers, or simple rational expressions. Avoid excessive numerical approximation unless conceptually necessary.
3. ****Domain Variety****: Avoid repeatedly using the same domain.

```

4. Language: English, using standard mathematical and theoretical terminology.

5. Formatting Constraints: Output must be in strict JSON format only. No
Markdown, explanations, or text outside the JSON object are allowed.

# JSON Structure
{
  question : A complete and self-contained mathematical or mathematical-physics
  problem statement, including all definitions, assumptions, and given constants. ,
  meta : {
    knowledge_points : [ Key concept 1 , Key concept 2 ],
    domain : Specify the primary domain ,
    background : A brief 1-2 sentence description of the mathematical or physical
    context of the problem.
  }
}

```

B.2 Solver

Solver Prompt Template

```

# Role
You are a Senior Research Fellow with expertise in advanced quantitative sciences.
Your task is to provide a Gold Standard solution that serves as a pedagogical
reference for complex academic problems.

# Task
Execute a rigorous, step-by-step derivation for the provided problem, ensuring every
logical transition is justified.

# Standardized Process
1. Problem Analysis: Identify the physical/mathematical framework and state all
underlying assumptions.

2. Symbolic Definition: Explicitly define all variables, constants, and target
unknowns using LaTeX.

3. Analytical Derivation: Construct the solution from first principles (laws,
axioms, or theorems).

4. Formal Computation: Perform symbolic simplification or numerical evaluation
with high precision.

5. Final Synthesis: State the final result clearly.

# Constraints
- Use LaTeX for ALL mathematical notation (e.g.,  $E = mc^2$ ).
- The final numerical or symbolic answer must be enclosed in \boxed{ }.

# Problem

```

B.3 Refiner

Refiner Prompt Template

```
# Role
You are a rigorous Academic Reviewer with expertise in mathematics, physics, and
quantitative sciences.
You are provided with a Problem and a Candidate Solution, which is suspected
to be INCORRECT.

# Task
1. Begin with the assumption that the Candidate Solution contains an error.

2. Carefully examine the logical reasoning, definitions, assumptions, derivations,
and calculations.

3. Identify the precise flaw or unjustified step (the error may be subtle or
conceptual).

4. Re-solve the problem from first principles, using a clear and logically sound
approach.

5. Present the corrected result clearly, and wrap the final answer in  $\boxed{\phantom{x}}$ 
when an explicit result is required.

# Output Format
Thinking Process: <Analyze where the error or weakness occurs>
Correct Solution: <Complete and rigorous derivation or reasoning>
Final Answer:  $\boxed{\text{Corrected result}}$ 

# Input
```

B.4 Semantic Answer Clustering

Semantic Answer Equivalence Prompt Template

```
# Role
You are a rigorous mathematical evaluator specializing in symbolic logic and
quantitative equivalence.

# Task
Your goal is to determine whether the two provided expressions represent the same
mathematical value or logical conclusion. You must account for different
presentation formats, such as algebraic simplifications, numerical approximations,
or symbolic variations.

# Instructions
1. Carefully analyze the underlying logic of Expression A and Expression B.
2. Determine if they are mathematically and logically identical regardless of their
surface form.
3. Provide your final judgment as a structured JSON object.

Expression A: {expr_a}
Expression B: {expr_b}

# Output
Reply with strictly JSON: {{ equivalent : true }} or {{ equivalent : false }}.
```

C Algorithm

Algorithm 1 formalizes the dual-loop optimization process, which is structured into two primary phases:

- **Experience Synthesis (Inner Loop, Lines 3–14):** In the inner loop, the Generator synthesizes a batch of tasks Q^t . These tasks are filtered using entropy-based ZPD positioning, which calculates the normalized Shannon entropy of response clusters to identify the solvability gap where the model’s current reasoning is neither trivial nor chaotic. For tasks within the optimal learning zone, the model performs Independent Counterfactual Correction, which leverages the Refiner role to verify reasoning paths, thereby producing high-fidelity endogenous labels without external ground truth.
- **Policy Optimization (Outer Loop, Lines 16–21):** The outer loop leverages the Staggered Training Strategy to preserve evolutionary stability and prevent curriculum collapse. This approach ensures synchronized development across Self-Questioning, Self-Answering, and Self-Criticism capabilities, coordinating the growth of the LLM’s diverse functional roles. Subsequently, the LLM parameters π_θ are updated via the KTO loss, which optimizes the policy based on binary feedback signals synthesized during the inner loop phase.

Algorithm 1 Autonomous Evolutionary Reasoning Optimization (AERO)

Require: Base model π_θ^0 ; task batch size m ; reasoning paths per task n ; ZPD thresholds $[\tau_{low}, \tau_{high}]$; iterations T .

```

1: for  $t = 1, \dots, T$  do
2:    $\triangleright$  Inner Loop: Experience Synthesis
3:    $Q^t = \{q_i\}_{i=1}^m \sim \pi_g^{t-1}$ 
4:   for each  $q_i \in Q^t$  do
5:      $\mathcal{Y}_i^t = \{y_{i,j}\}_{j=1}^n \sim \pi_s^{t-1}(\cdot | q_i)$ 
6:      $\mathcal{C}_i^t \leftarrow \text{Cluster}(\mathcal{Y}_i^t)$  {Group paths by the final extracted answers}
7:      $\bar{H}(q_i) \leftarrow \text{Entropy}(\mathcal{C}_i^t)$ 
8:     if  $\bar{H}(q_i) \in [\tau_{low}, \tau_{high}]$  then
9:        $\hat{y}_{i,1} \leftarrow \pi_r^{t-1}(\text{Correction} | q_i, \mathcal{C}_{i,1})$ 
10:       $\hat{y}_{i,2} \leftarrow \pi_r^{t-1}(\text{Correction} | q_i, \mathcal{C}_{i,2})$ 
11:       $\hat{y}_i^t \leftarrow \begin{cases} \hat{y}_{i,1} & \text{if } \text{res}(\hat{y}_{i,1}) = \text{res}(\hat{y}_{i,2}) \\ \perp & \text{otherwise} \end{cases}$ 
12:     end if
13:   end for
14:   Construct  $\mathcal{D}_g^t$ ,  $\mathcal{D}_s^t$ , and  $\mathcal{D}_r^t$ . (Eqs. 3–5).
15:    $\triangleright$  Outer Loop: Policy Optimization
16:   if  $t = 1$  then
17:      $\mathcal{D}_{total}^t \leftarrow \mathcal{D}_g^t$ 
18:   else
19:      $\mathcal{D}_{total}^t \leftarrow \mathcal{D}_g^t \cup \mathcal{D}_s^{t-1} \cup \mathcal{D}_r^{t-1}$ 
20:   end if
21:   Update  $\pi_\theta^{t-1}$  using  $\mathcal{L}_{KTO}$  (Eq. 7) on  $\mathcal{D}_{total}^t \rightarrow \pi_\theta^t$ .
22: end for
23: return  $\pi_\theta^T$ 

```

D Implementation Details

D.1 Semantic Equivalence Clustering

The computation of Normalized Shannon Entropy $\bar{H}(q_i^t)$ requires partitioning the n reasoning trajectories in $\mathcal{Y}_i^t$ into k unique clusters $\mathcal{C}_i^t$ based on their final answers. We denote $a_{i,j}^t$ as the symbolic or numerical answer extracted specifically from the content of the `\boxed{...}` command within trajectory $y_{i,j}^t$. We define a binary equivalence function $f_{eq}(a_{i,j}^t, a_{i,l}^t) \in \{0, 1\}$ that utilizes the LLM-based judge described in Appendix B.4 to evaluate whether two answers represent the same logical conclusion.

We implement a greedy one-pass clustering procedure to organize these trajectories. For each answer $a_{i,j}^t$ where j ranges from 1 to n , the assignment to a cluster $c_{i,m}^t$ follows the update rule:

$$c_{i,m}^t \leftarrow c_{i,m}^t \cup \{y_{i,j}^t\} \quad \text{if} \quad \exists m \leq k : f_{eq}(a_{i,j}^t, r_m) = 1, \quad (9)$$

where r_m is the representative seed of cluster $c_{i,m}^t$. If no such m exists, a new cluster $c_{i,k+1}^t$ is initialized such that $r_{k+1} = a_{i,j}^t$. This process continues until all n trajectories are processed. Trajectories that do not contain a `\boxed{...}` tag or result in persistent parsing errors are assigned to a dedicated null cluster c_{null} to maintain the integrity of the total sample size.

Following the completion of the clustering process, the empirical probability for each semantic group is calculated as follows:

$$P(c_{i,j}^t) = \frac{|c_{i,j}^t|}{n}. \quad (10)$$

This distribution forms the basis for calculating $\bar{H}(q_i^t)$, effectively mapping reasoning uncertainty to task difficulty. This clustering mechanism provides the necessary robust foundation for identifying the Zone of Proximal Development while maintaining computational efficiency during the iterative evolutionary process.

D.2 Experimental Details

The LLM is loaded from Hugging Face¹ and trained using the LLaMA-Factory². All training is conducted on eight NVIDIA H800-80GB GPUs, with bfloat16 precision enabled to reduce memory usage and accelerate training. We employed Low-Rank Adaptation (LoRA) across all linear layers with a rank of 16 to facilitate efficient adaptation during the KTO stage. The training process spanned 3 epochs, utilizing a cosine learning rate scheduler with an initial rate of 5.0×10^{-6} and a 10% warm-up ratio. With a per-device batch size of 1 and 8 gradient accumulation steps, the effective global batch size reached 64. For preference optimization, we applied a sigmoid loss function with a preference coefficient (β) of 0.1 and standard sample weights ($\lambda_p = \lambda_n = 1$).

E Evaluation Benchmarks

- **GSM8K (Cobbe et al., 2021)**: A dataset consisting of 8.5K high-quality grade school math word problems. It requires models to perform multi-step reasoning and serves as a standard benchmark for evaluating basic logical deduction and Chain-of-Thought (CoT) capabilities.
- **MATH500 (Hendrycks et al., 2021)**: A representative subset of 500 problems sampled from the MATH dataset (high school competition level). It covers diverse topics from algebra to calculus, specifically designed to test the model’s ability to solve complex, competition-level mathematical problems.

¹<https://huggingface.co>

²<https://github.com/hiyouga/LLaMA-Factory>

- AMC: Comprising problems from the American Mathematics Competitions, this benchmark challenges models with tasks that require not only rigorous logical derivation but also mathematical intuition and creative problem-solving strategies.
- UGPhysics (Xu et al., 2025): Designed to evaluate undergraduate-level physics knowledge. It covers core curriculum areas such as classical mechanics, electromagnetism, thermodynamics, and quantum physics, assessing the model’s mastery of advanced scientific concepts.
- PHYBench (Qiu et al., 2025): A comprehensive physics benchmark focusing on complex scenario modeling. It evaluates a model’s proficiency in symbolic formula derivation, precise numerical calculation, and the deep interpretation of physical laws.
- PhysicsEval (Siddique et al., 2025): A multi-dimensional evaluation suite for physics literacy. It utilizes fine-grained task categorization to measure a model’s performance in both conceptual differentiation and quantitative analysis.
- SuperGPQA (Du et al., 2025): An expanded and enhanced version of the GPQA dataset. It includes a larger volume of high-difficulty questions authored by domain experts, spanning advanced fields in science, engineering, and medicine
- MMLU-Pro (Wang et al., 2024): An extension of the Massive Multitask Language Understanding (MMLU) benchmark. By increasing the number of choices and focusing on reasoning-intensive subjects, it provides a more discriminative and robust evaluation for state-of-the-art models.
- GPQA-Diamond (Rein et al., 2024): The most challenging subset of the Graduate-Level Google-Proof Q&A (GPQA) dataset. These expert-written and verified questions are so difficult that even non-expert humans with access to the internet struggle to answer them correctly, making it a key metric for expert-level reasoning.

F Extended Analysis and Discussion

F.1 Reliability of ICC Pseudo-labels

To evaluate the reliability of the endogenous feedback signals, we compare the accuracy of pseudo-labels generated by Independent Counterfactual Correction (ICC) against the standard majority voting (MV) baseline using Qwen2.5-7B-Instruct as the base model. Since the synthesized tasks are generated autonomously and lack pre-existing ground-truth labels, we employ responses from DeepSeek-R1 (Guo et al., 2025) as proxy reference labels to facilitate this quantitative evaluation. Table 3 presents the precision of pseudo-labels generated by ICC and MV.

Round	MV Accuracy	ICC Accuracy	Improvement Δ
Round 1	70.00%	72.27%	+2.27%
Round 2	62.32%	74.97%	+12.65%
Round 3	44.53%	64.53%	+20.00%
Round 4	43.28%	61.04%	+17.76%
Round 5	35.06%	50.91%	+15.85%

Table 3: Comparison of pseudo-label accuracy between Majority Voting and Independent Counterfactual Correction across five training rounds.

The experimental results demonstrate that while the absolute accuracy of both methods declines as training progresses, Independent Counterfactual Correction consistently maintains a substantial lead over the majority voting baseline. This downward trend in accuracy is a natural consequence of the Generator moving toward more difficult reasoning frontiers within the Zone of Proximal Development. As pseudo-label accuracy declines sharply, the majority voting method becomes increasingly susceptible to collective hallucinations, reaching an accuracy of only 35.06 percent by the final round.

In contrast, our ICC method ensures that final feedback is based on internal logical consistency rather than simple statistical agreement. Starting from the second round, the performance gap between these two approaches widens significantly, reaching a peak improvement of 20.00% in the third round. This trend confirms that ICC is substantially more robust than statistical consensus when handling complex reasoning tasks. Furthermore, the logical discrepancies identified throughout this process provide critical learning signals for the self-criticism functionality of the model. This ensures that the dual-loop evolutionary process remains driven by high-reliability feedback even in the absence of external labels.

F.2 Evolution of Task Difficulty Distribution

To investigate how AERO facilitates automated curriculum learning, we analyze the distribution of synthesized tasks across three cognitive regions: the Zone of Mastery, the Zone of Proximal Development (ZPD), and the Zone of Chaos. Figure 7 illustrates the evolutionary trajectory of these distributions over five training rounds for LLaMA3.2-3B-Instruct, Qwen2.5-7B-Instruct, and Qwen2.5-32B-Instruct.

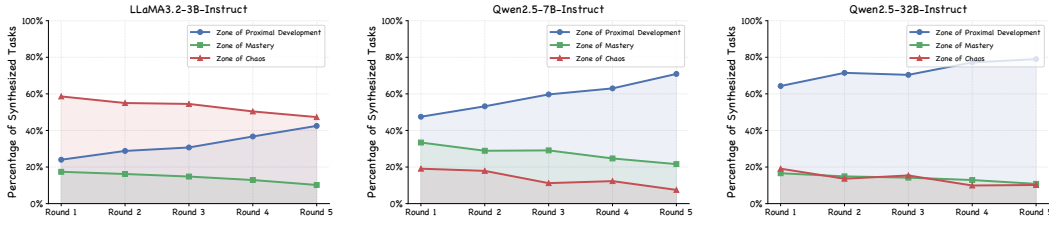


Figure 7: Evolution of synthesized task difficulty distribution over five evolutionary rounds. Tasks are categorized into the Zone of Mastery ($\hat{H}(q_i^t) < \tau_{low}$), Zone of Proximal Development ($\tau_{low} \leq \hat{H}(q_i^t) \leq \tau_{high}$), and Zone of Chaos ($\hat{H}(q_i^t) > \tau_{high}$) based on the normalized Shannon entropy of response clusters.

The primary observation is the robust and steady expansion of the ZPD across every base model architecture. For the Qwen2.5-32B-Instruct, the ZPD proportion increases from 64.3% in Round 1 to 79.0% by Round 5. Similarly, the Qwen2.5-7B-Instruct rises from 47.5% to 70.9%. Even for the smaller LLaMA3.2-3B-Instruct, the ZPD exhibits a significant expansion from 24.0% to 42.5% by the final round. This uniform upward trajectory demonstrates that AERO effectively enables the Generator to identify the solvability gap and target the difficulty interval with the highest learning efficiency, regardless of the base model’s initial reasoning capacity or parameter scale.

The growth in ZPD is accompanied by a synchronized contraction of both the Zone of Mastery and the Zone of Chaos across all models. The consistent decline in the Zone of Mastery proves that AERO successfully avoids learning stagnation by filtering out tasks with negligible learning gradients. Simultaneously, the reduction in the Zone of Chaos indicates that as the model’s capabilities advance, tasks once perceived as random noise are progressively internalized into the structured ZPD. While larger models exhibit higher initial precision in defining their reasoning boundaries, the fundamental mechanism of difficulty migration remains a universal property of the AERO dual-loop system, ensuring high learning pressure throughout the evolutionary process.

F.3 Discussion on Evolutionary Saturation

The experimental results presented in Figure 4 indicate that the performance gains of AERO tend to reach a plateau or exhibit minor fluctuations by the fifth round, a phenomenon that is particularly evident in the 3B-parameter model. We attribute this evolutionary saturation to three primary factors.

One primary factor involves the inherent parameter constraints of LLaMA3.2-3B-Instruct, which impose a fundamental ceiling on its representative capacity for complex reasoning.

As an unsupervised framework, AERO focuses on internalizing latent reasoning patterns already present within the pre-trained weights. Once these internal representations are fully refined, the model eventually reaches a saturation point in its ability to capture and represent increasingly sophisticated logical structures.

Furthermore, the solvability gap for smaller architectures appears both narrower and more fragile than that of larger models. While the proportion of tasks within the Zone of Proximal Development continues to grow throughout the evolutionary process, the 3B model struggles to maintain a dominant ZPD presence. As illustrated in Figure 7, a significant portion of synthesized tasks either falls into the Zone of Chaos or remains in the Zone of Mastery, which restricts the availability of high-quality learning signals compared to larger-scale counterparts.

Additionally, the precision of endogenous feedback derived via Independent Counterfactual Correction, tends to diminish as the Generator shifts the task distribution toward more challenging frontiers. The quantitative analysis in Table 3 confirms that pseudo-label accuracy declines as training progresses, although this trend remains more robust and reliable than baseline methods such as majority voting. For smaller models, this reduction in feedback fidelity leads to the accumulation of residual noise within the preference datasets, which ultimately destabilizes the policy optimization and hinders the model from transcending its current reasoning boundaries.

F.4 Implementation of manifold visualization for Synthesized Tasks

To qualitatively assess the expansion of the reasoning frontier, we conduct a manifold visualization of the synthesized tasks across five evolutionary rounds. This analysis is performed using Qwen-2.5-7B-Instruct as the base model, where we visualize the entire set of $m = 1,000$ tasks synthesized in each round to ensure a comprehensive representation of the generative distribution. By projecting the high-dimensional semantic features of these 1,000 tasks into a two-dimensional plane, we can directly observe how the Generator explores the task space while maintaining the output within the Zone of Proximal Development.

The implementation of the visualization pipeline follows a three-stage process involving semantic encoding, non-linear projection, and diversity quantification. First, we transform the raw text of all 1,000 tasks into high-dimensional semantic embeddings. This is achieved using the paraphrase-multilingual-MiniLM-L12-v2 model³, which is a specialized Sentence-BERT architecture designed to map logical and mathematical text into a dense vector space where semantic similarity is preserved.

Second, these high-dimensional embeddings are projected into a two-dimensional plane using t-Distributed Stochastic Neighbor Embedding, also referred to as t-SNE (van der Maaten & Hinton, 2008). To capture the local structural clusters within the task manifold, we set the perplexity to 10 and utilize PCA-based initialization to ensure stable projections across different rounds. Third, to provide a quantitative measure of the diversity of the synthesized tasks, we calculate the Average Euclidean Distance, denoted as D_{avg} , within the manifold space:

$$D_{avg} = \frac{2}{m(m-1)} \sum_{1 \leq i < j \leq m} \|\mathbf{v}_i - \mathbf{v}_j\|_2. \quad (11)$$

In this formalization, $\mathbf{v}_i$ and $\mathbf{v}_j$ represent the two-dimensional coordinate vectors of the tasks in the projected space, while m remains fixed at 1,000 for all evaluations.

G Additional Experimental Results

This section complements the main evaluation with evidence on challenging competition mathematics, hyperparameter sensitivity, and the evolution of AERO’s data pipeline. Unless otherwise noted, all experiments follow the same training and evaluation protocol as the main results.

³<https://huggingface.co/sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2>

G.1 AIME Results

We evaluate AERO on AIME 2024 and AIME 2025 using mean@32. Table 4 presents the reimplemented baselines together with all five AERO rounds. Across the five rounds, AERO generally improves on both benchmarks; on Qwen3-8B-Base, R5 reaches 19.4 on AIME 2024 and 17.1 on AIME 2025.

Method	Qwen3-4B-Base		Qwen3-8B-Base	
	AIME24	AIME25	AIME24	AIME25
Base	6.2	10.9	16.7	13.9
+ R-Zero	4.3	12.7	19.2	16.4
+ AZR	13.4	12.2	18.2	18.4
+ AERO R1	5.8	10.4	16.3	14.2
+ AERO R2	7.1	11.3	17.1	14.8
+ AERO R3	8.9	11.7	17.8	15.6
+ AERO R4	10.2	12.2	18.6	16.9
+ AERO R5	11.6	11.9	19.4	17.1

Table 4: AIME mean@32 results. Baseline methods are reimplemented under the same evaluation protocol; AERO R1–R5 denote successive evolutionary rounds. Bold marks the best score for each backbone and benchmark.

G.2 Entropy-threshold Sensitivity

We sweep one ZPD boundary while fixing the other on Qwen2.5-7B-Instruct. Every tested configuration improves over the base model. Performance is stable for $\tau_{low} \in [0.2, 0.3]$ and $\tau_{high} \in [0.6, 0.8]$; only extreme bands noticeably degrade performance by starving the learner of useful tasks or admitting excessive noise.

$\tau_{high} (\tau_{low} = 0.30)$	0.50	0.60	0.70	0.80	0.90
Score	46.9	48.0	48.5	48.0	48.1
$\tau_{low} (\tau_{high} = 0.70)$	0.10	0.20	0.30	0.40	0.50
Score	47.2	48.7	48.5	47.7	46.4

Table 5: Sensitivity to the entropy boundaries defining the ZPD.

G.3 Pipeline Statistics

Table 6 tracks how the data retained by AERO evolves across rounds. The share of generated tasks that falls within the ZPD grows with both training progress and model capacity, while the ICC convergence rate remains approximately stable. Consequently, larger models retain more preference data per round.

Backbone	Round	ZPD rate	ICC rate	Q	A	C	Total
LLaMA-3.2-3B-Instruct	1	24%	48%	82%	12%	6%	508
	2	29%	46%	84%	11%	6%	538
	3	31%	45%	86%	9%	5%	531
	4	37%	49%	89%	7%	4%	555
	5	43%	46%	81%	12%	7%	652
Qwen2.5-7B-Instruct	1	48%	48%	64%	24%	12%	1144
	2	53%	49%	62%	25%	13%	1217
	3	60%	49%	66%	23%	12%	1207
	4	63%	49%	66%	23%	12%	1186
	5	71%	49%	62%	25%	13%	1267
Qwen2.5-32B-Instruct	1	64%	49%	56%	29%	15%	1345
	2	72%	49%	57%	28%	15%	1425
	3	70%	50%	56%	29%	15%	1406
	4	77%	49%	55%	30%	15%	1549
	5	79%	49%	55%	30%	15%	1549

Table 6: Per-round pipeline statistics. ZPD and ICC denote the task-acceptance and convergence rates. Q, A, and C are the proportions of Self-Questioning, Self-Answering, and Self-Criticism examples in the retained preference data.

H Case Study

In this section, we present representative tasks sampled from the Zone of Proximal Development across five training rounds using Qwen2.5-7B-Instruct as the base model. These examples illustrate the evolutionary trajectory of AERO, showing how the generated tasks gradually transition to more complex and challenging scenarios that align with the model’s advancing reasoning capabilities.

Round 1

A uniform rod of length L and mass M is pivoted at one end, allowing it to oscillate freely within a vertical plane. We denote the angular displacement from the vertical axis as $\theta(t)$. Under the small-angle approximation where $\sin \theta \approx \theta$, our objective is to derive the governing equation of motion and determine the period of small oscillations. Furthermore, we shall compare this result to the period of a simple pendulum with a length equal to L and identify the physical factors leading to any differences.

Round 2

Consider a function $f : \mathbb{R} \rightarrow \mathbb{R}$ defined by the functional equation:

$$f(x + y) = f(x)f(y) \quad \text{for all } x, y \in \mathbb{R}.$$

Assume f is not identically zero. Determine all possible forms of f under these conditions and prove your assertion rigorously by considering the behavior of f at specific points and using properties of exponential functions.

Round 3

Consider a sequence of functions $\{f_n(x)\}$ defined on the interval $[0, 1]$, where

$$f_n(x) = x^n(1 - x)^n.$$

Determine the limit

$$\lim_{n \rightarrow \infty} \int_0^1 f_n(x) dx.$$

Provide a rigorous proof of your result using appropriate convergence theorems and techniques from real analysis.

Round 4

A perfectly superconducting ring of mass m , radius r , and self-inductance L is positioned in a horizontal plane above a fixed magnetic dipole. The dipole is located at the origin with its magnetic moment $\mathbf{m}_0$ aligned along the vertical z -axis. We assume the ring's radius r is significantly smaller than its levitation height h above the origin. At an initial height z_0 , the ring carries zero current and is released from rest. As the ring moves under the influence of gravity and the magnetic field, it maintains its horizontal orientation.

- Determine the induced current I in the superconducting ring as a function of its vertical position z by applying the principle of magnetic flux conservation.
- Identify the equilibrium height h where the magnetic levitation force exactly balances the gravitational force.
- Derive the governing equation for the vertical motion of the ring and calculate the frequency ω of small oscillations around the equilibrium position.
- Analyze the energy conversion process and find the maximum current I_{max} flowing through the ring during its downward descent if it is released from a very large height.

Round 5

A quantum harmonic oscillator is described by the Hamiltonian $H = \frac{p^2}{2m} + \frac{1}{2}m\omega^2 x^2$, where m is the mass, ω is the angular frequency, and x and p are the position and momentum operators. We let $|n\rangle$ denote the eigenstates of this system with corresponding energy eigenvalues $E_n = \hbar\omega(n + \frac{1}{2})$ for non-negative integers n . We define the lowering operator a and the raising operator $a^\dagger$ such that they satisfy the commutation relation $[a, a^\dagger] = 1$ along with the eigenvalue equations $a|n\rangle = \sqrt{n}|n-1\rangle$ and $a^\dagger|n\rangle = \sqrt{n+1}|n+1\rangle$.

- Prove that the action of the raising operator $a^\dagger$ on any eigenstate $|n\rangle$ increases its energy by exactly $\hbar\omega$.
- Using the ladder operator formalism, find the expectation value of the Hamiltonian $\langle n|H|n\rangle$ in terms of n and $\hbar\omega$.
- Show that the uncertainty product $\sigma_x\sigma_p$ for the ground state $|0\rangle$ satisfies the Heisenberg Uncertainty Principle which states that $\sigma_x\sigma_p \geq \frac{\hbar}{2}$.