Reward Design for Reinforcement-Learning Fine-Tuning of Navigation Policies

Yunshan Wang
Department of Physics
Stanford University United States
jerryw@stanford.edu

Extended Abstract

I was working on Reasoning VLAs, particularly for autonomous driving, for a while. I started noticing many instances of the model producing perfectly good reasoning that did not match its output. That motivate me to ask, *how could be reward model reasoning and output in a way that it is encouraged to generate reasoning that corresponds to its output*?

I reviewed the RL rewards used by reasoning VLAs, particularly in autonomous driving, and noticed that (i) they usually reward on outcome, (ii) they often reward on reasoning quality, and (iii) they use (if at all) hand-crafted rules to enforce consistency between reasoning and action. These rewards are problematic because they can reward good-sounding reasoning that doesn't improve outcomes, and they need hand-crafted rules. Thus, I ask, *can we design a simple reward that couples reasoning to its effect on results*?

I found RLPR, which rewards a Chain-of-Thought only when it raises the model's own probability of the correct answer. This is a natural way to reward reasoning only when it is useful, thus encouraging reasoning that leads to better outcomes rather than simply superficial nice-looking reasoning. However, the RLPR paper tested on math, which has a discrete state and action space. But VLAs output in continuous space, which makes it much harder to compute the probability of the "reference trajectory."

I came up with two ways to measure the trajectory probability. First, I realized I can train a Gaussian head and use the probability density from that head as a reference probability density reward. That corresponds to the experiment I labelled R1. Secondly, I realized that I can set a tolerance and compute probability mass that the model's output waypoints would fall under that tolerance range of ground truth. In addition to these, I tested the most trivial measure of useful reasoning — difference in ADE between reasoning and no-reasoning — and just directly rewarding outcome.

I found that direct reward is on par with reasoning delta. None of the RLPR base rewards worked. This was quite surprising and disappointing. Though I know there are lots of design choices I made that may have been shaky, so I did some diagnostics, and found some surprising things that explain my results. Firstly, I found that the reasoning has rather little influence on the ADE. Most prominent is that reasoning is actually a small fraction of total context due to how SFT reasoning was constructed. Also, I realized there are a few design mistakes I made in my RLPR rewards that lead to poor performance. Firstly, the Gaussian head didn't work because the CoT embedding states were out-of-distribution to it. Secondly, the tolerance ball approach didn't work because CoT drifted probability mass away from tolerance.

In short, I realized my results weren't a fair test of RLPR — there are some fixes that have to be made to really test my hypothesis well.

Abstract: LLM reasoning has become a key tool to improve performance of autonomous vehicle navigation policies. But the generated reasoning often does not causally drive predictions. Existing RL remedies grade outcomes only or rely on hand-crafted consistency rules, neither of which directly tests whether the reasoning improved the prediction. We propose adapting RLPR - a verifier-free method that rewards reasoning only when it raises the model's own probability of the reference answer - and a simpler reasoning-delta reward, to trajectory prediction. We GRPO-fine-tune an SFT Qwen3-VL-4B navigation VLA checkpoint. Across our reward designs, held-out ADE differences are within measurement noise and the policy barely moves at our compute budget. A post-mortem set of diagnostics explains why. (i), the chain-of-thought contributes little to the predicted trajectory partly by construction, as the reasoning is short, meta-action-focused, and 5% of a boilerplate-dominated prompt. CoT vs no-CoT ADE essentially equal once output length is controlled, so a reasoning-fidelity reward has little signal to amplify. (ii), both reference rewards failed for identifiable, fixable implementation reasons - a density head frozen on no-CoT hidden states then applied to CoT states, which are out of distribution for it; and a tolerance ball set far too tight. Thus these rewards were never fairly tested. We thus neither confirm nor refute the RLPR hypothesis for trajectory VLAs. Our contribution is a concrete diagnostic account of why reasoning-reward RL is hard in this setting and what must change to test it fairly.

1 Introduction

Autonomous-driving policies tend to fail in long-tail scenarios that are under-represented in their training data. A growing body of work injects large language model reasoning — chain-of-thought (CoT) deliberation [1] — into navigation policies to add "common-sense" robustness in exactly these cases (Section 2). Reasoning about the scene presumably should help most where pattern-matching from data runs out. However, the generated reasoning may not *causally* drive the prediction, and the existing RL remedies either grade outcomes only or rely on hand-crafted reasoning–action consistency rules — neither of which directly tests whether the reasoning *improved the prediction* (Section 2).

We study a reward that targets this gap. We adapt RLPR — a verifier-free method that rewards a CoT only when it raises the model's own probability of the reference answer — together with a simpler reasoning-delta reward (the ADE improvement of a CoT rollout over a no-CoT baseline), to trajectory prediction, and GRPO-fine-tune an SFT **Qwen3-VL-4B** [2] navigation VLA trained on NVIDIA Physical AI autonomous-vehicle reasoning data [3]. Because the policy emits waypoints as text one digit per token, the naive reference-probability reward is ill-defined, so we compare four reward designs spanning three families: direct trajectory accuracy, the reasoning-delta, and two RLPR-style reference rewards (a frozen Gaussian density head and a waypoint tolerance-ball probability).

Our headline result is negative, and our contribution is the explanation. Across all four designs the held-out ADE differences are within measurement noise and the policy barely moves at our compute budget, so the leaderboard alone cannot adjudicate the hypothesis. A post-mortem battery of diagnostics explains why, and yields the substance of this paper. Concretely, we contribute:

- an adaptation of RLPR and a reasoning-delta reward to a continuous, multimodal trajectory VLA, including the per-digit-token obstacle that rules out the naive reward and the two coarsened references it motivates (Section 3);
- the empirical finding that, at our budget, rewarding reasoning fidelity gives no gain over rewarding accuracy directly, on a near-static policy (Section 4);

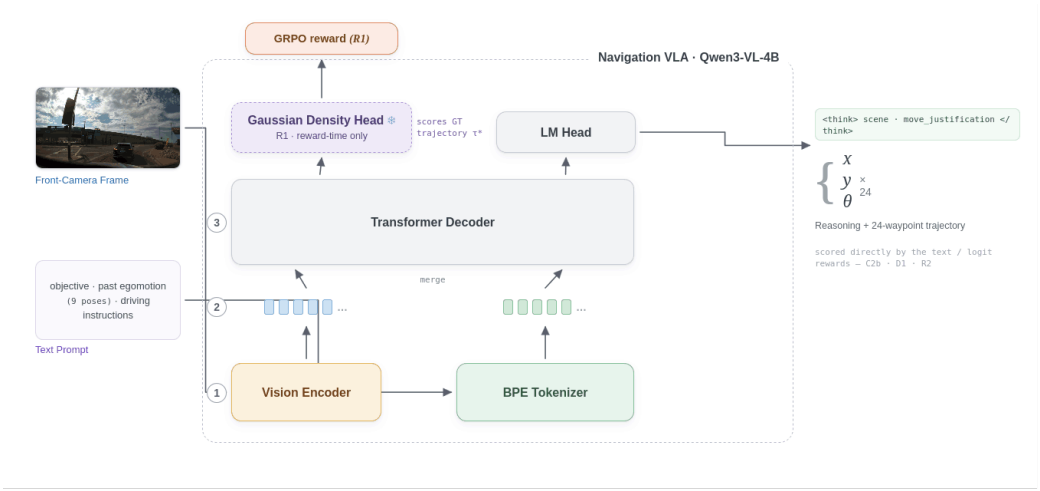

Figure 1: **Reasoning-VLA architecture.** A Qwen3-VL-4B backbone encodes the front-camera frame and text prompt, then autoregressively decodes a reasoning trace and 24-waypoint trajectory through the LM head. The R1 variant adds a frozen Gaussian density head (used only at reward time) that scores the ground-truth trajectory from the decoder's hidden state to produce the GRPO reward.

- a diagnostic account of *why* — the CoT is ≈neutral for the trajectory partly by construction, and the two reference rewards failed for identifiable, fixable implementation reasons and were never fairly tested (Section 5); and
- the concrete changes a fair re-test would require, leaving the original hypothesis neither confirmed nor refuted (Section 6).

2 Related Work

Reasoning-augmented driving and embodied policies. A number of vision-language-action (VLA) models couple chain-of-thought (CoT) reasoning with low-level control to improve robustness in the long tail of driving [4, 5] and robotic manipulation [6, 7]. Most train reasoning component either by supervised fine-tuning on free-form CoT or with simple outcome-based RL rewards such as task success or trajectory quality [6, 7, 5]. Some works like Zhou et al. [5], use GRPO to learn *when* to reason (fast vs. slow thinking) rather than *whether* the reasoning improved the prediction. Wang et al. [4] target the reasoning-action link directly with a composite RL reward that combines an LM judge on the reasoning, hand-crafted reasoning-action consistency rules, and a trajectory-quality term. The rules bake in human assumptions and the LM judge can reward reasoning that merely *looks* correct. Across these works the central gap is that nothing during training certifies that the reasoning *causally* improves the action.

The faithfulness gap. That is an instance of a broader finding that a model's stated CoT need not reflect or drive its answer: explanations can be plausible yet unfaithful [8], and the answer often barely changes when the CoT is perturbed or truncated [9]. A reward that grades reasoning by its measured effect on the prediction is therefore an attractive remedy — *if* such an effect exists to grade.

RL with verifiable and reference rewards. Our method sits in the reinforcement-learning-with-verifiable-rewards (RLVR) paradigm popularized by DeepSeek-AI [10], optimized here with GRPO

[11] and its DAPO variant [12]. RLVR's reliance on a domain verifier confines it largely to math and code. Yu et al. [13] remove the verifier with RLPR, rewarding a CoT by the model's own probability of the reference answer, so reasoning that does not raise that probability earns nothing. This is the verifier-free, reference-probability signal we adapt. Rewarding the reasoning rather than only the outcome mirrors the process- vs. outcome-supervision distinction studied for math reasoning [14, 15]: our direct-ADE controls are outcome rewards, while the reasoning-delta and RLPR-style rewards are process-flavored.

Trajectory prediction as token-level language modeling. Unlike math QA, a trajectory is a continuous, multimodal target. Following motion-forecasting-as-language- modeling [16] and action-as-text VLAs [17], our policy emits waypoints as text, one digit per token. This makes the naive RLPR reward — the sequence log-probability of the reference trajectory — a step function of geometric distance rather than a smooth density, which is exactly why we introduce the two coarsened references studied here (a frozen Gaussian density head and a tolerance-ball probability). Our contribution is not a new model but a diagnostic account of why these rewards, and the simpler reasoning-delta reward, do not yet beat a direct accuracy reward in this setting.

3 Methodology and Setup

We study a single, fixed navigation policy and vary only *how it is rewarded* during reinforcement-learning fine-tuning. All design choices except the reward function are held constant across runs.

3.1 Policy, output format, and metric

The policy is a one-epoch supervised-fine-tuned (SFT) checkpoint of **Qwen3-VL-4B**. From a single front-camera frame and a system/user prompt (driving objective, nine past ego-poses, a fixed meta-action menu), it emits a structured chain-of-thought (CoT): a `<think>` block (scene description + move justification), an `<action>` block (longitudinal/lateral meta-action labels), and 24 future waypoints `<wp>` $[x, y, \theta]$ `</wp>` at 4 Hz (a 6 s horizon) in the ego frame. Two properties matter for reward design: the model is **strictly text-autoregressive** with no continuous output head, and the tokenizer emits numbers **one digit per token** (`3.69` $\rightarrow [3][.][6][9]$). The sequence log-probability of a trajectory is therefore a step function of geometric distance, not a smooth density, so a naive $\log p(\tau_{gt})$ reward is unusable — which is why the RLPR references (Section 3.4) need either a bolted-on density head (R1) or a coarsened tolerance probability (R2). The primary metric is **average displacement error (ADE)**: the mean L_2 distance over (x, y) between predicted and ground-truth waypoints (metres, lower is better), with last-point padding when waypoint counts differ.

3.2 Data and evaluation regimes

We train on US data with about 14,000 filtered records. We evaluate in two regimes: **US in-distribution (US-ID)**, carved from the same US data source, and **Germany out-of-distribution (DE-OOD)**. Unless noted, all evaluation numbers are greedy decode over $N = 2000$ held-out samples per split.

3.3 RL algorithm and budget

All runs use Group Relative Policy Optimization (GRPO) via the TRL `GRPOTrainer` under `accelerate` on 2×H100 (bf16). The reward is the only thing that varies between configurations; every other hyperparameter is shared and pinned (Table 1). Notably, each run only a short nudge on top of SFT, not a converged policy. We adopt `scale_rewards=group` (GRPO group-std advantage normalization) for all configurations.

Table 1: Shared GRPO training configuration (identical across all runs; only the reward function differs).

Setting	Value
Algorithm	GRPO (TRL `GRPOTrainer`), `loss_type=dapo`, `num_iterations=1`
Init checkpoint	`PhysicalAI-reason-VLA-MetaAction-1e` (Qwen3-VL-4B SFT, 1 epoch)
Hardware / precision	2×H100 (Modal), bf16
Train data	`PhysicalAI-Reason-US`, first 14,000 filtered records
Generations per prompt	16
Batch × accum × procs	$8 \times 3 \times 2 = 48$ prompt-visits/step
Steps	500 (≈ 0.11 epoch)
Max completion length	768 tokens
Advantage normalization	`scale_rewards=group`: $(r - \mu_{\text{group}})/(\sigma_{\text{group}} + \epsilon)$
Grad checkpointing	on, `use_reentrant=False` (DDP-safe)
Learning rate	linear decay from ≈1e-6
Cached-baseline TTL	32 grad-steps (delta/reference rewards)

3.4 Reward designs (Methodology)

We compare configurations spanning three families (Table 2). All "delta" rewards (C2b, R1, R2) use the sign convention *positive = reasoning helped*: the CoT context scores the ground-truth trajectory better than a no-CoT baseline.

Direct accuracy (D-series). The controls score the rollout's own trajectory against ground truth with no reasoning term. **D1′** (`NegADEReward`, the headline control) and **D1** (`ADEnReward`, retained only to quantify the cost of exponential squashing) are

$$r_{\text{D1}'} = -\,\text{ADE}(\hat{\tau}, \tau_{gt}), \qquad r_{\text{D1}} = \exp\big(-\,\text{ADE}(\hat{\tau}, \tau_{gt})\big) \in (0, 1]. \tag{1}$$

Reasoning delta (C-series). **C2b** (`FaithfulnessReward`, raw variant) is the most direct test of the hypothesis — it rewards how much the CoT lowers ADE relative to a no-CoT baseline:

$$r_{\text{C2b}} = \text{ADE}_{\text{base}} - \text{ADE}_{\text{cot}}, \tag{2}$$

where ADE_{cot} is the rollout's ADE and ADE_{base} is a greedy no-CoT (`regular`-prompt) generation cached per image (TTL=32 grad-steps). Positive means the reasoning helped.

RLPR-style reference rewards (R-series). These adapt RLPR — rewarding the CoT only when it raises the model's own probability of the reference (ground-truth) trajectory — to a text-autoregressive VLA, scoring the *change* the CoT induces:

$$r_{\text{R1}} = \log p(\tau_{gt} \mid o, \text{CoT}) - \log p(\tau_{gt} \mid o, \text{base}), \qquad r_{\text{R2}} = P_{\text{ball}}(\tau_{gt} \mid \text{CoT}) - P_{\text{ball}}(\tau_{gt} \mid \text{base}). \tag{3}$$

For **R1** (`GaussianDensityReward`), p is a small *frozen* diagonal-Gaussian MLP head over the VLA's last-token hidden state (trunk $2560 \to 1024 \to 1024$ GELU, μ-head $\to 72$, per-step/per-dim oracle σ), evaluated over all 72 dims (x, y, θ) via two `output_hidden_states` forward passes (no `.generate()`); the head is used only at reward time and its gradients never reach the policy. For **R2** (`WaypointToleranceBallReward`), P_{ball} is the mean over 24 waypoints of $P(|x - x^*| \le \epsilon)\, P(|y - y^*| \le \epsilon)$ computed from teacher-forced digit softmaxes via a per-axis digit dynamic program, with shipped tolerance $\epsilon = 0.1$ m. R1 is the only configuration that adds any module to the policy, and even then only as a reward evaluator; otherwise the VLA is untouched.

4 Primary Results (Quantitative Evaluation)

Held-out ADE for all six configurations on both splits (Table 3, Figure 2) show that direct ADE reward is the best, slightly beating reasoning delta reward. However, the gaps are small and the policy barely moved, as expected given our training budget.

Table 2: The reward configurations. ADE is over (x, y) in metres; lower ADE is better, so the direct rewards negate it. All use `scale_rewards=group`.

Tag	Family	Reward per rollout	Forward pass(es)
SFT	—	(pre-RL reference point)	—
D1′	direct	$-\text{ADE}(\text{pred}, \text{GT})$ (raw)	none (post-hoc on text)
D1	direct	$\exp(-\text{ADE}(\text{pred}, \text{GT})) \in (0, 1]$	none
C2b	reasoning delta	$\text{ADE}_{\text{base}} - \text{ADE}_{\text{cot}}$ (raw)	`.generate()` no-CoT baseline
R1	RLPR reference	$\Delta \log p(\tau_{gt})$ via frozen Gaussian head	$2\times$ hidden-state fwd
R2	RLPR reference	$\Delta P_{\text{ball}}(\tau_{gt})$, digit-DP, $\epsilon = 0.1$ m	1 teacher-forced fwd

Table 3: Held-out ADE (metres, lower is better) over $N = 2000$ greedy-decoded samples per split, at checkpoint-500. Ordered by US-ID mean.

		US in-distribution			DE out-of-distribution		
Config	Reward family	mean	median	p90	mean	median	p90
D1′ (raw $-$ADE)	direct	**6.080**	**5.041**	**11.927**	9.294	8.754	14.601
C2b (reasoning delta)	reasoning-aware	6.102	5.046	12.021	**9.226**	8.796	**14.346**
D1 (exp($-$ADE))	direct	6.202	5.133	12.183	9.366	8.891	14.558
SFT base (pre-RL)	—	6.332	5.292	12.387	9.431	8.936	14.766
R2 (tolerance ball)	RLPR reference	6.372	5.422	12.282	9.342	**8.956**	14.779
R1 (density head)	RLPR reference	6.403	5.445	12.642	9.437	9.027	14.696

Rewarding the reasoning buys nothing over rewarding accuracy. The reasoning -delta reward C2b (US-ID 6.10) and the plain direct control D1′ (US-ID 6.08) are a **tie** ($\Delta 0.02$ m, within noise) and simply trade the top spot across splits: on DE-OOD, C2b (9.23) edges D1′ (9.29). The experiment's central question,*does scoring the CoT's contribution beat scoring accuracy directly?*, gets a **no** answer from our current experiments.

Both RLPR-style reference rewards fail to beat SFT. R2 (6.37) and R1 (6.40) both land **at or below the pre-RL SFT baseline (6.33)** on US-ID, and R1 is last on both splits, fractionally *worse* than the un-fine-tuned model (DE-OOD 9.44 vs SFT 9.43). At face value, the verifier-free reference rewards we set out to validate do not help. Section 5 shows this conclusion is unearned: both rewards carry fixable implementation bugs and were never given a fair test.

The simplest reward wins, and $\exp(\cdot)$ hurts. Raw $-$ADE (D1′, 6.08) is the single best configuration and beats its exp-squashed sibling $\exp(-\text{ADE})$ (D1, 6.20) by ≈ 0.12 m on US-ID. The exponential compresses the metre-scale learning signal in the tail; among what was tested, a clean unbounded $-$ADE with group-std normalization is the right call.

5 Qualitative Analysis

To understand the results from Section 4, we ran a set of cheap diagnostics to ask *why* the reasoning-aware and reference rewards bought nothing. Four findings emerge: the policy barely moved (Section 5.1); the chain-of-thought is ≈neutral for the trajectory, partly by construction (Section 5.2); R1 failed from a fixable out-of-distribution head bug rather than "density $\neq$ accuracy" (Section 5.3); and R2's CoT-induced distribution shift lands outside its tolerance bound (Section 5.4).

5.1 Policy barely moved

Across all five runs the policy hardly leaves its SFT initialization: KL-to-reference stays ∼1e-3 and entropy is flat at ≈0.51 for the full 500 steps (Figure 3). The ≈ 0.3 m eval spread in Table 3 is therefore a comparison of reward *designs* on a near-static policy — suggestive, not converged. Any claim about converged behaviour would need a much longer run.

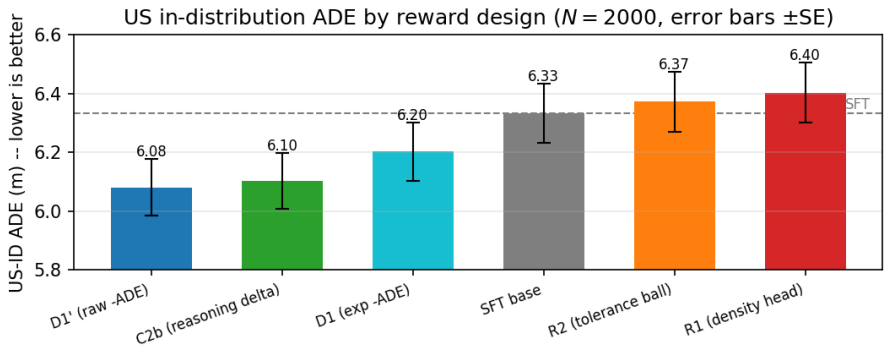

Figure 2: Mean US-ID ADE by reward design ($N = 2000$), with ±SE error bars ($\sigma/\sqrt{N} \approx 0.10$ m) and the SFT baseline (dashed).

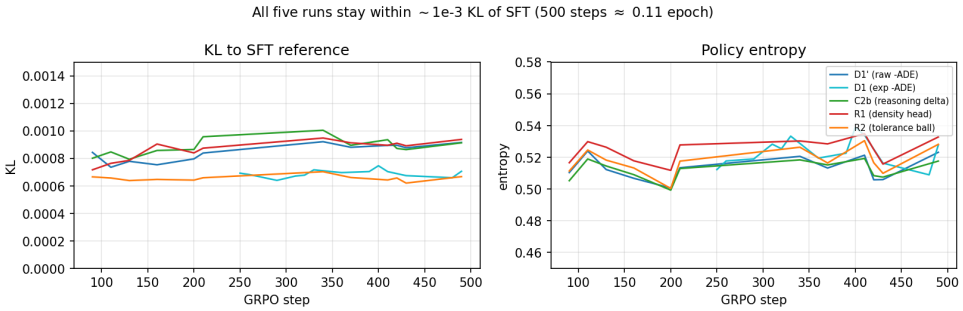

Figure 3: Training dynamics across all five runs (full 500 steps): KL to the SFT reference (left) never exceeds ~1e-3 and entropy (right) holds ≈0.51, so the policy barely moved at this 0.11-epoch budget.

5.2 The chain-of-thought is ≈neutral for the trajectory

Length-controlled, CoT ≈ no-CoT. Comparing greedy SFT with and without the CoT on the same US-ID split, the *naive* numbers (CoT 6.33 vs no-CoT 4.31) suggest reasoning *hurts* by ≈2 m. However, `compute_ade` scores over $\min(\text{len}_{\text{pred}}, \text{len}_{\text{gt}})$ waypoints, and the no-CoT prompt (off-distribution for this SFT model) truncates — a median of 20/24 waypoints, only 17% emitting the full 24 — so its ADE is scored over the easy near-horizon. **Length-controlled**, on the 336 scenes where both emit a full 24 waypoints, the gap vanishes: **CoT 6.840 vs no-CoT 6.841** ($\Delta \approx 0$). The reasoning is essentially neutral for ADE.

Why, structurally. The CoT has little mechanical leverage to begin with. Of a ≈2293-token context, the reasoning (`<think>`, ≈112 tokens) is only ≈5%, and is shorter than the past-motion block alone (Figure 4); 83% of the 1388 prompt tokens is fixed instruction boilerplate (system meta-action menus + user framing). Crucially, the reasoning was *trained to be short and to pick discrete maneuvers*, not to deliberate about geometry or refine waypoints. A reasoning channel built that way has little path to the trajectory.

Consequence. Every reasoning-aware reward — C2b's ADE delta, R1's $\Delta \log p$, R2's ΔP_{ball} — is trying to amplify a small CoT-vs-base signal. This, more than any single reward bug, is why the sophistication bought nothing over a plain $-$ADE reward. If reasoning is to help the trajectory, the lever is the *reasoning construction*, not the RL reward.

Figure 4: Token budget of the ≈2293-token context: the reasoning is ≈5% and 83% of the prompt is fixed boilerplate.

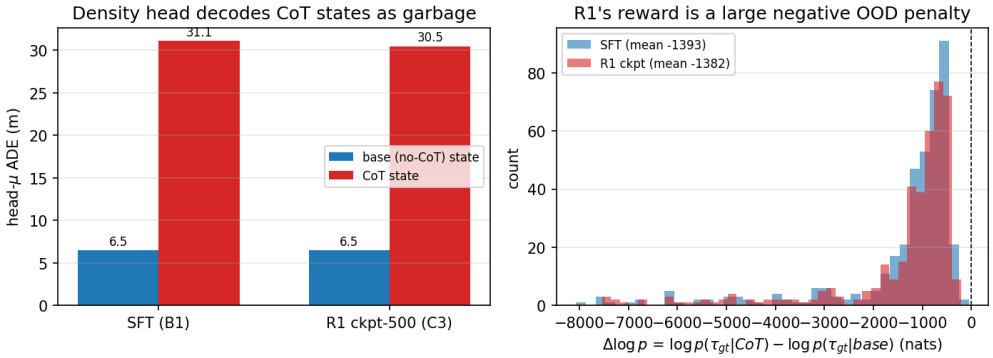

Figure 5: R1's out-of-distribution head bug. **Left:** the frozen density head decodes base (no-CoT) states to a sane ≈6.5 m μ-ADE but CoT states to ≈31 m garbage. **Right:** R1's actual reward, $\Delta \log p$, is a large negative penalty (≈ -1390 nats) — the signature of scoring an OOD state, not a density gradient.

5.3 R1's real failure is a fixable OOD-head bug

The head is a good ADE proxy — on the states it was trained on. On base/no-CoT hidden states (the condition the head was frozen on), the frozen Gaussian head is healthy: head-μ ADE P50 ≈3.6 m, no μ-collapse, and $\log p \leftrightarrow -$ADE Spearman $\rho = 0.95$ over (x, y). The density genuinely tracks accuracy.

But at reward time it is fed CoT states, which are OOD. R1's reward conditions the head on the *CoT-context* hidden state, which is out of distribution for a head frozen on base states. Measured directly, the head's μ-ADE jumps from **6.5 m on the base state to 31 m on the CoT state**, and the reward $\Delta \log p$ collapses to ≈ -1390 nats (Figure 5). R1's reward was measuring *how unfamiliar the CoT state is to a base-trained head* — not trajectory density. Consistently, the R1 checkpoint is statistically identical to SFT on every head-space metric, i.e. the reward delivered no usable gradient.

5.4 R2's CoT shift lands outside the tolerance bound

R2 was also never fairly tested. A forward-only probe shows the CoT *does* shift the waypoint-digit distribution — the per-axis pmf KL(cot‖base) is ≈0.077 > 0 — but at the shipped $\epsilon = 0.1$ m the shift drifts the pmf mass away from the tight ball, so the reward differential ΔP_{ball} has little usable signal to reward. We do not have enough evidence to claim a specific better tolerance; we note only that the diagnostic locates the failure in the interaction between the CoT's distribution shift and the $\epsilon = 0.1$ m bound, not in the absence of any CoT effect.

5.5 The rewards that moved the needle share one mechanism

Decoding SFT, C2b, and D1′ on shared scenes, both rewards reach the same small ADE gain the same way: they barely change the reasoning (cosine change ≈0.02 for both) and shift the waypoints by a similar amount. C2b changes the reasoning fractionally more and the waypoints fractionally less — the direction the reasoning-reward hypothesis predicts — but the difference is within noise. We flag this only as suggestive: it offers no distinct "reasoning mechanism" to separate C2b from D1′, consistent with their leaderboard tie.

6 Discussion

Across six configurations the *reward*, not the policy or the data, was the only thing we varied — yet none of the reasoning-aware or reference rewards beat a plain accuracy reward (Section 4). The diagnostics explain why with two upstream constraints rather than any single reward flaw. First, at our budget the policy barely moved (Section 5.1), so the leaderboard mostly measures headroom, not the merits of the ideas. Second, the chain-of-thought carries almost no trajectory signal here — partly by construction, since it is short, meta-action-focused, and a small fraction of a boilerplate-dominated prompt (Section 5.2) — so every reasoning-aware reward (C2b, R1, R2) was amplifying a near-zero signal, and the two reference rewards additionally never ran as intended (Sections 5.3 and 5.4). The net position is that the original hypothesis is *neither confirmed nor refuted*: a reasoning-fidelity reward was never given a setting in which it could plausibly help.

Toward a fair test. The diagnostics point to concrete prerequisites rather than a verdict. The highest-leverage change is upstream of the reward entirely: make the reasoning able to matter, i.e. an SFT/data change so the CoT deliberates about geometry and waypoints instead of selecting discrete maneuvers. Given such reasoning, the reference rewards must be re-wired to score it faithfully — R1 with a CoT-condition or co-updated (target-network) density head so it evaluates in-distribution states rather than penalizing their novelty (Section 5.3), and R2 with a tolerance matched to the CoT's measured distribution shift (Section 5.4) — and runs must be long enough ($\gg 0.11$ epoch) for the policy to actually move before any ranking is read. The one result that survives all of this is mundane but robust: among the rewards we did test, a clean raw $-$ADE with group normalization is best, and exponentiating it hurts.

7 Limitations

Our conclusions are bounded in several ways, most of which sharpen the "neither confirmed nor refuted" stance. **Budget and convergence:** every run is 500 GRPO steps ≈ 0.11 epoch with KL-to-reference $\sim$1e-3, so the policy is near-static and the rankings are suggestive, not converged; we also have no calibration for what a given KL magnitude means for this VLA. **Scope:** we study a single VLA, dataset, and one seed per configuration, with no across-seed variance estimate, which makes it difficult to distinguish signal from noise. **Metric footgun:** `compute_ade` scores over the predicted/ground-truth waypoint overlap, so truncated outputs look artificially accurate. This produced the "reasoning hurts" artifact (Section 5.2). And, it likely contaminates the training-time no-CoT baseline used by C2b, which, unlike our diagnostics, did not length-control. **Probe confounds:** our CoT-contribution probes compare a sampled CoT rollout against a greedy baseline and swap the prompt between the CoT and no-CoT conditions (the no-CoT prompt is itself off-distribution). A clean ablation would hold the prompt fixed and toggle only the `<think>` step.

8 Conclusion

We asked whether rewarding reasoning fidelity — via an RLPR-style reference reward or a simpler reasoning-delta reward — beats rewarding trajectory accuracy directly when GRPO-fine-tuning a navigation VLA. At our budget the held-out ADE differences across reward designs are small, and we *neither confirm nor refute* the hypothesis. The contribution is the diagnosis of why: the chain-of-thought is about neutral for the trajectory partly by construction, so reasoning-aware rewards have little to amplify; both reference rewards failed for identifiable, fixable implementation reasons and were therefore never fairly tested; and all of this sits on a policy that barely moved. We pair this account with the concrete changes a fair re-test would require — reasoning that deliberates about geometry, a CoT-condition or co-updated density head, a tolerance matched to the CoT's measured shift, and longer training. We hope the failure modes catalogued here save others the same debugging in adapting verifier-free reasoning rewards to continuous, multimodal action spaces.

9 Team Contributions

As mentioned in my project proposal, this project was originally conceived as part of a larger group effort on faithfulness in reasoning VLAs. All the datasets I used, the SFT checkpoint I built on, and also the RL training pipeline (besides my custom rewards) were built by my collaborators.

My contributions were the custom rewards I designed and implemented for RL fine-tuning, the training and evaluation of the RL fine-tuning runs, and the subsequent diagnostic analysis.

References

[1] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, and D. Zhou. Chain-of-thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*, 2022.

[2] Qwen Team. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025.

[3] NVIDIA. Cosmos-Reason1: From physical common sense to embodied reasoning. *arXiv preprint arXiv:2503.15558*, 2025.

[4] Y. Wang, W. Luo, J. Bai, Y. Cao, T. Che, K. Chen, Y. Chen, B. Ivanovic, M. Pavone, et al. Alpamayo-R1: Bridging reasoning and action prediction for generalizable autonomous driving in the long tail. *arXiv preprint arXiv:2511.00088*, 2025.

[5] Z. Zhou, T. Cai, S. Z. Zhao, Y. Zhang, Z. Huang, B. Zhou, and J. Ma. AutoVLA: A vision-language-action model for end-to-end autonomous driving with adaptive reasoning and reinforcement fine-tuning. *arXiv preprint arXiv:2506.13757*, 2025.

[6] C. Yin, Y. Lin, W. Xu, S. Tam, X. Zeng, Z. Liu, and Z. Yin. DeepThinkVLA: Enhancing reasoning capability of vision-language-action models. *arXiv preprint arXiv:2511.15669*, 2025.

[7] D. Kim, S. Park, H. Jang, J. Shin, J. Kim, and Y. Seo. Robot-R1: Reinforcement learning for enhanced embodied reasoning in robotics. *arXiv preprint arXiv:2506.00070*, 2025.

[8] M. Turpin, J. Michael, E. Perez, and S. R. Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *arXiv preprint arXiv:2305.04388*, 2023.

[9] T. Lanham, A. Chen, A. Radhakrishnan, B. Steiner, C. Denison, D. Hernandez, D. Li, E. Durmus, E. Hubinger, J. Kernion, et al. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*, 2023.

[10] DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[11] Z. Shao, P. Wang, Q. Zhu, R. Xu, J. Song, X. Bi, H. Zhang, M. Zhang, Y. K. Li, Y. Wu, and D. Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[12] Q. Yu, Z. Zhang, R. Zhu, Y. Yuan, X. Zuo, et al. DAPO: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

[13] T. Yu, B. Ji, S. Wang, S. Yao, Z. Wang, G. Cui, L. Yuan, N. Ding, Y. Yao, Z. Liu, M. Sun, and T.-S. Chua. RLPR: Extrapolating RLVR to general domains without verifiers. *arXiv preprint arXiv:2506.18254*, 2025.

[14] J. Uesato, N. Kushman, R. Kumar, F. Song, N. Siegel, L. Wang, A. Creswell, G. Irving, and I. Higgins. Solving math word problems with process- and outcome-based feedback. *arXiv preprint arXiv:2211.14275*, 2022.

[15] H. Lightman, V. Kosaraju, Y. Burda, H. Edwards, B. Baker, T. Lee, J. Leike, J. Schulman, I. Sutskever, and K. Cobbe. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023.

[16] A. Seff, B. Cera, D. Chen, M. Ng, A. Zhou, N. Nayakanti, K. S. Refaat, R. Al-Rfou, and B. Sapp. MotionLM: Multi-agent motion forecasting as language modeling. *arXiv preprint arXiv:2309.16534*, 2023.

[17] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, et al. RT-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.