Reasoning with Exploration: An Entropy Perspective

Daixuan Cheng[1,4,5†], **Shaohan Huang**[2†], **Xuekai Zhu**[3†], **Bo Dai**[4]
Wayne Xin Zhao[1,5‡], **Zhenliang Zhang**[4‡], **Furu Wei**[2]

[1]Gaoling School of Artificial Intelligence, Renmin University of China
[2]Microsoft Research
[3]Shanghai Jiaotong University
[4]Beijing Institute for General Artificial Intelligence
[5]Beijing Key Laboratory of Research on Large Models and Intelligent Governance

Abstract

Balancing exploration and exploitation is a central goal in reinforcement learning (RL). Despite recent advances in enhancing language model (LM) reasoning, most methods lean toward exploitation, and increasingly encounter performance plateaus. In this work, we revisit entropy—a signal of exploration in RL—and examine its relationship to exploratory reasoning in LMs. Through empirical analysis, we uncover positive correlations between high-entropy regions and three types of exploratory actions: (1) pivotal tokens that determine or connect logical steps, (2) reflective actions such as self-verification and correction, and (3) rare behaviors underexplored by the base LMs. Motivated by this, we introduce a minimal modification to standard RL with only one line of code: augmenting the advantage function with an entropy-based term. Unlike traditional maximum-entropy methods which encourage exploration by promoting uncertainty, we encourage exploration by promoting deeper and longer reasoning chains. Notably, our method achieves significant gains on the Pass@K metric—an upper-bound estimator of reasoning capabilities—even when evaluated with extremely large K values, pushing the boundaries of LM reasoning.

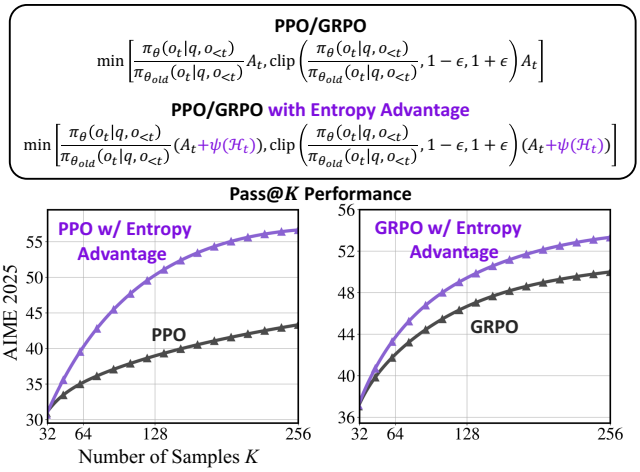

Figure 1: We augment the advantage in PPO or GRPO with a minimal entropy-based term (Top). Our entropy-based advantage effectively encourages exploratory reasoning in LMs, achieving superior Pass@K performance (Bottom).

1 Introduction

Recent reinforcement learning methods for language models (LMs), particularly those using verifiable rewards (RLVR; Lambert et al. 2024), typically rely on signals that reflect output accuracy to guide training. These approaches have proven effective in enhancing reasoning by reinforcing correct outputs and discouraging incorrect ones (Guo et al. 2025). However, as training progresses under purely accuracy-driven objectives, these benefits often diminish. LMs gradually lose the incentive to take exploratory actions for sustained, multi-step reasoning, leading to performance plateaus or even regression (Yu et al. 2025; Cui et al. 2025b).

In traditional RL, exploration plays a vital role alongside exploitation by encouraging the policy model to explore alternative strategies (Ladosz et al. 2022). A common signal for exploration is entropy, which quantifies uncertainty in the policy's action distribution (Haarnoja et al. 2018; Ziebart et al. 2008). Motivated by this, we investigate the relationship between entropy and exploratory reasoning in LMs, and uncover positive correlations: (1) Pivotal tokens that guide or connect reasoning steps (e.g., `first`, `because`, and `however`) exhibit higher entropy; (2) Reflective actions such as self-verification and correction, tend to emerge in high-entropy regions; (3) During RL training, rare solutions also coincide with elevated entropy. Together, these findings suggest entropy can be a valuable signal for recognizing exploratory reasoning behaviors of LMs.

Based on these findings, we propose incorporating entropy as an auxiliary term to encourage exploratory reasoning of LMs. While traditional maximum entropy methods encourage exploration by promoting uncertainty (O'Donoghue et al. 2016), our method takes a different path to balance exploration and exploitation: we introduce a clipped, gradient-detached entropy term into the advantage function of standard RL algorithms. Clipping ensures that the entropy term neither dominates nor reverses

[†]Core Contributors.
[‡]Corresponding Authors.

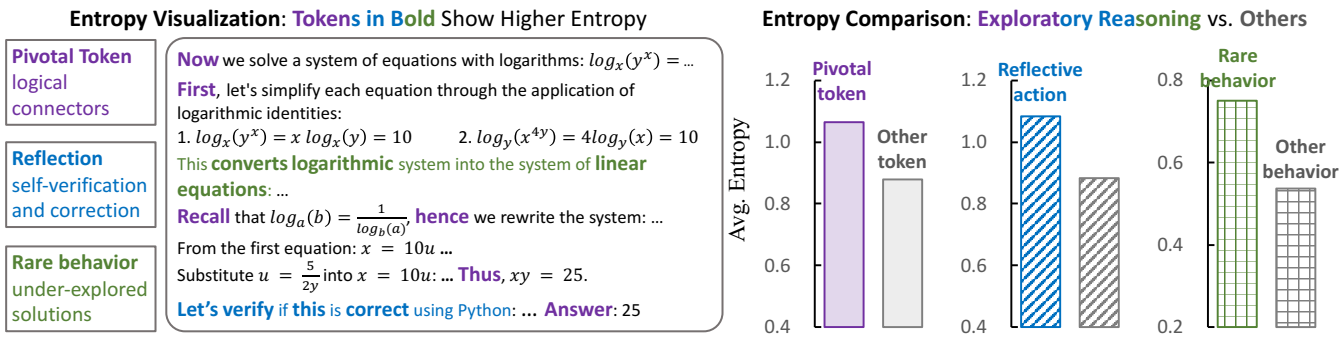

Figure 2: Entropy Visualization and Comparison between Exploratory Reasoning and Others.

the sign of the original advantage, while gradient detachment preserves the original optimization direction. This amplifies exploratory reasoning behaviors that emerge under uncertainty while maintaining the original policy gradient flow. Moreover, because of the intrinsic tension between entropy and confidence, the entropy-based term naturally diminishes as confidence increases—encouraging exploration in early stages while avoiding over-encouragement as training progresses. Furthermore, our method is extremely simple, requiring only one line of code to seamlessly integrate into existing RLVR training pipelines (Sheng et al. 2024).

We validate our method on mainstream RLVR algorithms, GRPO (Shao et al. 2024) and PPO (Schulman et al. 2017), and observe distinct benefits. First, it amplifies exploratory reasoning behaviors—such as the use of pivotal tokens and reflective actions—by decreasing the policy's uncertainty at these decision points. Second, it encourages the generation of longer, more exploratory responses without increasing the repetition rate, enabling coherent multi-step reasoning. Consequently, our method consistently improves Pass@1 accuracy across different LMs. We further increase the number of attempts K per question to evaluate Pass@K—a metric recently regarded as an upper-bound estimator of reasoning capability (Yue et al. 2025a). As shown in Figure 1, our method achieves substantial improvements even at large K, pushing the boundaries of LM reasoning.

In summary, the contributions of this work are as follows:

- We reveal a positive correlation between entropy and exploratory reasoning, showing that pivotal tokens, reflective actions, and rare behaviors emerge with higher entropy.
- We propose a minimal yet effective method that augments the standard RL advantage with a clipped, gradient-detached entropy term, encouraging exploration while preserving the original policy optimization direction.
- We validate our method on mainstream RLVR algorithms: GRPO and PPO, achieving consistent gains on Pass@1 and substantial improvements on Pass@K.

2 Preliminary Analysis: Entropy and Exploratory Reasoning

We examine the relationship between *entropy*, a signal of exploration in RL (Ladosz et al. 2022), and exploratory reasoning of LMs. We start by visualizing token-level entropy in responses from Qwen2.5-Base (Yang et al. 2024a) on mathematical reasoning tasks. As shown in Figure 2, high-entropy tokens exhibit different reasoning dynamics compared to low-entropy ones. Based on this observation, we categorize exploratory reasoning-related content at both token and sentence levels to support the following analysis.

Pivotal Tokens Figure 2 shows that pivotal reasoning tokens, which serve as logical connectors at key decision points (e.g., `first`, `recall`, `thus`), tend to have higher entropy. To quantify this observation, we compute the average entropy of commonly used pivotal tokens, such as causal terms (`because`, `therefore`), contrastive markers (`however`, `although`), sequential indicators (`first`, `then`), and reasoning verbs (`suggest`, `demonstrate`), across all responses and compare it with that of other tokens. The results in the right panel of Figure 2 confirm an increase in entropy. Similar observations have also been noted in concurrent works (Wang et al. 2025a; Qian et al. 2025).

Reflective Actions Reflection is a form of meta-cognition that examines generated information, evaluates underlying reasoning, and adapts future behavior accordingly (Shah et al. 2025). As shown in Figure 2, the LM assigns higher entropy to sentences such as "Let's verify if this is correct...". To quantify this behavior, we segment responses into sentences, compute the average entropy for each, and use regular expressions to identify reflective actions, such as sentences containing keywords like "verify" or "check". These reflective sentences exhibit higher average entropy, suggesting that reflection tends to occur under greater uncertainty.

Rare Behaviors Emergent During RL We examine whether under-explored behaviors, which are rarely exhibited by the base model, show distinct entropy patterns during RL. As shown in Figure 2, one such example is converting logarithmic systems into systems of linear equations. To quantify rarity, we embed all responses with SBERT (Reimers and Gurevych 2019) and compute each RL-generated sentence's average distance to its $k = 5$ nearest neighbors among base model outputs. The top 10% by this distance are labeled as rare. These rare behaviors exhibit higher entropy, revealing a correlation between semantic novelty and predictive uncertainty.

3 Method

Our analysis reveals a positive correlation between entropy and exploratory reasoning in LMs, motivating us to incorporate entropy as an auxiliary term to encourage exploration in RL. To this end, we propose an advantage shaping method that augments the standard RL advantage with an entropy-based term, acting as a robust, self-regulating guide without altering the gradient flow of the base RL algorithm.

Our method is compatible with mainstream RLVR algorithms such as Proximal Policy Optimization (PPO; Schulman et al. 2017) and Group Relative Policy Optimization (GRPO; Shao et al. 2024). We first briefly review these algorithms and then present our advantage shaping method.

RL Baselines: PPO and GRPO

PPO Let q denote a question sampled from a dataset $\mathcal{D}$, and let $o = (o_1, o_2, \ldots, o_{|o|})$ be the corresponding output response generated by a policy model π_θ. PPO optimizes the policy by maximizing a clipped surrogate objective:

$$\mathcal{J}_{\text{PPO}}(\theta) = \mathbb{E}\Big[\min\left(\rho_t(\theta)A_t, \text{clip}(\rho_t(\theta), 1-\epsilon_{\text{L}}, 1+\epsilon_{\text{H}})A_t\right)\Big], \tag{1}$$

where $\rho_t(\theta) = \frac{\pi_\theta(o_t|q,o_{<t})}{\pi_{\theta_{\text{old}}}(o_t|q,o_{<t})}$ denotes the likelihood ratio between the current and old policy models, and A_t is the advantage typically computed using GAE (Schulman et al. 2015). We omit the expectation index to simplify notation. The clipping range ϵ_{L} and ϵ_{H} prevents excessively large changes. While standard PPO uses symmetric clipping (i.e., $\epsilon_{\text{L}} = \epsilon_{\text{H}}$), recent work slightly increases ϵ_{H} to avoid entropy collapse (Yu et al. 2025). The gradient of the PPO objective is (we omit $\min$ and clip operations under the single-update-per-rollout assumption (Shao et al. 2024)):

$$\nabla_\theta \mathcal{J}_{\text{PPO}}(\theta) = \mathbb{E}\left[A_t \nabla_\theta \log \pi_\theta(o_t \mid q, o_{<t})\right]. \tag{2}$$

GRPO GRPO is an alternative to GAE-based PPO that avoids learning a separate value function by using the average reward of multiple sampled outputs, produced in response to the same question, as the baseline. For each question q, a group of G outputs $\{o_1, o_2, \ldots, o_G\}$ is sampled from the old policy $\pi_{\theta_{\text{old}}}$, a reward model is then used to score the outputs, yielding G rewards $\{r_1, r_2, \ldots, r_G\}$ correspondingly. These scores are then normalized as:

$$\tilde{r}_i = \frac{r_i - \text{mean}(\{r_1, r_2, \ldots, r_G\})}{\text{std}(\{r_1, r_2, \ldots, r_G\})}. \tag{3}$$

In outcome-supervised settings, the normalized reward is assigned at the end of each output o_i, and every token in o_i receives the same advantage, i.e., $A_{i,t} = \tilde{r}_i$. The policy is then optimized using the PPO objective in Equation 2.

Entropy-Based Advantage Shaping

To encourage exploratory reasoning, we propose an entropy-based advantage shaping method. The key idea is to inject an entropy-based term into the advantage function.

For each token o_t in an output o, the entropy of the current policy over the vocabulary $\mathcal{V}$ is:

$$\mathcal{H}_t = -\sum_{v \in \mathcal{V}} \pi_\theta(v \mid q, o_{<t}) \log \pi_\theta(v \mid q, o_{<t}). \tag{4}$$

Listing 1: PyTorch Implementation

```
# Compute advantages as in PPO or GRPO
adv = compute_advantages(...)

# Apply entropy-based advantage shaping
adv += torch.min(alpha * entropy.detach
    (), adv.abs() / kappa)

# Compute policy loss as in PPO or GRPO
loss = compute_policy_loss(adv, ...)
```

We then define an entropy-based advantage term $\psi(\mathcal{H}_t)$ and use it to shape the advantage:

$$\psi(\mathcal{H}_t) = \min\left(\alpha \cdot \mathcal{H}_t^{\text{detach}}, \frac{|A_t|}{\kappa}\right), \tag{5}$$

$$A_t^{\text{shaped}} = A_t + \psi(\mathcal{H}_t). \tag{6}$$

Here, α is a positive scaling coefficient ($\alpha > 0$), and κ controls the clipping threshold ($\kappa > 1$). To balance the entropy-based term with the original advantage, we (i) **detach the entropy term** $\mathcal{H}_t^{\text{detach}}$ from the computational graph, so it acts as a fixed offset rather than introducing new gradient paths, and (ii) **apply a clipping operation** to enforce $\psi(\mathcal{H}_t) \leq \frac{|A_t|}{\kappa}$, preventing the entropy-based term from dominating the original advantage. Moreover, when $A_t < 0$, this ensures that the shaped advantage does not flip its sign. Therefore, **our method preserves the update direction, ensuring harmful actions with negative advantages remain penalized**.

As a result, the policy gradient retains a form similar to PPO in Equation 2, with the only difference being that the original advantage A_t is replaced by the shaped advantage:

$$\nabla_\theta \mathcal{J}_{\text{PPO}}^{\text{shaped}}(\theta) = \mathbb{E}\left[(A_t + \psi(\mathcal{H}_t))\nabla_\theta \log \pi_\theta(o_t \mid q, o_{<t})\right]. \tag{7}$$

Practical Implementation Our method can be integrated into existing RL training pipelines with just one line of code. Specifically, after computing the advantages with PPO or GRPO, we add the entropy-based advantage term before calculating the policy loss, as shown in Listing 1[1]. Moreover, our method requires minimal hyper-parameter tuning: a fixed set of hyper-parameter values consistently improve performance across diverse scenarios (Section 5).

Robustness of Entropy-Based Advantage: Avoiding Over-Encouragement Prior work (Chen et al. 2025) attempts to enhance reasoning by rewarding the policy based on the frequency of reasoning-like tokens (e.g., `wait` and `verify`), but this leads to reward hacking—the policy model repeatedly generates such tokens to exploit the reward without performing genuine reasoning. In contrast, our method naturally avoids such over-encouragement due to the intrinsic tension between entropy and confidence. As shown in Figure 3, our method initially assigns high advantage to

[1]In the veRL codebase (Sheng et al. 2024), this corresponds to a one-line code insertion in the `update_policy` function of `verl/workers/actor/dp_actor.py`.

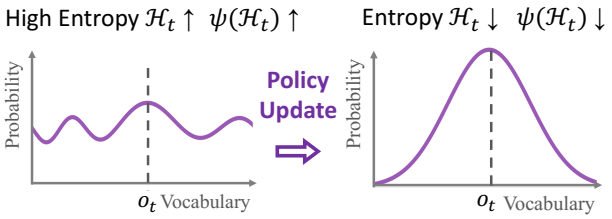

Figure 3: Dynamics of Entropy-Based Advantage. High entropy initially largely amplifies the advantage, accelerating confidence gain and leading to reduced entropy-based shaping in subsequent steps.

	Entropy Regularization	**Entropy Adv.**
Objective	$\mathcal{J}_{\text{PPO}}(\theta) + \beta\,\mathbb{E}\left[\mathcal{H}_t\right]$	$\mathcal{J}_{\text{PPO}}^{\text{shaped}}(\theta)$
Gradient	$\nabla_\theta \mathcal{J}_{\text{PPO}}(\theta) + \beta\,\mathbb{E}\left[\nabla_\theta \mathcal{H}_t\right]$	$\nabla_\theta \mathcal{J}_{\text{PPO}}^{\text{shaped}}(\theta)$

Table 1: Comparison of Entropy Regularization and Entropy-Based Advantage Shaping. We simplify expressions by omitting expectation indices. $\mathcal{J}_{\text{PPO}}^{\text{shaped}}(\theta)$ denotes the PPO objective with shaped advantages.

tokens with high-entropy distributions but gradually reduces the entropy-based advantage as model confidence increases over training iterations.

Formally, let k denote the training iteration and t denote the token position within the output response. The policy model parameters are updated via gradient ascent:

$$\theta_{k+1} = \theta_k + \eta\,\nabla_\theta \mathcal{J}_{\text{PPO}}^{\text{shaped}}(\theta_k), \tag{8}$$

where η is the learning rate, and the policy gradient $\mathcal{J}_{\text{PPO}}^{\text{shaped}}(\theta_k)$ (Equation 7) uses the shaped advantage $A_{k,t}^{\text{shaped}} = A_{k,t} + \psi(\mathcal{H}_{k,t})$, which is positively correlated with the detached entropy $\mathcal{H}_{k,t}^{\text{detach}}$ (Equation 5). When the original advantage $A_{k,t} > 0$, higher entropy leads to a stronger update on the selected token o_t, largely increasing its likelihood $\pi_\theta(o_t \mid \cdot)$ and thus sharpening the output distribution. According to the entropy definition (Equation 4), a sharper distribution lowers entropy, which in turn reduces the entropy-based advantage $\psi(\mathcal{H}_t)$ and weakens subsequent updates. This self-regulating effect is empirically validated by the entropy advantage ratio in Section 7.

Comparison with Entropy Regularization In traditional RL, entropy regularization is commonly adopted to prevent the policy from becoming overly deterministic (Ladosz et al. 2022). Practically, this means adding an entropy loss term to the policy loss. To clarify the difference with our method, we present a comparison in Table 1.

Entropy regularization adds an entropy term $\mathbb{E}[\mathcal{H}_t]$ to the objective. Since $\mathcal{H}_t$ depends on the current policy π_θ, this introduces an additional gradient $\nabla_\theta \mathcal{H}_t$, promoting uncertainty during training. In contrast, our method modifies the advantage by adding a clipped entropy term $\mathcal{H}_t^{\text{detach}}$, which is detached from the computation graph. Thus, $\nabla_\theta \mathcal{H}_t^{\text{detach}} = 0$, and the entropy affects optimization only via the shaped advantages. This makes our method fundamentally distinct and orthogonal to entropy regularization. We also report an empirical comparison of the two methods' effects on RL-trained model performance in Section 7.

4 Experiment Settings

We conduct extensive experiments across diverse backbone LMs and RL algorithms. Unless otherwise specified, we adopt Qwen2.5-Base-7B as the backbone and GRPO as the RL algorithm.

Backbone LMs and Datasets Our training data is sourced from DAPO (Yu et al. 2025). The backbone LMs include Qwen2.5-Base-7B, Qwen2.5-Math-Base-7B (Yang et al. 2024b), and a domain-adapted variant of Llama-3.2-Base-3B (AI@Meta 2024) further pre-trained on mathematical corpora (Wang et al. 2025c). We adopt the domain variant because the DAPO dataset is too challenging for the original Llama-3.2-Base-3B, where RL training with vanilla GRPO or PPO fails to improve performance (Gandhi et al. 2025).

RL Training Configuration The RL baseline algorithms include GRPO and PPO. To build strong baselines, we incorporate several techniques from DAPO and VAPO (Yue et al. 2025b), including *Clip-Higher*, *Token-level Loss*, *Critic Pre-training*, and *Group Sampling*. Building on these RL baselines, we apply our advantage shaping method. We use a fixed set of hyperparameter values throughout all experiments: $\kappa = 2$, $\alpha = 0.4$ for GRPO, and $\alpha = 0.1$ for PPO.

Evaluation Benchmarks and Metrics We evaluate on AIME 2025/2024 (MAA 2025), AMC 2023 (MAA 2023), and MATH500 (Hendrycks et al. 2021). We report both Pass@1 (averaged over 16 runs) and Pass@K (using the unbiased estimator in (Chen et al. 2021; Bai et al. 2025)), with K scaled based on benchmark difficulty and size—larger K for small and challenging benchmark.

5 Main Results

As shown in Table 2, our method consistently outperforms baselines in Pass@1 across different LMs and RL algorithms, and also surpasses strong existing approaches (Cui et al. 2025a; Liu et al. 2025; Chu et al. 2025). This improvement also holds for Pass@K: our method continues to show performance gains even at large K values, where most baselines plateau.

On AIME2024, AMC2023, and MATH500, we observe a phenomenon also noted by Yue et al. (2025a): while RL-trained models improve Pass@1, their performance in Pass@K can fall behind the backbone LMs when K is large, suggesting that vanilla RL may restrict reasoning capacity. Our method mitigates this issue. Notably, on AIME2025—the most challenging benchmark released after the base model's training data cutoff—our method not only outperforms RL baselines but also surpasses the backbone LMs' performance ceiling, highlighting its potential to extend LM reasoning capabilities.

	AIME25		AIME24		AMC23		MATH500	
	Pass@256	*Pass@1*	*Pass@256*	*Pass@1*	*Pass@128*	*Pass@1*	*Pass@16*	*Pass@1*
Qwen2.5-Base-7B	50.0	2.2	66.7	5.2	90.4	28.3	88.8	54.4
+ GRPO	50.0	10.7	46.7	11.9	91.6	55.6	65.4	55.3
+ GRPO w/ Entropy Adv.	53.3	11.8	56.7	12.6	91.6	57.8	74.0	58.5
Δ	+3.3	+1.1	+10.0	+0.7	+0.0	+2.2	+8.6	+3.2
+ PPO	43.3	7.9	46.7	14.2	85.5	51.8	68.4	57.9
+ PPO w/ Entropy Adv.	56.7	11.7	50.0	16.8	88.0	56.1	75.2	60.9
Δ	+13.4	+3.8	+3.3	+2.6	+2.5	+4.3	+6.8	+3.0
Llama-3.2-3B	16.7	0.1	20.0	0.2	73.5	5.1	50.4	14.0
+ GRPO	13.3	0.2	16.7	0.2	69.9	6.6	56.6	18.1
+ GRPO w/ Entropy Adv.	16.7	0.2	23.3	0.3	65.1	6.3	56.6	18.6
Δ	+3.3	+0.0	+6.7	+0.1	-4.8	-0.3	+0.0	+0.5
+ PPO	16.7	0.1	16.7	0.2	61.4	8.8	55.6	20.7
+ PPO w/ Entropy Adv.	23.3	0.5	26.7	0.4	60.2	9.6	57.4	22.6
Δ	+6.7	+0.4	+10.0	+0.2	-1.2	+0.9	+1.8	+1.9
Qwen2.5-Math-7B	50.7	4.4	70.0	10.7	90.0	34.4	88.6	47.5
Eurus-2-PRIME	-	-	-	26.7	-	57.8	-	79.2
Oat-Zero†	-	-	-	30.0	-	55.4	-	80.6
GPG†	-	-	-	33.3	-	65.0	-	80.0
+ GRPO	57.4	16.3	83.3	30.9	92.8	66.9	94.6	83.0
+ GRPO w/ Entropy Adv.	63.6	17.6	80.0	33.7	95.2	69.8	94.8	83.1
Δ	+6.2	+1.3	-3.3	+2.8	+2.4	+2.9	+0.2	+ 0.1

Table 2: Main Results. †: results from (Chu et al. 2025). "+ GRPO" and "+ PPO" indicate RL training from the backbone LMs, while "w/ Entropy Adv." denotes applying our entropy-based advantage to the corresponding RL algorithms. Δ denotes the performance difference between without and with applying our method.

Clip	Clipped Fraction		Eval Performance	
	Initial	*Final*	*Avg.*	*Win Rate*
✗	-	-	48.5	5/8
✓	84%	38%	52.0	**8/8**

	α	Entropy Adv. Ratio		Eval Performance	
		Initial	*Final*	*Avg.*	*Win Rate*
GRPO	0.1	4.1%	1.90%	52.1	6/8
	0.4	10.6%	3.2%	52.0	**8/8**
PPO	0.1	9.5%	2.00%	51.9	**8/8**
	0.4	25.0%	6.0%	47.1	4/8

Table 3: Ablations on the clip operation and coefficient α. We report clipped fractions, entropy ratios at the initial and final stages of RL training, and final model performance. *Avg.* is the mean score across benchmarks; *Win Rate* is the ratio of benchmark scores where our method outperforms the RL baseline.

6 Ablations

We conduct ablations to analyze the impact of the clip operation and the scaling coefficient α, both aimed at balancing the entropy-based term with the original advantage.

Clip Operation The top part of Table 3 shows the effect of the clip operation, where the clipped fraction denotes the fraction of entropy-based terms that are clipped, i.e., where $\alpha \cdot \mathcal{H}_t^{\text{detach}} > \frac{|A_t|}{\kappa}$. Our clipping effectively ensures that the entropy-based term does not dominate or reverse the original advantage, particularly during early training stages when model entropy is typically high. Its effectiveness is also validated by the improved evaluation performance.

Coefficient α We select α to keep the ratio of the entropy-based advantage to the original advantage (i.e., $\frac{\psi(\mathcal{H}_t)}{|A_t|}$) within a moderate range. As shown in Table 3, $\alpha = 0.4$ for GRPO and $\alpha = 0.1$ for PPO yield similar trends in the entropy-advantage ratio: around 10% initially and 3% in the final stages, ensuring the ratio remains moderate for both PPO and GRPO. This is also validated by the evaluation results, with consistent improvements in win rate.

7 Analysis

We analyze key metrics throughout RL training and reasoning dynamics at test time. In addition, we provide a comprehensive comparison with entropy regularization.

RL Training Process

Figure 4 shows reward, response length, entropy, and our entropy-based advantage, throughout RL-training process.

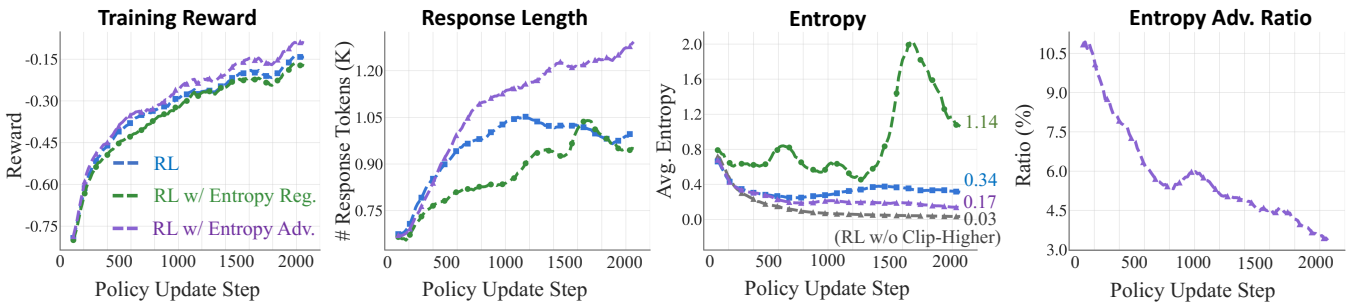

Figure 4: Metrics during RL training. The RL baseline is GRPO; "RL w/ Entropy Reg." applies entropy regularization; "RL w/ Entropy Adv." applies entropy-based advantage shaping; "RL w/o Clip-Higher" removes the clip-higher technique from the RL baseline (i.e., $\epsilon_{\text{H}} = \epsilon_{\text{L}} = 0.2$). "Entropy Adv. Ratio" denotes the ratio of entropy-based advantage to the original advantage.

	AIME25		AIME24		AMC23		MATH500	
	Pass@256	*Pass@1*	*Pass@256*	*Pass@1*	*Pass@128*	*Pass@1*	*Pass@16*	*Pass@1*
RL w/ Entropy Reg.	50.0	9.3	50.0	**16.0**	90.4	54.3	70.4	57.4
RL w/ Entropy Adv.	**53.3**	**11.8**	**56.7**	12.6	**91.6**	**57.8**	**74.0**	**58.5**

Table 4: Comparison of model performance trained with RL using entropy regularization vs. entropy-based advantage shaping.

Training Reward and Response Length RL with entropy-based advantage yields slightly higher rewards in the later stages of training, suggesting more sustained learning effect. While the response length of the RL baseline slightly declines after step 1000, adding our entropy-based advantage maintains the upward trend throughout training. This alignment between response length and reward progression may reflect enhanced reasoning, where the LM generates more tokens to explore and reach correct answers (Guo et al. 2025; Jiang et al. 2025).

Overall Entropy Compared to the RL baseline, entropy regularization increases overall entropy but induces a sharp spike, indicating instability. In contrast, our method maintains the overall entropy at a similar magnitude with slightly lower values, while enabling more sustained reward improvements. Its superiority is further supported by better test-time benchmark performance, as shown in Table 4.

Our method does not aim to increase token entropy uniformly. Instead, it promotes exploratory reasoning behaviors by amplifying high-entropy actions and gradually builds confidence (reflected by lower entropy) at these points. We further analyze this in the next subsection on exploratory reasoning dynamics.

We also compare with RL without *clip-higher*, which causes entropy to collapse to 0.03—a level typically regarded as entropy collapse (Yu et al. 2025; Cui et al. 2025b). In contrast, our method stabilizes entropy at 0.17, nearly six times higher, indicating a balanced range that avoids both collapse and excess.

Entropy-Based Advantage As shown by the trend of the entropy-based advantage ratio (i.e., $\frac{\psi(\mathcal{H}_t)}{|A_t|}$), this ratio gradually decreases as training progresses and the model gains confidence. This supports our hypothesis that the intrinsic tension between model confidence and entropy naturally drives exploration in uncertain regions, while the reduction of the entropy-based advantage over time helps prevent over-encouragement once sufficient confidence is achieved.

Exploratory Reasoning Dynamics

We further analyze the reasoning dynamics of the RL-trained models on the testing benchmarks in Figure 5.

Pivotal Tokens and Reflective Actions Applying our entropy-based advantage reinforces the model's ability to generate pivotal reasoning tokens and reflective actions. These regions exhibit lower entropy, indicating increased model confidence when producing such actions. The increased occurrence of these reasoning patterns, together with improvements on evaluation benchmarks, suggests that the model learns to master higher-level reasoning behaviors, enabling deeper and more effective reasoning chains.

Response Length and Repetition Rate We also observe an increase in response length on the testing benchmark. Additionally, we record the n-gram-based repetition rate of generated responses and find that our method yields much longer responses while maintaining a repetition rate comparable to that of the RL baseline, demonstrating its ability to scale effectively at test time without increasing redundancy.

Case Study As shown in the example responses in Figure 6, our method produces more structured and persistent reasoning compared to the baseline. The model explicitly identifies problem constraints, performs case analysis (e.g., odd vs. even list lengths), and adapts its approach when initial attempts fail. It also verifies constraint satisfaction while iterating over candidate values (e.g., $n = 5, 6, \ldots$), demonstrating stronger reasoning flexibility and control.

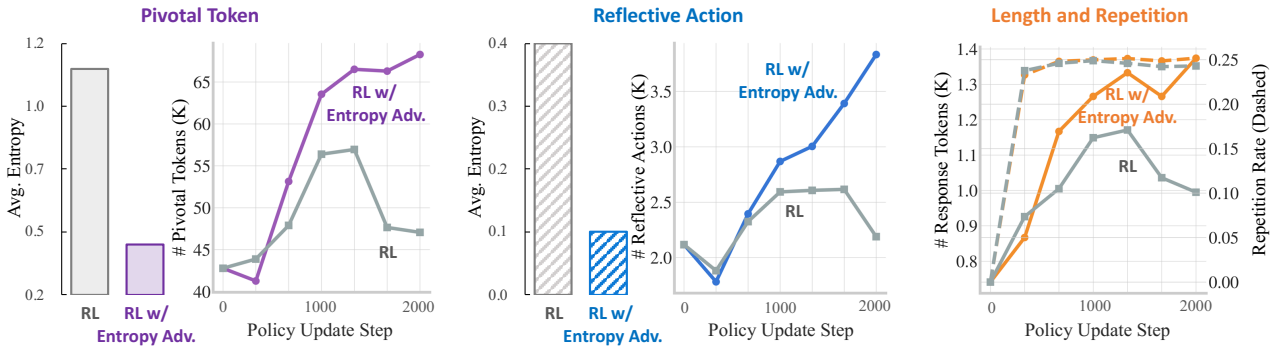

Figure 5: Comparison of reasoning dynamics on the testing benchmarks between the RL baseline and RL with entropy adv.

Question and Ground Truth

Question: A list of positive integers has the following properties: (1) The sum of the items in the list is 30; (2) The unique mode of the list is 9; (3) The median of the list is a positive integer that does not appear in the list itself. Find the sum of the squares of all the items in the list.
Ground Truth: 236

Response from RL **725 Tokens**

To find the sum of the squares of all items in the list that satisfies the given properties, let's analyze the requirements step-by-step:

1. The sum of the integers in the list must be equal to 30. (...)

Let's guess the list might be something like:

Suppose the list has an odd number of elements (...)

Let's try another configuration:

Let's try a list with sum 30: [1, 9, 9, 9, 7]: Sum is 30 correct.

Let's check the median is not in the list and unique mode is 9 is unique correctly.

The final valid list sum correctly would be: Let's do correct summing and checking properties correctly: (...)

One possible valid list correctly is: [1, 9, 9, 9, 7]: Sums correctly. Correct sum properties and correctly summed properties correctly would sum correctly.

Answer: 251

Response from RL w/ Entropy Adv. **3023 Tokens**

To solve this problem, we will systematically analyze the given conditions and construct the list of positive integers that (...).

Step-by-Step Analysis:

Step 1: Understanding the Mode (...)
Step 2: Understanding the Median (...)
Step 3: Setting Up the List (...)

Case Analysis:

Case 1: n is odd: If n is odd, (...)

Case 2: n is even If n is even, (...)

Constructing the List

Let's start by trying small values for n and see if we can satisfy all conditions.

Trying $n = 5$ (...)

Trying $n = 6$ (...)

Trying $n = 7$ (...)

Trying $n = 8$ (...)

After testing several more combinations, let's try $n = 6$ again with a different strategy: (...)

This works. The list is: (...)

Now, we need to find the sum of the squares of all the items (...)

Answer: $\boxed{236}$

Figure 6: Case study. Response comparison between RL-trained LMs with and without entropy-based advantage shaping.

8 Related Work

Exploration in Reinforcement Learning Exploration in RL (Ladosz et al. 2022; Li et al. 2025) is addressed through theoretical frameworks (Cai et al. 2020; Agarwal et al. 2020; Ishfaq et al. 2021), or empirical heuristics (Burda et al. 2019; Pathak et al. 2017; Raileanu and Rocktäschel 2020; Henaff et al. 2022). Motivated by the use of entropy to guide exploration (Haarnoja et al. 2018; Ziebart et al. 2008), we investigate its role in LM reasoning. A concurrent work (Gao et al. 2025) also studies exploration-driven reasoning but designs custom metrics rather than using entropy. Other studies apply entropy regularization (He et al. 2025; Wang et al. 2025b), while we focus on the advantage function.

Training Signals for LM Reasoning RL of LMs can leverage supervised or unsupervised training signals. Supervised methods rely on reward signals from verifiable correctness. Unsupervised methods leverage consistency-based signals (Prasad et al. 2024; Zuo et al. 2025) or entropy minimization (Zhang et al. 2025; Agarwal et al. 2025). We focus on unsupervised signals for exploratory reasoning.

9 Conclusion

This work investigates reasoning with exploration through the lens of entropy. We analyze the relation between entropy and exploration, revealing that exploratory reasoning actions consistently emerge in regions of higher entropy. Motivated by this, we introduce a minimal modification to RL algorithms by augmenting the advantage with a entropy-based term. This fosters exploration by promoting deeper reasoning, while preserving the original policy optimization direction. Experiments show that our method achieve strong improvements in Pass@K—highlighting a promising direction for exploration-aware LM training.

Acknowledgements

This paper was partially supported by the National Natural Science Foundation of China No. 92470205 and 62222215.

References

Agarwal, A.; Kakade, S.; Henaff, M.; and Sun, W. 2020. PC-PG: Policy Cover Directed Exploration for Provable Policy Gradient Learning. In *Advances in Neural Information Processing Systems*.

Agarwal, S.; Zhang, Z.; Yuan, L.; Han, J.; and Peng, H. 2025. The unreasonable effectiveness of entropy minimization in llm reasoning. *arXiv preprint arXiv:2505.15134*.

AI@Meta. 2024. Llama 3 Model Card.

Bai, F.; Min, Y.; Zhang, B.; Chen, Z.; Zhao, W. X.; Fang, L.; Liu, Z.; Wang, Z.; and Wen, J.-R. 2025. Towards Effective Code-Integrated Reasoning. *arXiv preprint arXiv:2505.24480*.

Burda, Y.; Edwards, H.; Storkey, A.; and Klimov, O. 2019. Exploration by Random Network Distillation. In *Proceedings of the 7th International Conference on Learning Representations*.

Cai, Q.; Yang, Z.; Jin, C.; and Wang, Z. 2020. Provably Efficient Exploration in Policy Optimization. In *Proceedings of the 37th International Conference on Machine Learning*.

Chen, M.; Tworek, J.; Jun, H.; Yuan, Q.; Pinto, H. P. D. O.; Kaplan, J.; Edwards, H.; Burda, Y.; Joseph, N.; Brockman, G.; et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Chen, Z.; Min, Y.; Zhang, B.; Chen, J.; Jiang, J.; Cheng, D.; Zhao, W. X.; Liu, Z.; Miao, X.; Lu, Y.; et al. 2025. An empirical study on eliciting and improving r1-like reasoning models. *arXiv preprint arXiv:2503.04548*.

Chu, X.; Huang, H.; Zhang, X.; Wei, F.; and Wang, Y. 2025. Gpg: A simple and strong reinforcement learning baseline for model reasoning. *arXiv preprint arXiv:2504.02546*.

Cui, G.; Yuan, L.; Wang, Z.; Wang, H.; Li, W.; He, B.; Fan, Y.; Yu, T.; Xu, Q.; Chen, W.; et al. 2025a. Process reinforcement through implicit rewards. *arXiv preprint arXiv:2502.01456*.

Cui, G.; Zhang, Y.; Chen, J.; Yuan, L.; Wang, Z.; Zuo, Y.; Li, H.; Fan, Y.; Chen, H.; Chen, W.; et al. 2025b. The Entropy Mechanism of Reinforcement Learning for Reasoning Language Models. *arXiv preprint arXiv:2505.22617*.

Gandhi, K.; Chakravarthy, A.; Singh, A.; Lile, N.; and Goodman, N. D. 2025. Cognitive behaviors that enable self-improving reasoners, or, four habits of highly effective stars. *arXiv preprint arXiv:2503.01307*.

Gao, J.; Pan, L.; Wang, Y.; Zhong, R.; Lu, C.; Cai, Q.; Jiang, P.; and Zhao, X. 2025. Navigate the Unknown: Enhancing LLM Reasoning with Intrinsic Motivation Guided Exploration. *arXiv preprint arXiv:2505.17621*.

Guo, D.; Yang, D.; Zhang, H.; Song, J.; Zhang, R.; Xu, R.; Zhu, Q.; Ma, S.; Wang, P.; Bi, X.; et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Haarnoja, T.; Zhou, A.; Abbeel, P.; and Levine, S. 2018. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. In *Proceedings of the 35th International Conference on Machine Learning*. PMLR.

He, J.; Liu, J.; Liu, C. Y.; Yan, R.; Wang, C.; Cheng, P.; Zhang, X.; Zhang, F.; Xu, J.; Shen, W.; et al. 2025. Skywork open reasoner 1 technical report. *arXiv preprint arXiv:2505.22312*.

Henaff, M.; Raileanu, R.; Jiang, M.; and Rocktäschel, T. 2022. Exploration via Elliptical Episodic Bonuses. In *Advances in Neural Information Processing Systems*.

Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; and Steinhardt, J. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Ishfaq, H.; Cui, Q.; Nguyen, V.; Ayoub, A.; Zhuoran, Y.; Wang, Z.; Precup, D.; and Yang, F. L. 2021. Randomized Exploration for Reinforcement Learning with General Value Function Approximation. In *Proceedings of the 38th International Conference on Machine Learning*.

Jiang, L.; Wu, X.; Huang, S.; Dong, Q.; Chi, Z.; Dong, L.; Zhang, X.; Lv, T.; Cui, L.; and Wei, F. 2025. Think only when you need with large hybrid-reasoning models. *arXiv preprint arXiv:2505.14631*.

Ladosz, P.; Weng, L.; Kim, M.; and Oh, H. 2022. Exploration in deep reinforcement learning: A survey. *Information Fusion*, 85: 1–22.

Lambert, N.; Morrison, J.; Pyatkin, V.; Huang, S.; Ivison, H.; Brahman, F.; Miranda, L. J. V.; Liu, A.; Dziri, N.; Lyu, S.; et al. 2024. T\" ulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*.

Li, Y.; Wong, P.; Zhang, H.; Chen, S.; and Qi, S. 2025. CAE: Repurposing the Critic as an Explorer in Deep Reinforcement Learning. *arXiv preprint arXiv:2503.18980*.

Liu, Z.; Chen, C.; Li, W.; Qi, P.; Pang, T.; Du, C.; Lee, W. S.; and Lin, M. 2025. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*.

MAA. 2023. American Mathematics Competitions - AMC. https://maa.org/.

MAA. 2025. American Invitational Mathematics Examination - AIME. https://maa.org/.

O'Donoghue, B.; Munos, R.; Kavukcuoglu, K.; and Mnih, V. 2016. PGQ: Combining policy gradient and Q-learning. *CoRR*.

Pathak, D.; Agrawal, P.; Efros, A. A.; and Darrell, T. 2017. Curiosity-driven Exploration by Self-supervised Prediction. In *Proceedings of the 34th International Conference on Machine Learning*. PMLR.

Prasad, A.; Yuan, W.; Pang, R. Y.; Xu, J.; Fazel-Zarandi, M.; Bansal, M.; Sukhbaatar, S.; Weston, J.; and Yu, J. 2024. Self-Consistency Preference Optimization. *arXiv preprint arXiv:2411.04109*.

Qian, C.; Liu, D.; Wen, H.; Bai, Z.; Liu, Y.; and Shao, J. 2025. Demystifying Reasoning Dynamics with Mutual Information: Thinking Tokens are Information Peaks in LLM Reasoning. *arXiv preprint arXiv:2506.02867*.

Raileanu, R.; and Rocktäschel, T. 2020. RIDE: Rewarding Impact-Driven Exploration for Procedurally-Generated Environments. In *Proceedings of the 8th International Conference on Learning Representations*.

Reimers, N.; and Gurevych, I. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Schulman, J.; Moritz, P.; Levine, S.; Jordan, M.; and Abbeel, P. 2015. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*.

Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal Policy Optimization Algorithms. *arXiv preprint arXiv: 1707.06347v2*.

Shah, D. J.; Rushton, P.; Singla, S.; Parmar, M.; Smith, K.; Vanjani, Y.; Vaswani, A.; Chaluvaraju, A.; Hojel, A.; Ma, A.; et al. 2025. Rethinking reflection in pre-training. *arXiv preprint arXiv:2504.04022*.

Shao, Z.; Wang, P.; Zhu, Q.; Xu, R.; Song, J.; Bi, X.; Zhang, H.; Zhang, M.; Li, Y.; Wu, Y.; et al. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Sheng, G.; Zhang, C.; Ye, Z.; Wu, X.; Zhang, W.; Zhang, R.; Peng, Y.; Lin, H.; and Wu, C. 2024. HybridFlow: A Flexible and Efficient RLHF Framework. *arXiv preprint arXiv: 2409.19256*.

Wang, S.; Yu, L.; Gao, C.; Zheng, C.; Liu, S.; Lu, R.; Dang, K.; Chen, X.; Yang, J.; Zhang, Z.; et al. 2025a. Beyond the 80/20 Rule: High-Entropy Minority Tokens Drive Effective Reinforcement Learning for LLM Reasoning. *arXiv preprint arXiv:2506.01939*.

Wang, Y.; Yang, Q.; Zeng, Z.; Ren, L.; Liu, L.; Peng, B.; Cheng, H.; He, X.; Wang, K.; Gao, J.; et al. 2025b. Reinforcement learning for reasoning in large language models with one training example. *arXiv preprint arXiv:2504.20571*.

Wang, Z.; Zhou, F.; Li, X.; and Liu, P. 2025c. Octothinker: Mid-training incentivizes reinforcement learning scaling. *arXiv preprint arXiv:2506.20512*.

Yang, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Li, C.; Liu, D.; Huang, F.; Wei, H.; Lin, H.; Yang, J.; Tu, J.; Zhang, J.; Yang, J.; Yang, J.; Zhou, J.; Lin, J.; Dang, K.; Lu, K.; Bao, K.; Yang, K.; Yu, L.; Li, M.; Xue, M.; Zhang, P.; Zhu, Q.; Men, R.; Lin, R.; Li, T.; Xia, T.; Ren, X.; Ren, X.; Fan, Y.; Su, Y.; Zhang, Y.; Wan, Y.; Liu, Y.; Cui, Z.; Zhang, Z.; and Qiu, Z. 2024a. Qwen2.5 Technical Report. *CoRR*, abs/2412.15115.

Yang, A.; Zhang, B.; Hui, B.; Gao, B.; Yu, B.; Li, C.; Liu, D.; Tu, J.; Zhou, J.; Lin, J.; Lu, K.; Xue, M.; Lin, R.; Liu, T.; Ren, X.; and Zhang, Z. 2024b. Qwen2.5-Math Technical Report: Toward Mathematical Expert Model via Self-Improvement. *CoRR*, abs/2409.12122.

Yu, Q.; Zhang, Z.; Zhu, R.; Yuan, Y.; Zuo, X.; Yue, Y.; Fan, T.; Liu, G.; Liu, L.; Liu, X.; et al. 2025. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*.

Yue, Y.; Chen, Z.; Lu, R.; Zhao, A.; Wang, Z.; Song, S.; and Huang, G. 2025a. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? *arXiv preprint arXiv:2504.13837*.

Yue, Y.; Yuan, Y.; Yu, Q.; Zuo, X.; Zhu, R.; Xu, W.; Chen, J.; Wang, C.; Fan, T.; Du, Z.; et al. 2025b. Vapo: Efficient and reliable reinforcement learning for advanced reasoning tasks. *arXiv preprint arXiv:2504.05118*.

Zhang, Q.; Wu, H.; Zhang, C.; Zhao, P.; and Bian, Y. 2025. Right question is already half the answer: Fully unsupervised llm reasoning incentivization. *arXiv preprint arXiv:2504.05812*.

Ziebart, B. D.; Maas, A. L.; Bagnell, J. A.; Dey, A. K.; et al. 2008. Maximum entropy inverse reinforcement learning. In *Aaai*, volume 8, 1433–1438. Chicago, IL, USA.

Zuo, Y.; Zhang, K.; Sheng, L.; Qu, S.; Cui, G.; Zhu, X.; Li, H.; Zhang, Y.; Long, X.; Hua, E.; et al. 2025. Ttrl: Test-time reinforcement learning. *arXiv preprint arXiv:2504.16084*.