Robotics: Science and Systems 2026
Sydney, Australia, July 13–17, 2026

Self-Supervised Bootstrapping of Action-Predictive Embodied Reasoning

Milan Ganai[♯♫], Katie Luo[♯♫], Jonas Frey[♯♭], Clark Barrett[♯], and Marco Pavone[♯♮]
[♯]Stanford, [♭]UC Berkeley, [♮]NVIDIA
{mganai, katieluo, jonfrey, barrettc, pavone}@stanford.edu

Abstract*—Embodied Chain-of-Thought (CoT) reasoning has significantly enhanced Vision-Language-Action (VLA) models, yet current methods rely on rigid templates to specify reasoning primitives (e.g., objects in the scene, high-level plans, structural affordances). These templates can force policies to process irrelevant information that distracts from critical action-prediction signals. This creates a bottleneck: without successful policies, we cannot verify reasoning quality; without quality reasoning, we cannot build robust policies. We introduce `R&B-EnCoRe`, which enables models to bootstrap embodied reasoning from internet-scale knowledge through self-supervised refinement. By treating reasoning as a latent variable within importance-weighted variational inference, models can generate and distill a refined reasoning training dataset of embodiment-specific strategies without external rewards, verifiers, or human annotation. We validate `R&B-EnCoRe` across manipulation (Franka Panda in simulation, WidowX in hardware), legged navigation (bipedal, wheeled, bicycle, quadruped), and autonomous driving embodiments using various VLA architectures with 1B, 4B, 7B, and 30B parameters. Our approach achieves 28% gains in manipulation success, 101% improvement in navigation scores, and 21% reduction in collision-rate metric over models that indiscriminately reason about all available primitives. `R&B-EnCoRe` enables models to distill reasoning that is predictive of successful control, bypassing manual annotation engineering while grounding internet-scale knowledge in physical execution.

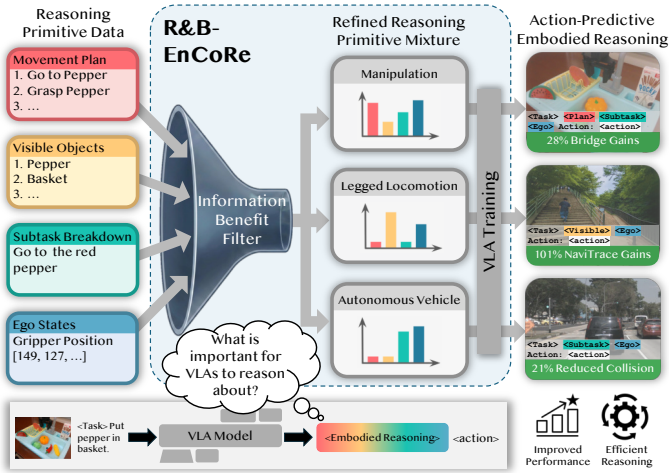


Fig. 1: We generate diverse embodied reasoning primitives and refine them based on action-prediction information benefit. We bootstrap policy performance by retraining on these self-refined, high-quality reasoning traces, discovering embodiment-specific reasoning distributions that reveal effective strategies, significantly improving VLA task success while producing more efficient CoT traces.

I. Introduction

Vision-Language-Action models (VLAs) have begun to successfully harness the momentum of internet-scale foundation models [1] to serve as powerful generalist robot policies [2, 3]. These models inherit vast semantic and visual knowledge from Vision-Language Models (VLMs) pretrained on web-scale image-text corpora [4], and adapt to embodied tasks through continued training on relatively data-scarce robotics demonstrations [5–9]. To transform reactive policies that directly map tasks to actions into models with deeper understanding of the physical world, recent approaches leverage structured Chain-of-Thought (CoT) reasoning that bridges the gap between abstract visual and semantic knowledge and embodied control [10–12]. Specifically, these VLAs learn to output intermediate textual reasoning to elucidate the thinking process of converting high-level intent into low-level control [13].

However, generating effective robotic reasoning traces remains a major bottleneck. Industry currently invests months in manual annotation guidelines to structure visual and spatial details [14]. To scale this labeling process, foundation models are increasingly used to synthetically generate content in visual question answering (VQA) style for reasoning primitives (e.g., objects in the scene, what the planned steps should be) [15–18]. Yet, indiscriminately applying these rich, internet-scale priors for all tasks risks overwhelming embodied reasoning VLAs with irrelevant or distracting information [19–21]. Without principled mechanisms to identify which reasoning primitives predictively inform physical interaction, embodied CoT models often exhibit verbose reasoning that fails to ground internet-scale knowledge in robotics contexts [22]. Thus, a critical challenge remains: *how do we distill visual and semantic knowledge to construct action-predictive embodied reasoning data, reducing the burden of manual CoT engineering* [23–25]*?*

Applying CoT to robotics is hindered by a fundamental embodiment-to-reasoning grounding gap: unlike VLMs pretrained on massive image-text datasets, robotics lacks internet-scale corpora linking vision and language to action, making it difficult to verify what reasoning truly informs control. Consequently, the effective reasoning process for physical action remains an unobserved latent variable—there is no oracle to verify if a thought effectively informs a movement, creating a circular dependency where robust training requires high-quality

[♫]Equal contribution.

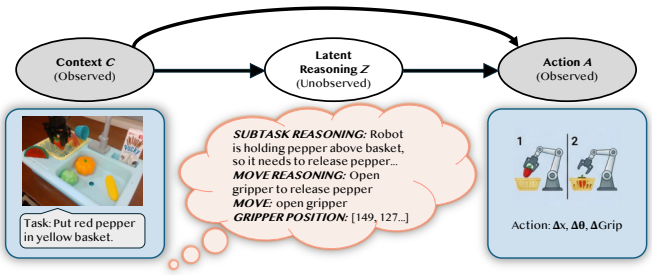


Fig. 2: **Top:** Probabilistic Graphical Model relating the Task Context (C), Reasoning (Z), and Action (A). The latent reasoning Z is induced from a set of primitives $\mathcal{R}$ (e.g., subtask reasoning, move reasoning). **Bottom:** An example reasoning trace on the Bridge setup.

reasoning, yet identifying such reasoning requires an already-successful policy. Current approaches struggle to bridge this gap, often resorting to rigid, "one-size-fits-all" templates that force embodied reasoning models to process irrelevant details—like social norms in an empty scene environment—thereby ignoring the heterogeneity of robotic tasks. To address this, we introduce R&B-EnCoRe (**R**efine and **B**ootstrap **E**mbodimen**t**-specific **C**hain-**o**f-Thought **Re**asoning), a framework that treats reasoning not as a fixed sequence, but as a latent strategy (see Fig. 2) to be discovered and optimized.

Given a training dataset of robotics demonstrations, R&B-EnCoRe leverages synthetic reasoning traces generated from foundation models and addresses the challenge of *unverifiable quality* by formulating embodied reasoning in a variational inference framework. Instead of relying on external rewards, heuristics, or verifiers, our self-supervised approach uses VLA models' generative probabilities to weigh reasoning strategies based on information benefit in expert action prediction. By importance sampling on these weights, R&B-EnCoRe selects refined reasoning traces—pruning confounding or verbose information inherited from web-scale priors—to further bootstrap VLA training. R&B-EnCoRe helps mitigate data scarcity in robotics by distilling informative, high-quality embodied reasoning from internet-scale, multimodal priors, bypassing the need for expensive human annotations.

Through R&B-EnCoRe, we find that VLAs can be distracted by irrelevant information and slowed by the verbosity of exhaustive reasoning—such as enumerating every visible object or always analyzing counterfactual paths—whereas selective reasoning primitives tailored to the embodiment improve both task success and efficiency. Empirically, across manipulation, legged locomotion, and autonomous driving benchmarks, R&B-EnCoRe discovers distinct, interpretable reasoning distributions that bootstrap task success, yielding concise traces that reflect critical signals driving the model's decision-making. Ultimately, R&B-EnCoRe creates potential for self-improving policies that learn not only to act, but to leverage priors to ponder the right questions before acting. Our primary contributions are as follows:

1) We introduce R&B-EnCoRe, a method that formulates embodied reasoning as a latent variable and provides a drop-in training recipe that self-improves from synthetic priors. By utilizing importance-weighted variational inference, our approach enables VLAs to refine and bootstrap reasoning strategies based on their information benefit for action prediction and eliminates "one-size-fits-all" heuristics and expensive human annotations.
2) We demonstrate that R&B-EnCoRe effectively filters out "distractor" information (e.g., bounding boxes of task-irrelevant objects) while amplifying critical signals (e.g., structural affordances for legged robots). This process yields high-quality, interpretable reasoning traces that distinguish essential visual cues from redundant noise without external heuristics, rewards, or verifiers.
3) We validate R&B-EnCoRe on manipulation, legged locomotion, and autonomous driving using various VLA architectures with 1B, 4B, 7B, and 30B parameters. R&B-EnCoRe consistently produces models with action-predictive embodied reasoning, outperforming baseline models like those reasoning on all primitives, with 28% gain in manipulation success, 101% improvement in legged navigation scores, and 21% reduction in autonomous driving collision-rate metric.

II. Related Works

Vision-Language-Action Models. We build on recent advancements in VLA architectures to leverage their inherent multimodal understanding across diverse embodiments. Robot learning has begun to develop generalist policies that integrate visual and semantic representations to operate in unstructured environments [2, 3]. Leveraging foundation models [1] such as Vision-Language Models (VLMs) [17, 26, 27] pre-trained on internet-scale corpora, these approaches adapt semantic and visual priors to embodied tasks [28]. This paradigm treats action prediction as a vision-language problem: models are trained via behavioral cloning on diverse datasets [29–32] to map scene images and task descriptions to discrete action tokens representing expert control [6, 33, 34]. This allows models to inherit the generalization capabilities of foundation models [35] and has been applied to manipulation [5–9, 36–39], legged navigation [39–43], and autonomous driving [44].
Semantic and Visual Reasoning. Chain-of-Thought (CoT) reasoning has enhanced LLM and VLM performance by generating intermediate logical steps before producing a final answer [45, 46]. This computation increases expressivity and search capabilities [47, 48], refining internal representations to better answer complex queries [49, 50] in domains ranging from math and coding to visual question answering [51–54]. Beyond standard prompting, recent efforts explicitly integrate reasoning objectives during pre-training and post-training [55–59], or they improve reasoning and instruction following via supervised finetuning [60–62] or reinforcement learning and self-play [63–66]. Recent works [67–70] leverage Variational Inference to formulate reasoning as a latent variable for language tasks, using external verifiable rewards. Similarly, Expectation-Maximization algorithms have been used to iteratively bootstrap reasoning quality [71, 72]. Our work

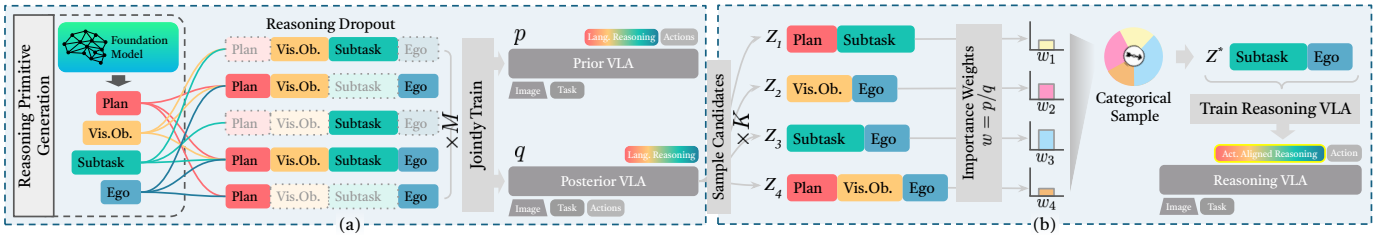


Fig. 3: Overview of R&B-EnCoRe. (a) We generate diverse reasoning primitives (e.g., Plan, Visible Objects) and combine them via dropout to warmstart model capturing prior and posterior distributions. (b) We sample candidates from posterior and apply importance weighting to filter for reasoning that maximizes action prediction power. These refined, high-quality reasoning traces are used to bootstrap the final VLA.

adapts these variational inference techniques to the multimodal robotics domain, enabling VLAs to refine and bootstrap embodied reasoning without external supervision.

Embodied Reasoning. Recent work on generalist policies has extended CoT to robotics by generating intermediate reasoning traces before action prediction [13], finding that such "thinking" enhances generalization and robustness [11]. Prior work employs diverse reasoning primitives: manipulation policies reason about task decomposition [73, 74], end-effector position [75, 76], bounding boxes [77], language motions [10], future frames [78], spatial relations [79–81], and affordances [82–84]. Navigation models similarly have reasoned about perception [85], planning [86], meta-actions [87], failure modes [88], and counterfactuals [89, 90]. These methods typically rely on rigid, structured CoT annotations to ground visual and semantic knowledge [11, 12, 91, 92]. Determining the optimal combination of reasoning primitives for a specific robot and task remains an open challenge; our work formulates reasoning as a latent variable, allowing the VLA to filter irrelevant information and bootstrap the strategies most critical for its specific embodiment.

III. Preliminaries on Variational Inference

In this work, we leverage variational inference to identify effective embodied reasoning for VLA action generation. We briefly describe the latent variable model framework and the Importance Weighted Autoencoder (IWAE), which will serve as the foundation for our algorithm R&B-EnCoRe (see Fig. 3).

A. Latent Variable Models and the Variational Autoencoders

Given a context C (e.g., observation and task), latent variable Z (e.g., textual reasoning) and observed variable A (e.g., action) as in Fig. 2, the objective in Variational Inference is to find a latent distribution that maximizes marginal log-likelihood of the ground truth observed data. For generative model $p(Z, A \mid C) = p(A \mid C, Z)p(Z \mid C)$, which we term the prior distribution, the objective can be written as:

$$\log p(A \mid C) = \log \int p(Z, A \mid C)\, dz.$$

In the rest of the paper, we drop the context C for brevity when its usage as a conditional variable is implicit.

Typically for generative models, computing this integral is intractable. Variational Autoencoders (VAE) address this by introducing a posterior distribution estimate model $q(Z \mid A)$ to approximate the true posterior $p(Z \mid A)$. By applying Jensen's Inequality to the evidence, the Evidence Lower Bound (ELBO) for VAE can be obtained:

$$\log p(A) = \log \mathop{\mathbb{E}}_{Z\sim q}\left[\frac{p(Z,A)}{q(Z \mid A)}\right] \geq \underbrace{\mathop{\mathbb{E}}_{Z\sim q}\left[\log \frac{p(Z,A)}{q(Z \mid A)}\right]}_{\text{ELBO}_{\text{VAE}}}.$$

This ELBO is composed of computationally tractable terms. Note, the difference between ELBO_{VAE} and the true evidence is the Kullback-Leibler (KL) divergence between the approximate and true posterior: $D_{KL}(q(Z \mid A) \,\|\, p(Z \mid A))$ [93].

B. Importance Weighted Autoencoders (IWAE)

The IWAE framework [94, 95] improves on VAE by creating an importance sampling estimate of the evidence via multiple samples. We can draw K i.i.d. samples $Z_1, \ldots, Z_K$ from the posterior distribution $q(Z \mid A)$. The K-sample importance-weighted lower bound is defined as:

$$\mathcal{L}_K \doteq \mathbb{E}_{Z_1,\ldots,Z_K\sim q(Z|A)}\left[\log\left(\frac{1}{K}\sum_{k=1}^{K}\frac{p(Z_k,A)}{q(Z_k \mid A)}\right)\right].$$

To see that this is a lower bound on $\log p(A)$, we observe that the expectation of the term inside the log is an unbiased estimator of the evidence:

$$\begin{aligned}\mathop{\mathbb{E}}_{Z_{1:K}\sim q}\left[\frac{1}{K}\sum_{k=1}^{K}\frac{p(Z_k,A)}{q(Z_k \mid A)}\right] &= \frac{1}{K}\sum_{k=1}^{K}\mathop{\mathbb{E}}_{Z_k\sim q}\left[\frac{p(A,Z_k)}{q(Z_k \mid A)}\right] \\ &= \frac{1}{K}\sum_{k=1}^{K}\int\frac{p(A,Z_k)}{q(Z_k \mid A)}q(Z_k \mid A)dZ_k = p(A).\end{aligned}$$

Using Jensen's Inequality on the concave log function yields:

$$\log p(A) = \log \mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K} w_k\right] \geq \mathbb{E}\left[\log \frac{1}{K}\sum_{k=1}^{K} w_k\right] = \mathcal{L}_K,$$

where $w_k = \frac{p(Z_k,A)}{q(Z_k|A)}$ are importance weights. For $K > 1$, [94] shows $\mathcal{L}_{K+1} \geq \mathcal{L}_K \geq \mathcal{L}_1 = \text{ELBO}_{\text{VAE}}$, so by increasing sample count, IWAE theoretically improves evidence estimate. In the next section, we propose our algorithm which is based on a sampling-importance-resampling technique introduced in [95] that improves upon IWAE by estimating the posterior distribution using a categorical distribution of the importance weights. More details on the theory behind this technique can be found in Appendix C.

Algorithm 1 R&B-EnCoRe: Warmstarting

Require: Dataset $\mathcal{D} = \{(C^i, A^i)\}_{i=1}^N$, Reasoning Primitives $\mathcal{R}$, Dropout rate d, VLM $\mathcal{M}$, Foundation Model FM

$\mathcal{D}_{\text{warm}} \leftarrow \emptyset$
for each $(C^i, A^i) \in \mathcal{D}$ **do**
  $\{z_R\}^i_{R\in\mathcal{R}} \leftarrow \text{Query}(\text{FM}, (C^i, A^i), \mathcal{R})$
  for $j = 1$ **to** M **do**
    Construct reasoning trace $Z^i_j \subseteq \{z_R\}^i_{R\in\mathcal{R}}$ by including each $z \in \{z_R\}^i_{R\in\mathcal{R}}$ with probability $1 - d$
    w.p. 0.5 : $\mathcal{D}_{\text{warm}} \leftarrow \mathcal{D}_{\text{warm}} \cup \{(Z^i_j, A^i \mid C^i)\}$
    o.w. : $\mathcal{D}_{\text{warm}} \leftarrow \mathcal{D}_{\text{warm}} \cup \{(Z^i_j \mid C^i, A^i)\}$
  end for
end for
$\mathcal{M}_{pq} \leftarrow \text{Train}(\mathcal{M}, \mathcal{D}_{\text{warm}})$
return $\mathcal{M}_{pq}$

Algorithm 2 R&B-EnCoRe: Refinement & Bootstrapping

Require: Dataset $\mathcal{D} = \{(C^i, A^i)\}_{i=1}^N$, $\mathcal{M}_{pq}$, VLM $\mathcal{M}$

$\mathcal{D}_{\text{refined}} \leftarrow \emptyset$
for each $(C^i, A^i) \in \mathcal{D}$ **do**
  Sample K candidates $\{Z^i_k\}_{k=1}^K \sim q(\cdot|C^i, A^i)$
  for $k = 1$ to K **do**
    Compute weight $w(Z^i_k) \leftarrow \frac{p(Z^i_k, A^i|C^i)}{q(Z^i_k|C^i, A^i)}$
  end for
  Sample reasoning trace $Z^{i*} \sim \text{Cat}(w(Z^i_k))$
  $\mathcal{D}_{\text{refined}} \leftarrow \mathcal{D}_{\text{refined}} \cup \{(Z^{i*}, A^i \mid C^i)\}$
end for
$\mathcal{M}_{\text{VLA}} \leftarrow \text{Train}(\mathcal{M}, \mathcal{D}_{\text{refined}})$
return $\mathcal{M}_{\text{VLA}}$

IV. R&B-EnCoRe Framework

We propose our approach, R&B-EnCoRe, which treats embodied reasoning not as a fixed annotation, but as a latent variable Z that explains the relationship between observation and task context C and physical action A (Fig. 2).

A. Warmstarting Strategy Hypotheses via Reasoning Dropout

While foundation models possess the capability to reason about diverse aspects of a physical scene, we lack ground-truth data determining which specific signals are useful for low-level robotic control. We hypothesize a set of ρ potentially relevant reasoning primitives $\mathcal{R} = \{R_1, \ldots, R_\rho\}$ (e.g., high-level plans, visual objects, subtask breakdowns, ego state) and extract textual explanations for each by posing them as VQA tasks to Foundation Models or using human annotated datasets.

To discover the optimal combination of these primitives for specific tasks, we must expose the model to various strategies $\mathbf{R} \subseteq \mathcal{R}$ of combinations during training rather than a single fixed template. To this end, we construct a "warmstart" dataset $\mathcal{D}_{\text{warm}}$ by sampling from the powerset of reasoning primitives. Specifically, for each demonstration (C, A), we generate M synthetic traces $Z_j,\ j \in \{1 \cdots M\}$ using Reasoning Dropout (refer to Alg. 1). Each primitive $R_r \in \mathcal{R}$ is independently included or dropped with a fixed probability d. This mechanism exposes the model to various reasoning strategies—ranging from concise to verbose—providing diversity to our refining stage to identify signals that improve action prediction.

B. Jointly Training Prior and Posterior

We utilize these diverse traces to train a Vision-Language-Action (VLA) model that serves as the posterior and prior roles simultaneously, mirroring the encoder-decoder structure of variational autoencoders but adapted for embodied reasoning (Fig. 3a). The prior model represents the agent's ability to generate reasoning and actions at test time, while the posterior model represents the ability to explain actions in hindsight. Specifically, the prior is trained on the warmstart dataset $\mathcal{D}_{\text{warm}}$ to generate reasoning strategies with correct actions, and the posterior is trained on the same dataset to propose reasoning candidates conditioned on actions:

Prior Model $p(Z, A \mid C)$: Trained on the sequences $\{(Z_j, A \mid C)\}$ by conditioning on the context tokens. This represents the online policy: it observes context C, generates reasoning Z_j, and predicts action A. This distribution captures the predictive power of reasoning for control.

Posterior Model $q(Z \mid C, A)$: Trained on the sequences $\{(Z_j \mid C, A)\}$ by conditioning on context *and action* tokens. By conditioning on task C and ground-truth action A, the posterior learns to generate diverse reasoning Z_j, explaining how action A is the correct response, acting as a proposal distribution from which we sample action-relevant reasoning.

C. Refining and Bootstrapping via Importance Sampling

We refine VLA reasoning through a two-step process: first, we use our trained prior and posterior model to generate and refine reasoning traces through importance sampling; second, we retrain a VLA with these more action-predictive embodied reasoning traces.

Stage 1: Information-Beneficial Reasoning (Refine). We refine the reasoning-enriched data by identifying strategies that improve action prediction through importance weighting (Alg. 2). For each demonstration (C, A), we sample K candidate reasoning traces $\{Z_k\}_{k=1}^K \sim q(\cdot \mid C, A)$ from the trained posterior. For each trace Z_k, we compute importance weight:

$$w(Z_k) = \frac{p(Z_k, A \mid C)}{q(Z_k \mid C, A)}.$$

Then, we resample a single trace Z^* from these candidates according to a categorical distribution proportional to the importance weights $\text{Cat}(w(Z_k))$. Intuitively, this process yields reasoning traces with a natural interpretation—selecting a reasoning primitive for inclusion in the final strategy indicates that it provides information about the correct action. To formalize this, we define the *information benefit* of a reasoning strategy $\mathbf{R}$ as how much it reduces the divergence between our model's action distribution and the expert's distribution $p_{data}(A|C)$:

$$\Delta\mathcal{I}_{\mathbf{R}} \doteq D_{\text{KL}}(p_{\text{data}} \| p(A|C, \mathcal{Z}_{\not\mathbf{R}})) - D_{\text{KL}}(p_{\text{data}} \| p(A|C, \mathcal{Z}_{\mathbf{R}})),$$

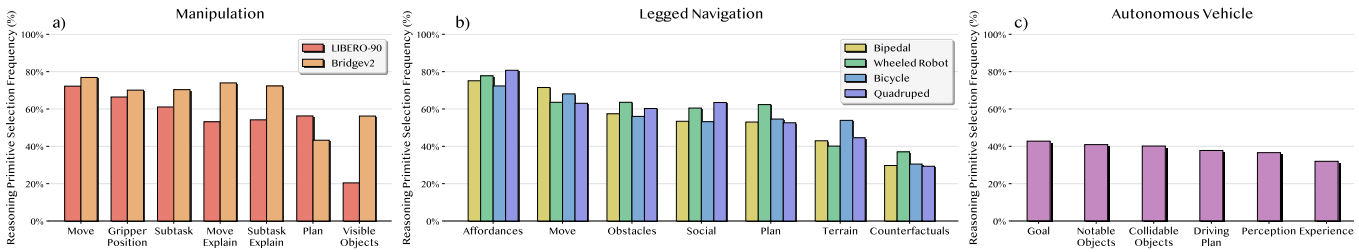


Fig. 4: This plot shows the reasoning primitives distributions that are generated from R&B-EnCoRe refining warmstarting diverse reasoning strategy data. In a) the distribution for manipulation shows differences between reasoning for Franka Panda in simulation versus WidowX hardware in real-world data, notably for Visible Object, Move Explain, and Subtask Explain reasoning primitives. In b) we observe that the four-legged locomotion embodiments we investigate benefit in similar frequencies across reasoning types, with structural affordances being critical. For autonomous vehicles, we find in c) that reasoning focuses on goals and constraints.

where $\mathcal{Z}_{\mathbf{R}}$ and $\mathcal{Z}_{\not\mathbf{R}}$ denote the set of traces with and without strategy $\mathbf{R}$. Our importance weighting estimates this quantity:

Proposition (Importance Weight Ratios Estimate Information Benefit)**.** *Under the training (Alg. 1) and sampling (Alg. 2) procedures, the expected log-ratio of importance weights equals the information benefit:*

$$\mathbb{E}_{A\sim p_{data}}\left[\log \frac{\mathbb{E}_{Z\sim q}[w(Z_{\mathbf{R}}) \mid Z_{\mathbf{R}} \in \mathcal{Z}_{\mathbf{R}}]}{\mathbb{E}_{Z\sim q}[w(Z_{\not\mathbf{R}}) \mid Z_{\not\mathbf{R}} \in \mathcal{Z}_{\not\mathbf{R}}]}\right] = \Delta\mathcal{I}_{\mathbf{R}}.$$

Proof in Appendix B.

More intuitively, R&B-EnCoRe automatically amplifies reasoning strategies that improve action prediction ($\Delta\mathcal{I}_{\mathbf{R}} > 0$) while suppressing distracting ones ($\Delta\mathcal{I}_{\mathbf{R}} < 0$) in a self-supervised manner. The result is a refined dataset of concise, action-predictive reasoning traces tailored to the specific embodiment and task.

Stage 2: Training Action-Aligned Reasoning (Bootstrap). We bootstrap VLA performance by retraining the model on the filtered, high-quality reasoning dataset $\{(Z^*, A \mid C)\}$ (Fig. 3b). This yields a policy that reasons about information beneficial for control. By training on reasoning traces selected for their information benefit, the final VLA policy learns from examples where the reasoning is aligned with task success. This mitigates noise introduced by training on all reasoning primitives, pruning those that distract from predicting expert action. Specifically, the model learns from the refined data robust internal representations that more effectively map context to action via task-relevant intermediate reasoning.

V. Experiments

We report the performance of R&B-EnCoRe across robotic manipulation (LIBERO-90 and real-world WidowX) in Sections V-A and V-B, legged navigation (bipedal, wheeled robot, bicycle, quadruped) in Section V-C, and autonomous driving in Section V-D. Our evaluation shows that by treating embodied reasoning as a latent variable, VLAs can refine informative reasoning traces with high action-predictive power from internet-scale priors, without external heuristics, rewards, or verifiers. For each embodiment, our primary baselines are VLAs with no reasoning, VLAs trained with all reasoning primitives, and the prior VLAs with random reasoning primitives. We evaluate in the following experiments how R&B-EnCoRe guides VLAs to self-refine their internal decision-making process to align more closely with expert control and ultimately bootstrap performance. We detail our empirical results, structured to evaluate R&B-EnCoRe's ability to refine reasoning distributions and improve task success in a self-supervised manner.

A. LIBERO-90 Franka Panda Manipulation

We evaluate our approach on the LIBERO-90 [96] manipulation benchmark by training a 1B-parameter MiniVLA [6], which combines a 0.5B Qwen2.5 LLM [97] backbone with DINOv2 [98] and SigLIP [99] vision encoders. Applying R&B-EnCoRe, we use a reasoning dropout rate of $d = 0.2$. The context C includes the scene image and task description. Following [12], the latent variable Z includes at most seven reasoning primitives generated by Llama 2 [15] and Molmo [16]: Plan, Visible Objects, Subtask, Subtask Explain, Move, Move Explain, and Gripper Position (see Appendix D-A for details). Action A is seven discrete tokens encoding a 10-step action chunk [100] via VQ-VAE [101]. More details on LIBERO experiments are in Appendix D-A.

Q1 How effective is R&B-EnCoRe at identifying task-salient perception reasoning in manipulation, i.e., discerning action-critical objects, without external supervision?

We validate whether R&B-EnCoRe can discern relevant perceptual signals in LIBERO-90. We first apply our framework exclusively on the Visible Objects reasoning primitive. Using the architecture described in Section V-A, we restrict the reasoning to subsets of visible objects. We construct warmstart data by applying a dropout rate of $d = 0.2$ to the ground-truth object list, generating traces with random subsets of the scene's objects. This forces the model to explore strategies ranging from sparse to exhaustive object enumeration.

As shown in Table I, refining visible objects significantly improves performance, outperforming baselines that reason with the full object list, random subsets, or no perception reasoning. This suggests that while perception is useful, exhaustive enumeration of every visible object can introduce

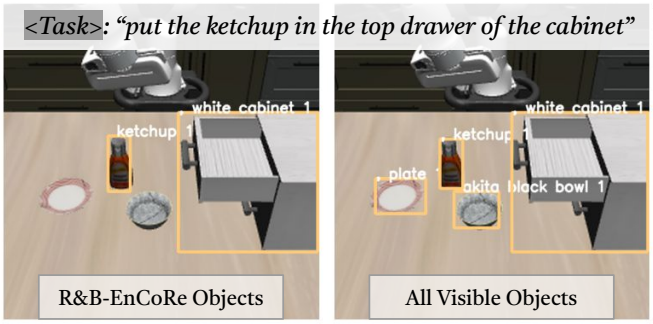


Fig. 5: Visible Objects generated in LIBERO-90 by R&B-EnCoRe's model and a model producing a full list. The latter model attends to task-irrelevant objects like plate and bowl, while our model emits reasoning focused on task-critical objects (refer to Appendix G for full qualitative comparisons).

noise that distracts from the immediate control task. To analyze refinement quality, we measure the *Object Criticality Rate*: the percentage of traces where every listed object is task-salient, i.e., explicitly mentioned in the a separate dataset of reasoning primitives (e.g., Plan, Move). Crucially, these reference primitives act as a separate validation source and were *not* used to train the models in this experiment, serving as an independent proxy for functional relevance. We find that the baseline model (listing all objects) almost never produces traces (0.03%) with only task-salient objects. In contrast, R&B-EnCoRe refines embodied CoT to prioritize task-relevant signals, increasing criticality rate to $> 25\%$ and filtering irrelevant objects without external guidance (Fig. 5).

Q2 How does R&B-EnCoRe refine verbose reasoning primitives in manipulation to improve task success?

Applying R&B-EnCoRe to refine a wider set of reasoning primitives in LIBERO-90, we see in Table II that R&B-EnCoRe achieves higher success over other reasoning strategies while reducing average token count by half compared to reasoning with all primitives. In Fig. 4a, we observe R&B-EnCoRe yields a refined training distribution of traces that prioritizes Move, Gripper Position, and Subtask reasoning primitives (Fig. 4a). Notably, the model assigns high importance to Move primitives—short meta-action language descriptions. This confirms research findings [10, 87] that generating these meta-action descriptions provides critical, nonredundant information that improves expert action prediction. We observe that R&B-EnCoRe favors the concise Subtask/Move over the more verbose Subtask/Move Explain, as it prunes redundant justifications lacking information benefit in action prediction. Additionally, the Visible Objects primitive appears with low frequency (20%) in the final distribution. This aligns with our findings in Q1: listing all objects is generally ineffective.

B. Bridge WidowX Hardware

We evaluate our approach on WidowX hardware by training with the Bridge v2 Dataset [30] a 7B-parameter OpenVLA [6], which combines a Llama 2 [97] backbone with DINOv2 [98] and SigLIP [99] vision encoders. With R&B-EnCoRe, we use a reasoning dropout rate of $d = 0.2$, with the same 7 primitives as in Section V-A, where the reasoning content was generated in [11] by the Gemini 1.0 [75] model. The action A consists of seven discrete tokens based on the tokenization of [5, 37]. To support high-throughput repeated sampling when generating our synthetic reasoning candidate for refinement, we extended SGLang-VLA [102, 103] to support rapid inferencing of reasoning-based [11, 12] Prismatic Models [5, 28]. More details on Bridge experiments are in Appendix D-B.

TABLE I: R&B-EnCoRe autonomously identifies critical objects in LIBERO-90 data that are most predictive of expert actions, pruning irrelevant objects and improving task success.

Training Strategy	Success Rate	Object Criticality Rate
No Reasoning	75.9%	N/A
List of All Objects	76.1%	0.03%
List of Random Objects	77.0%	3.43%
R&B-EnCoRe on Object List	**80.3%**	**25.02%**

TABLE II: Results on analyzing reasoning primitives in LIBERO-90: Plan, Visible Objects, Subtask, Subtask Explain, Move, Move Explain, and Gripper Position. R&B-EnCoRe refines action-predictive reasoning, resulting in shorter traces and improved success rate.

Training Strategy	Success Rate	Avg. # Gen. Token
No Reasoning	75.9%	10.0
All Primitives	78.6%	256.8
Random Primitives	76.5%	256.6
R&B-EnCoRe	**79.5%**	129.3

Q3 How effective can R&B-EnCoRe enable manipulation VLAs to generalize when test-time reasoning is suppressed for speed, compared to reasoning with all primitives?

Since generating textual reasoning traces at test-time can drastically increase latency (often taking several seconds per step), we first investigate whether VLAs can benefit from reasoning during training while bypassing the computational cost of generation during deployment. To this end, we employ "Action Forcing," where VLAs are prompted with an Action tag to immediately generate action tokens, effectively suppressing the reasoning output. For models trained with R&B-EnCoRe or random primitives, this capability is emergent: because some of their training data has no reasoning content due to dropout, these models can zero-shot generalize to the Action Forcing prompt. In contrast, the VLA trained with all reasoning primitives does not have reasoning-free training data; hence, we post-train this model for some epochs with no reasoning to acquire this ability, following the findings in [12].

We evaluated four models across 9 tasks totaling 468 trials, categorized into: In-Distribution, OOD Target (novel target objects), and OOD Scene/Distractions (cluttered environments with distracting objects). Task details can be found in the Appendix E. As seen in our results in Table III, for In-Distribution tasks, both R&B-EnCoRe and all reasoning primitive models outperform no reasoning and random reasoning primitives. This result confirms that training on valid reasoning

TABLE III: Success rates on WidowX hardware Bridgev2 setup. Reasoning Models are prompted and/or trained with Action forcing for reduced latency [12]. In total for the experiments presented in this table, we had 4 models $\times$ 9 tasks $\times$ 13 trials $=$ 468 total trials.

Category	Task	No Reason	All Primitives	Random	R&B-EnCoRe
In Distrib.	put red pepper in yellow basket	69.2%	**100.0%**	92.3%	**100.0%**
	put red pepper on black stove	53.8%	**92.3%**	46.2%	84.6%
	put orange carrot on pink plate	61.5%	76.9%	**84.6%**	**84.6%**
OOD Target Object	put blue peacock in sink	46.2%	38.5%	69.2%	**76.9%**
	put orange tape on green towel *(include orange carrot)*	38.5%	38.5%	38.5%	**76.9%**
	put pink pepto on green towel	46.2%	30.8%	53.8%	**69.2%**
OOD Scene with Distracting Objects	put yellow corn on blue plate *(include pink plate, carrot)*	53.8%	69.2%	38.5%	**76.9%**
	put orange carrot in yellow basket *(include distraction objects in basket, sink)*	61.5%	**92.3%**	84.6%	**92.3%**
	first put yellow corn in yellow basket then put red pepper in yellow basket *(human takes corn)*	46.2%	30.8%	46.2%	**61.5%**

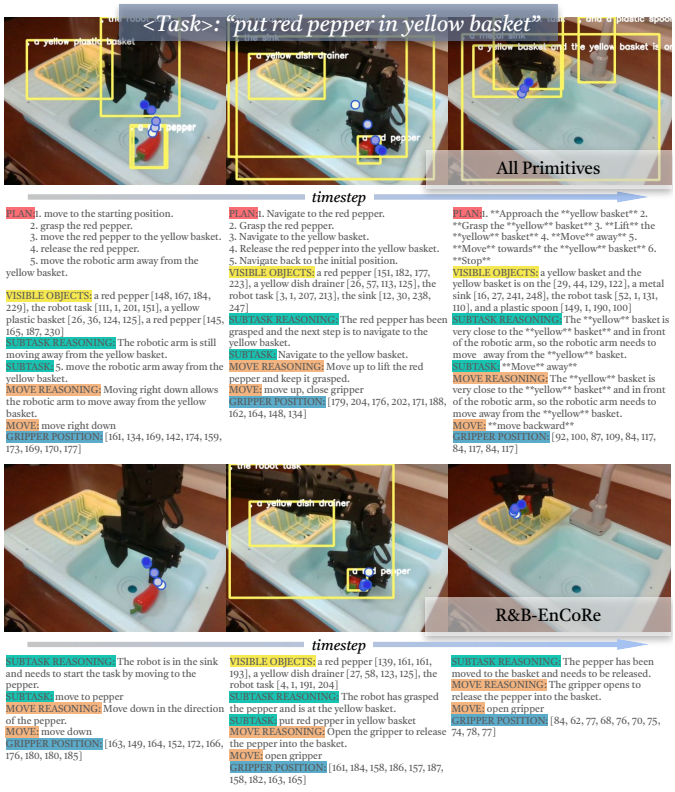


Fig. 6: Sample reasoning traces on WidowX hardware. R&B-EnCoRe produces more concise and effective reasoning compared to reasoning on all primitives.

traces improves the underlying policy representation, even when that reasoning is not explicitly generated at test-time. For both OOD task groups, reasoning with all primitives suffers a performance drop, suggesting that rigid, "one-size-fits-all" templates force the baseline model to attend to specific tasks or scene features (e.g., OOD distracting objects with the same color as the target) that may be irrelevant in OOD settings, confounding the policy. R&B-EnCoRe's model is robust to these shifts, having learned to prune irrelevant signals, and maintains its success rate. Overall, our approach applied

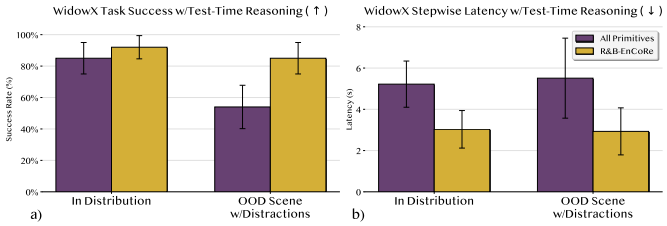


Fig. 7: Success rates and latency of test-time reasoning on WidowX hardware. R&B-EnCoRe produces performant reasoning VLAs with shorter reasoning traces (so faster inference). Reasoning on all primitives degrades performance for cluttered scenes with OOD objects.

in manipulation VLAs on real-world hardware consistently demonstrates that refining the reasoning distribution during training aligns the latent representation with successful task performance without needing to sacrifice test-time speed.

> **Q4** How does R&B-EnCoRe improve task performance and reduce test-time reasoning latency compared to baseline reasoning on all primitives for WidowX hardware?

We perform an ablation study evaluating the performance and latency of explicit test-time reasoning on the WidowX robot, comparing R&B-EnCoRe against the baseline trained on all primitives. We provide traces from a trajectory generated by models reasoning with all primitives and our approach in Fig. 6. Results in Fig. 7a show that while in-distribution performance is comparable, R&B-EnCoRe significantly outperforms the baseline in OOD settings with distractions, where the latter's performance degrades by 31%. This suggests R&B-EnCoRe effectively filters irrelevant strategies that otherwise yield confounding traces in novel environments. Furthermore, Fig. 7b demonstrates that R&B-EnCoRe reduces inference time from > 5 seconds to ≈ 3 seconds per step. We observed that slow generation creates control lag, often causing grasped objects to slip during the trajectory. By pruning verbose reasoning, our approach generates concise traces with lower test-time latency and improved task success.

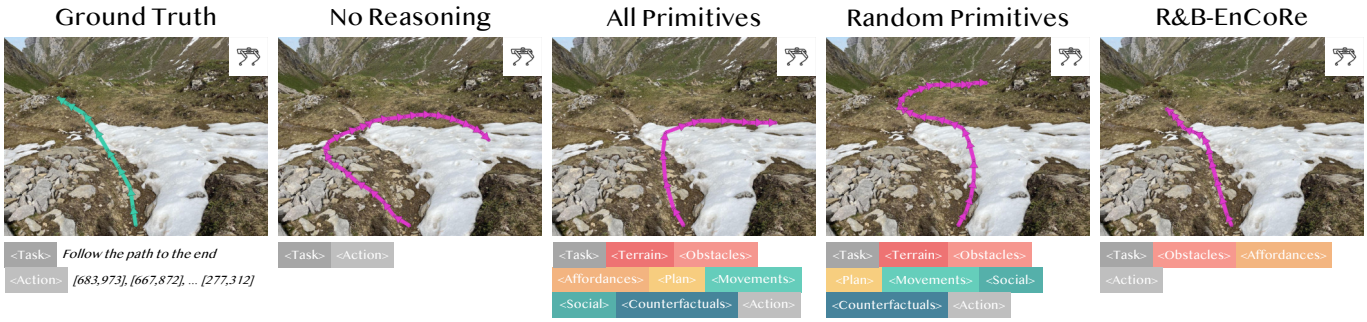


Fig. 8: Quadruped Navigation Waypoint Trajectories. The quadruped robot must follow the trail while avoiding slippery ice. No Reason navigation VLA ignores terrain hazards and traverses the ice. Reasoning with all primitives is confounded by irrelevant signals; while it has reduced ice contact (perhaps due to affordance reasoning), it fails to follow the path. Random Primitives tracks some of the path but likely due to lack of affordance reasoning, it traces through a lot of the slippery ice. R&B-EnCoRe identifies the effective reasoning strategy, with minimal ice contact while maintaining the path, matching the Ground Truth. This example was taken from holdout task set.

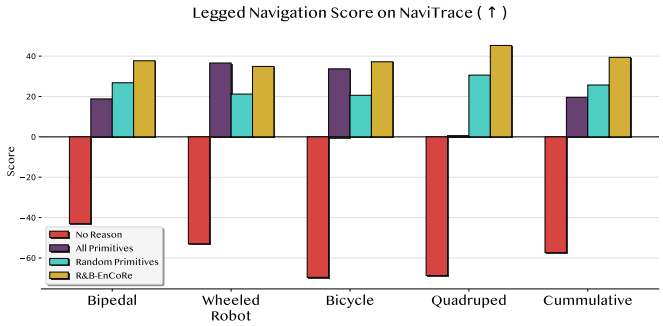


Fig. 9: The NaviTrace score is a normalized metric so 100 is perfect path alignment with expert, and 0 is the score for the naïve straight line path down the middle of the scene. R&B-EnCoRe generally performs best across embodiments on navigation metric, improving the cumulative score from 19.6 (all primitives) to 39.4 (R&B-EnCoRe). For comparison, native CoT of the Qwen3-VL-30b model (i.e., querying zero-shot) achieves a score of −260.

C. Legged Robots Navigation

We evaluate the capabilities of R&B-EnCoRe in the context of legged robot navigation using the NaviTrace dataset [104], which contains approximately 1000 tasks across 500 unique scenes for four embodiments: bipedal, wheeled, bicycle, and quadruped robots. Evaluation is performed on a holdout subset of these tasks and scenes using a downstream task metric from [104] that incorporates Dynamic Time Warping distance, goal endpoint error, and semantic penalties correlating with human preferences. We finetune a Qwen3-VL-30B-A3B-Instruct [17] Mixture-of-Experts model to process scene images and task specifications, outputting 2D waypoint coordinates. To enable fully self-reliant learning, we construct the reasoning dataset by querying the VLM itself via VQA (so FM $= \mathcal{M}$ in Alg. 1) to generate traces for seven hypothesized reasoning primitives, e.g., terrain, affordances, social norms.

Q5 How does R&B-EnCoRe self-bootstrap legged locomotion navigation VLA performance when refining and learning from reasoning generated by its base VLM?

By applying R&B-EnCoRe with dropout $d = 0.5$ to these self-generated traces, the model refines its own reasoning distribution to improve the downstream navigation score (Fig. 9). As in Fig. 4b, we observe that the refined distribution heavily prioritizes reasoning about actionable capabilities of the environment (Affordances) and proposed movements (Move), identifying these as critical for successful navigation. We also observe that counterfactual reasoning is not always necessary, dropping significantly in frequency, confirming recent findings [89, 90, 105, 106] that counterfactual reasoning is only reserved for specific decision moments. These results provide evidence for significant promise that frontier VLMs can self-bootstrap performance by synthetically generating VQAs on embodied reasoning and self-refining their data.

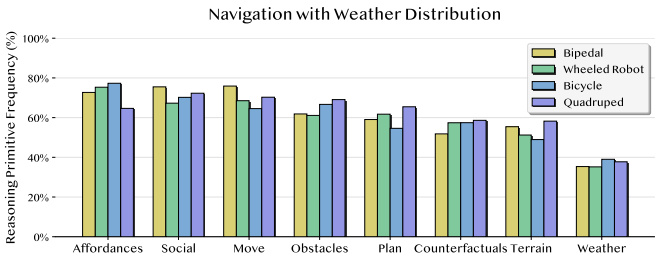


Fig. 10: R&B-EnCoRe prunes uninformative subjective weather reasoning from refined traces (∼36.7%; lower than other primitives).

Q6 How effective is R&B-EnCoRe in pruning patently irrelevant reasoning primitives like subjective descriptions of weather for legged navigation?

We introduce an uninformative reasoning primitive, subjective description of the weather, to the training data. Analyzing the reasoning distribution produced by R&B-EnCoRe in Fig. 10, we see our approach successfully identifies "weather" descriptions as largely irrelevant to the navigation task, pruning this primitive significantly more aggressively than others. Conversely, the model retains reasoning related to obstacles, affordances, and social norms. This refined primitive distribution provides evidence that R&B-EnCoRe can effectively filter out patently irrelevant information, generating high-quality relevant reasoning strategies tailored to the specific needs of various embodiments of legged locomotion navigation.

TABLE IV: Performance comparison of AV with UniAD metrics. R&B-EnCoRe is able to refine reasoning traces and reduce deviation from ground truth path and collision rate. For comparison, zero-shot native chain-of-thought achieves suboptimal 10.35m average L2 error.

	L2 Path Error (m)				Collision Rate (%)			
Method	**1s**	**2s**	**3s**	**Avg**	**1s**	**2s**	**3s**	**Avg**
No Reasoning	0.22	0.62	1.34	0.72	0.10	0.25	1.11	0.49
Full Reasoning	0.22	0.64	1.33	0.73	0.07	**0.17**	0.91	0.38
Random	0.22	0.62	1.31	0.72	**0.05**	0.20	0.81	0.35
R&B-EnCoRe	**0.21**	**0.59**	**1.25**	**0.68**	**0.05**	**0.17**	**0.70**	**0.30**

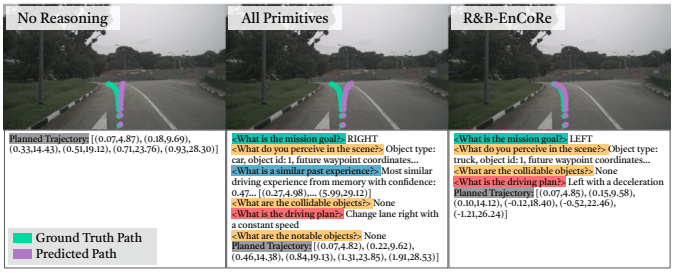


Fig. 11: Planned trajectories comparing driving VLAs using reasoning by R&B-EnCoRe's model and a model producing a full list.

D. Autonomous Vehicles

We extend R&B-EnCoRe to the autonomous vehicle (AV) nuScenes dataset [32] and study how reasoning traces for driving VLAs changes under our method. We leverage traces from pioneering LLM agent-based planners [85, 86], repurposed for training the reasoning component of a driving VLA. Specifically, we finetune a Qwen3-VL-4B-Instruct Dense Model [17] to take in the front camera image, and output the ego-vehicle's planning trajectory over 3 seconds. We report evaluation using the UniAD [107] planning metrics on the held-out validation set (refer to Appendix D-D for evaluation details). In this section, we aim to refine these commonly used reasoning traces to improve downstream performance.

Q7 How does R&B-EnCoRe further refine human-crafted reasoning for Autonomous Vehicle data?

To evaluate R&B-EnCoRe with reasoning dropout of $d = 0.5$ against handcrafted reasoning traces, we compare our method to using all reasoning primitives from [86], a random reasoning strategy, and no reasoning (Table IV). While all reasoning components reduce collision rate, they slightly increase L2 error compared to no reasoning, suggesting irrelevant information may harm trajectory prediction performance. R&B-EnCoRe is able to reduce irrelevant components (Fig. 4c) and subsequently improve downstream trajectory prediction (Fig. 11), suggesting that the embodied domain of autonomous vehicle VLAs can benefit from reasoning trace refinement, which has traditionally been handcrafted [87, 108].

Q8 How does R&B-EnCoRe's performance for AV scale with respect to the number of posterior samples?

To understand how performance scales with the number of posterior samples, we ablate the variational inference parameter K in Fig. 12. We observe that increasing the number of posterior samples improves performance on the collision rate metric, with performance saturating around 16 samples, substantiating that the multi-sampling IWAE process used by R&B-EnCoRe yields tighter bounds with increased samples.

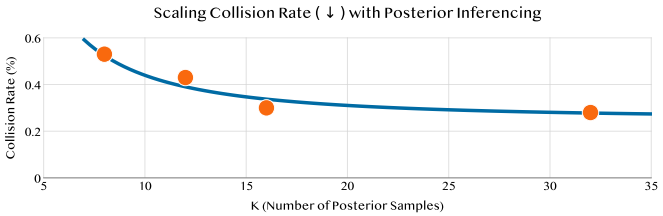


Fig. 12: Collision Rate scaling with posterior inference. More samples K from posterior distribution results in improved action prediction estimate, and ultimately lower collision rate. Scaling curve fitted with Collision Rate $= (3.65/K)^{1.65} + 0.25$.

VI. Discussion and Conclusion

We introduced R&B-EnCoRe, which treats embodied reasoning as a latent variable to ground internet-scale priors in physical control. Using importance-weighted variational inference, R&B-EnCoRe autonomously filters chain-of-thought traces to identify expert action-predictive, embodiment-specific reasoning without external supervision. Across manipulation, legged navigation, and autonomous driving, our method significantly outperforms fixed baselines. Notably, R&B-EnCoRe's effectiveness improves with the diversity and coverage of warmstart primitives rather than their individual quality: action-uninformative primitives are automatically down-weighted by the importance-sampling step, and in the limit where no primitive carries predictive signal the variational objective (see Appendix B and C) converges to an empty reasoning trace. While our approach trains an additional prior-posterior distribution model, the posterior effectively functions as a high-quality synthetic data generator, allowing for controlled, reward-free exploration of reasoning strategies that can further enhance policy robustness. Future research can investigate applying this framework to continual learning settings, as well as prompting mechanisms that are more native to the base VLM architecture, viewing the posterior as an answer explainer and the prior as a solution reasoning-answer generator. Such approach could further improve R&B-EnCoRe scalability for larger models where warmstart training may be computationally prohibitive.

Acknowledgments

This work was supported by NASA University Leadership Initiative, Toyota Research Institute (TRI), Defense Advanced Research Projects Agency (DARPA), Stanford Center for Automated Reasoning, Stanford Marlowe GPU Cluster [109], Stanford Google Cloud Platform, DoD High Performance Computing Modernization Program, Thinking Machines Lab, the National Artificial Intelligence Research Resource Pilot and the Anvil supercomputer (award NSF-OAC 2005632). This article solely reflects the opinions and conclusions of its authors and not of the aforementioned supporting entities.

References

[1] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[2] Kento Kawaharazuka, Jihoon Oh, Jun Yamada, Ingmar Posner, and Yuke Zhu. Vision-language-action models for robotics: A review towards real-world applications. *IEEE Access*, 2025.

[3] Yifan Zhong, Fengshuo Bai, Shaofei Cai, Xuchuan Huang, Zhang Chen, Xiaowei Zhang, Yuanfei Wang, Shaoyang Guo, Tianrui Guan, Ka Nam Lui, et al. A survey on vision-language-action models: An action tokenization perspective. *arXiv preprint arXiv:2507.01925*, 2025.

[4] Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. Vision-language models for vision tasks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 46(8):5625–5644, 2024.

[5] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P Foster, Pannag R Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. Openvla: An open-source vision-language-action model. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard, editors, *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pages 2679–2713. PMLR, 06–09 Nov 2025.

[6] Suneel Belkhale and Dorsa Sadigh. Minivla: A better vla with a smaller footprint, 2024. URL https://github.com/Stanford-ILIAD/openvla-mini.

[7] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, Laura Smith, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A Vision-Language-Action Flow Model for General Robot Control. In *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025. doi: 10.15607/RSS.2025.XXI.010.

[8] Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, brian ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization. In Joseph Lim, Shuran Song, and Hae-Won Park, editors, *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pages 17–40. PMLR, 27–30 Sep 2025.

[9] Physical Intelligence, Ali Amin, Raichelle Aniceto, Ashwin Balakrishna, Kevin Black, Ken Conley, Grace Connors, James Darpinian, Karan Dhabalia, Jared DiCarlo, et al. $\pi_{0.6}$: a vla that learns from experience. *arXiv preprint arXiv:2511.14759*, 2025.

[10] Suneel Belkhale, Tianli Ding, Ted Xiao, Pierre Sermanet, Quan Vuong, Jonathan Tompson, Yevgen Chebotar, Debidatta Dwibedi, and Dorsa Sadigh. RT-H: Action Hierarchies using Language. *Proceedings of Robotics: Science and Systems*, July 2024. doi: 10.15607/RSS.2024.XX.049.

[11] Michał Zawalski, William Chen, Karl Pertsch, Oier Mees, Chelsea Finn, and Sergey Levine. Robotic control via embodied chain-of-thought reasoning. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard, editors, *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pages 3157–3181. PMLR, 06–09 Nov 2025.

[12] William Chen, Suneel Belkhale, Suvir Mirchandani, Karl Pertsch, Danny Driess, Oier Mees, and Sergey Levine. Training strategies for efficient embodied reasoning. In Joseph Lim, Shuran Song, and Hae-Won Park, editors, *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pages 365–391. PMLR, 27–30 Sep 2025.

[13] Gemini Robotics Team, Abbas Abdolmaleki, Saminda Abeyruwan, Joshua Ainslie, Jean-Baptiste Alayrac, Montserrat Gonzalez Arenas, Ashwin Balakrishna, Nathan Batchelor, Alex Bewley, Jeff Bingham, et al. Gemini robotics 1.5: Pushing the frontier of generalist robots with advanced embodied reasoning, thinking, and motion transfer. *arXiv preprint arXiv:2510.03342*, 2025.

[14] Alisson Azzolini, Junjie Bai, Hannah Brandon, Jiaxin Cao, Prithvijit Chattopadhyay, Huayu Chen, Jinju Chu, Yin Cui, Jenna Diamond, Yifan Ding, et al. Cosmos-reason1: From physical common sense to embodied reasoning. *arXiv preprint arXiv:2503.15558*, 2025.

[15] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[16] Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, Jiasen Lu, Taira Anderson, Erin Bransom,

Kiana Ehsani, Huong Ngo, YenSung Chen, Ajay Patel, Mark Yatskar, Chris Callison-Burch, Andrew Head, Rose Hendrix, Favyen Bastani, Eli VanderBilt, Nathan Lambert, Yvonne Chou, Arnavi Chheda, Jenna Sparks, Sam Skjonsberg, Michael Schmitz, Aaron Sarnat, Byron Bischoff, Pete Walsh, Chris Newell, Piper Wolters, Tanmay Gupta, Kuo-Hao Zeng, Jon Borchardt, Dirk Groeneveld, Jen Dumas, Crystal Nam, Sophie Lebrecht, Caitlin Wittlif, Carissa Schoenick, Oscar Michel, Ranjay Krishna, Luca Weihs, Noah A. Smith, Hannaneh Hajishirzi, Ross Girshick, Ali Farhadi, and Aniruddha Kembhavi. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 91–104, 2025.

[17] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report, 2025. URL https://arxiv.org/abs/2511.21631.

[18] Pierre Sermanet, Tianli Ding, Jeffrey Zhao, Fei Xia, Debidatta Dwibedi, Keerthana Gopalakrishnan, Christine Chan, Gabriel Dulac-Arnold, Sharath Maddineni, Nikhil J Joshi, et al. Robovqa: Multimodal long-horizon reasoning for robotics. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 645–652. IEEE, 2024.

[19] Haowen Liu, Shaoxiong Yao, Haonan Chen, Jiawei Gao, Jiayuan Mao, Jia-Bin Huang, and Yilun Du. Simpact: Simulation-enabled action planning using vision-language models. *arXiv preprint arXiv:2512.05955*, 2025.

[20] Zeting Liu, Zida Yang, Zeyu Zhang, and Hao Tang. Evovla: Self-evolving vision-language-action model. *arXiv preprint arXiv:2511.16166*, 2025.

[21] Qi Sun, Pengfei Hong, Tej Deep Pala, Vernon Toh, U-Xuan Tan, Deepanway Ghosal, and Soujanya Poria. Emma-x: An embodied multimodal action model with grounded chain of thought and look-ahead spatial reasoning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14199–14214, 2025.

[22] Yunze Man, De-An Huang, Guilin Liu, Shiwei Sheng, Shilong Liu, Liang-Yan Gui, Jan Kautz, Yu-Xiong Wang, and Zhiding Yu. Argus: Vision-centric reasoning with grounded chain-of-thought. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 14268–14280, 2025.

[23] Andy Zhai, Brae Liu, Bruno Fang, Chalse Cai, Ellie Ma, Ethan Yin, Hao Wang, Hugo Zhou, James Wang, Lights Shi, et al. Igniting vlms toward the embodied space. *arXiv preprint arXiv:2509.11766*, 2025.

[24] Theodore Sumers, Kenneth Marino, Arun Ahuja, Rob Fergus, and Ishita Dasgupta. Distilling internet-scale vision-language models into embodied agents. *arXiv preprint arXiv:2301.12507*, 2023.

[25] Zhongyi Zhou, Yichen Zhu, Minjie Zhu, Junjie Wen, Ning Liu, Zhiyuan Xu, Weibin Meng, Yaxin Peng, Chaomin Shen, Feifei Feng, et al. Chatvla: Unified multimodal understanding and robot control with vision-language-action model. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 5377–5395, 2025.

[26] Andreas Steiner, André Susano Pinto, Michael Tschannen, Daniel Keysers, Xiao Wang, Yonatan Bitton, Alexey Gritsenko, Matthias Minderer, Anthony Sherbondy, Shangbang Long, et al. Paligemma 2: A family of versatile vlms for transfer. *arXiv preprint arXiv:2412.03555*, 2024.

[27] Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, et al. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

[28] Siddharth Karamcheti, Suraj Nair, Ashwin Balakrishna, Percy Liang, Thomas Kollar, and Dorsa Sadigh. Prismatic vlms: Investigating the design space of visually-conditioned language models. In *Forty-first International Conference on Machine Learning*, 2024.

[29] Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903. IEEE, 2024.

[30] Homer Rich Walke, Kevin Black, Tony Z. Zhao, Quan Vuong, Chongyi Zheng, Philippe Hansen-Estruch, Andre Wang He, Vivek Myers, Moo Jin Kim, Max Du, Abraham Lee, Kuan Fang, Chelsea Finn, and Sergey Levine. Bridgedata v2: A dataset for robot learning at scale. In Jie Tan, Marc Toussaint, and Kourosh Darvish, editors, *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pages 1723–1736. PMLR, 06–09 Nov 2023.

[31] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

[32] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020.

[33] Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, and Sergey Levine. FAST: Efficient Action Tokenization for Vision-Language-Action Models. In *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025. doi: 10.15607/RSS.2025.XXI.012.

[34] Yating Wang, Haoyi Zhu, Mingyu Liu, Jiange Yang, Hao-Shu Fang, and Tong He. Vq-vla: Improving vision-language-action models via scaling vector-quantized action tokenizers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

[35] Roya Firoozi, Johnathan Tucker, Stephen Tian, Anirudha Majumdar, Jiankai Sun, Weiyu Liu, Yuke Zhu, Shuran Song, Ashish Kapoor, Karol Hausman, et al. Foundation models in robotics: Applications, challenges, and the future. *The International Journal of Robotics Research*, 44(5):701–739, 2025.

[36] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Tomas Jackson, Sally Jesmonth, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Kuang-Huei Lee, Sergey Levine, Yao Lu, Utsav Malla, Deeksha Manjunath, Igor Mordatch, Ofir Nachum, Carolina Parada, Jodilyn Peralta, Emily Perez, Karl Pertsch, Jornell Quiambao, Kanishka Rao, Michael Ryoo, Grecia Salazar, Pannag Sanketi, Kevin Sayed, Jaspiar Singh, Sumedh Sontakke, Austin Stone, Clayton Tan, Huong Tran, Vincent Vanhoucke, Steve Vega, Quan Vuong, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. Rt-1: Robotics transformer for real-world control at scale. In *arXiv preprint arXiv:2212.06817*, 2022.

[37] Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pages 2165–2183. PMLR, 2023.

[38] Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Charles Xu, Jianlan Luo, Tobias Kreiman, You Liang Tan, Lawrence Yunliang Chen, Pannag Sanketi, Quan Vuong, Ted Xiao, Dorsa Sadigh, Chelsea Finn, and Sergey Levine. Octo: An open-source generalist robot policy. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, 2024.

[39] Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

[40] An-Chieh Cheng, Yandong Ji, Zhaojing Yang, Zaitian Gongye, Xueyan Zou, Jan Kautz, Erdem Biyik, Hongxu Yin, Sifei Liu, and Xiaolong Wang. NaVILA: Legged Robot Vision-Language-Action Model for Navigation. In *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025. doi: 10.15607/RSS.2025.XXI.018.

[41] Pengxiang Ding, Han Zhao, Wenjie Zhang, Wenxuan Song, Min Zhang, Siteng Huang, Ningxi Yang, and Donglin Wang. Quar-vla: Vision-language-action model for quadruped robots. In *European Conference on Computer Vision*, pages 352–367. Springer, 2024.

[42] Pengxiang Ding, Jianfei Ma, Xinyang Tong, Binghong Zou, Xinxin Luo, Yiguo Fan, Ting Wang, Hongchao Lu, Panzhong Mo, Jinxin Liu, et al. Humanoid-vla: Towards universal humanoid control with visual integration. *arXiv preprint arXiv:2502.14795*, 2025.

[43] Haoran Jiang, Jin Chen, Qingwen Bu, Li Chen, Modi Shi, Yanjie Zhang, Delong Li, Chuanzhe Suo, Chuang Wang, Zhihui Peng, et al. Wholebodyvla: Towards unified latent vla for whole-body loco-manipulation control. *arXiv preprint arXiv:2512.11047*, 2025.

[44] Xingcheng Zhou, Xuyuan Han, Feng Yang, Yunpu Ma, Volker Tresp, and Alois Knoll. Opendrivevla: Towards end-to-end autonomous driving with large vision language action model, 2025. URL https://arxiv.org/abs/2503.23463.

[45] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[46] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022.

[47] William Merrill and Ashish Sabharwal. The expressive power of transformers with chain of thought. *arXiv preprint arXiv:2310.07923*, 2023.

[48] Zhiyuan Li, Hong Liu, Denny Zhou, and Tengyu Ma. Chain of thought empowers transformers to solve inherently serial problems. *arXiv preprint arXiv:2402.12875*, 1, 2024.

[49] Guhao Feng, Bohang Zhang, Yuntian Gu, Haotian Ye, Di He, and Liwei Wang. Towards revealing the mystery

behind chain of thought: a theoretical perspective. *Advances in Neural Information Processing Systems*, 36: 70757–70798, 2023.

[50] Xuezhi Wang and Denny Zhou. Chain-of-thought reasoning without prompting. *Advances in Neural Information Processing Systems*, 37:66383–66409, 2024.

[51] Peng-Yuan Wang, Tian-Shuo Liu, Chenyang Wang, Ziniu Li, Yidi Wang, Shu Yan, Chengxing Jia, Xu-Hui Liu, Xinwei Chen, Jiacheng Xu, et al. A survey on large language models for mathematical reasoning. *ACM Computing Surveys*, 2025.

[52] Dayu Yang, Tianyang Liu, Daoan Zhang, Antoine Simoulin, Xiaoyi Liu, Yuwei Cao, Zhaopu Teng, Xin Qian, Grey Yang, Jiebo Luo, et al. Code to think, think to code: A survey on code-enhanced reasoning and reasoning-driven code intelligence in llms. *arXiv preprint arXiv:2502.19411*, 2025.

[53] Guowei Xu, Peng Jin, Ziang Wu, Hao Li, Yibing Song, Lichao Sun, and Li Yuan. Llava-cot: Let vision language models reason step-by-step. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2087–2098, 2025.

[54] Yiyang Zhou, Haoqin Tu, Zijun Wang, Zeyu Wang, Niklas Muennighoff, Fan Nie, Yejin Choi, James Zou, Chaorui Deng, Shen Yan, et al. When visualizing is the first step to reasoning: Mira, a benchmark for visual chain-of-thought. *arXiv preprint arXiv:2511.02779*, 2025.

[55] Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, et al. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*, 2024.

[56] Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. Textbooks are all you need ii: phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*, 2023.

[57] Ziyang Luo, Can Xu, Pu Zhao, Qingfeng Sun, Xiubo Geng, Wenxiang Hu, Chongyang Tao, Jing Ma, Qingwei Lin, and Daxin Jiang. Wizardcoder: Empowering code large language models with evol-instruct. *arXiv preprint arXiv:2306.08568*, 2023.

[58] Yuxiang Wei, Zhe Wang, Jiawei Liu, Yifeng Ding, and Lingming Zhang. Magicoder: Empowering code generation with oss-instruct. *arXiv preprint arXiv:2312.02120*, 2023.

[59] Bingbin Liu, Sebastien Bubeck, Ronen Eldan, Janardhan Kulkarni, Yuanzhi Li, Anh Nguyen, Rachel Ward, and Yi Zhang. Tinygsm: achieving >80% on gsm8k with small language models. *arXiv preprint arXiv:2312.09241*, 2023.

[60] Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. Wizardlm: Empowering large language models to follow complex instructions. *arXiv preprint arXiv:2304.12244*, 2023.

[61] Tao Ge, Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. Scaling synthetic data creation with 1,000,000,000 personas. *arXiv preprint arXiv:2406.20094*, 2024.

[62] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Stanford alpaca: An instruction-following llama model, 2023.

[63] Trieu H Trinh, Yuhuai Wu, Quoc V Le, He He, and Thang Luong. Solving olympiad geometry without human demonstrations. *Nature*, 625(7995):476–482, 2024.

[64] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[65] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529 (7587):484–489, 2016.

[66] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815*, 2017.

[67] Haolin Chen, Yihao Feng, Zuxin Liu, Weiran Yao, Akshara Prabhakar, Shelby Heinecke, Ricky Ho, Phil Mui, Silvio Savarese, Caiming Xiong, et al. Language models are hidden reasoners: Unlocking latent reasoning capabilities via self-rewarding. *arXiv preprint arXiv:2411.04282*, 2024.

[68] Matthew Douglas Hoffman, Du Phan, David Dohan, Sholto Douglas, Tuan Anh Le, Aaron Parisi, Pavel Sountsov, Charles Sutton, Sharad Vikram, and Rif A Saurous. Training chain-of-thought via latent-variable inference. In *NeurIPS*, 2023.

[69] Edward J Hu, Moksh Jain, Eric Elmoznino, Younesse Kaddar, Guillaume Lajoie, Yoshua Bengio, and Nikolay Malkin. Amortizing intractable inference in large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[70] Han Zhong, Yutong Yin, Shenao Zhang, Xiaojun Xu, Yuanxin Liu, Yifei Zuo, Zhihan Liu, Boyi Liu, Sirui Zheng, Hongyi Guo, et al. Brite: Bootstrapping reinforced thinking process to enhance language model reasoning. *arXiv preprint arXiv:2501.18858*, 2025.

[71] Avi Singh, John D Co-Reyes, Rishabh Agarwal, Ankesh Anand, Piyush Patil, Xavier Garcia, Peter J Liu, James Harrison, Jaehoon Lee, Kelvin Xu, Aaron T Parisi, Abhishek Kumar, Alexander A Alemi, Alex Rizkowsky, Azade Nova, Ben Adlam, Bernd Bohnet, Gamaleldin Fathy Elsayed, Hanie Sedghi, Igor Mor-

datch, Isabelle Simpson, Izzeddin Gur, Jasper Snoek, Jeffrey Pennington, Jiri Hron, Kathleen Kenealy, Kevin Swersky, Kshiteej Mahajan, Laura A Culp, Lechao Xiao, Maxwell Bileschi, Noah Constant, Roman Novak, Rosanne Liu, Tris Warkentin, Yamini Bansal, Ethan Dyer, Behnam Neyshabur, Jascha Sohl-Dickstein, and Noah Fiedel. Beyond human data: Scaling self-training for problem-solving with language models. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=lNAyUngGFK. Expert Certification.

[72] Yangjun Ruan, Neil Band, Chris J Maddison, and Tatsunori Hashimoto. Reasoning to learn from latent thoughts. *arXiv preprint arXiv:2503.18866*, 2025.

[73] Pratyusha Sharma, Antonio Torralba, and Jacob Andreas. Skill induction and planning with latent language. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1713–1726, 2022.

[74] Yilin Wu, Anqi Li, Tucker Hermans, Fabio Ramos, Andrea Bajcsy, and Claudia P'erez-D'Arpino. Do what you say: Steering vision-language-action models via runtime reasoning-action alignment verification. *arXiv preprint arXiv:2510.16281*, 2025.

[75] Gemini Team, R Anil, S Borgeaud, JB Alayrac, J Yu, R Soricut, J Schalkwyk, AM Dai, A Hauth, K Millican, et al. Gemini: A family of highly capable multimodal models. arxiv 2023. *arXiv preprint arXiv:2312.11805*, 2024.

[76] Jason Lee, Jiafei Duan, Haoquan Fang, Yuquan Deng, Shuo Liu, Boyang Li, Bohan Fang, Jieyu Zhang, Yi Ru Wang, Sangho Lee, et al. Molmoact: Action reasoning models that can reason in space. *arXiv preprint arXiv:2508.07917*, 2025.

[77] Minjie Zhu, Yichen Zhu, Jinming Li, Zhongyi Zhou, Junjie Wen, Xiaoyu Liu, Chaomin Shen, Yaxin Peng, and Feifei Feng. Objectvla: End-to-end open-world object manipulation without demonstration. *arXiv preprint arXiv:2502.19250*, 2025.

[78] Qingqing Zhao, Yao Lu, Moo Jin Kim, Zipeng Fu, Zhuoyang Zhang, Yecheng Wu, Zhaoshuo Li, Qianli Ma, Song Han, Chelsea Finn, et al. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 1702–1713, 2025.

[79] Boyuan Chen, Zhuo Xu, Sean Kirmani, Brain Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14455–14465, 2024.

[80] Ruijie Zheng, Yongyuan Liang, Shuaiyi Huang, Jianfeng Gao, Hal Daumé III, Andrey Kolobov, Furong Huang, and Jianwei Yang. Tracevla: Visual trace prompting enhances spatial-temporal awareness for generalist robotic policies. *arXiv preprint arXiv:2412.10345*, 2024.

[81] Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. Palm-e: an embodied multimodal language model. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

[82] Oier Mees, Jessica Borja-Diaz, and Wolfram Burgard. Grounding language with visual affordances over unstructured data. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11576–11582, 2023. doi: 10.1109/ICRA48891.2023.10160396.

[83] Rongtao Xu, Jian Zhang, Minghao Guo, Youpeng Wen, Haoting Yang, Min Lin, Jianzheng Huang, Zhe Li, Kaidong Zhang, Liqiong Wang, et al. A0: An affordance-aware hierarchical model for general robotic manipulation. *arXiv preprint arXiv:2504.12636*, 2025.

[84] Jessica Borja-Diaz, Oier Mees, Gabriel Kalweit, Lukas Hermann, Joschka Boedecker, and Wolfram Burgard. Affordance learning from play for sample-efficient policy learning. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 6372–6378. IEEE, 2022.

[85] Jiageng Mao, Yuxi Qian, Junjie Ye, Hang Zhao, and Yue Wang. Gpt-driver: Learning to drive with gpt. *arXiv preprint arXiv:2310.01415*, 2023.

[86] Jiageng Mao, Junjie Ye, Yuxi Qian, Marco Pavone, and Yue Wang. A language agent for autonomous driving. In *First Conference on Language Modeling*, 2024.

[87] NVIDIA, :, Yan Wang, Wenjie Luo, Junjie Bai, Yulong Cao, Tong Che, Ke Chen, Yuxiao Chen, Jenna Diamond, Yifan Ding, Wenhao Ding, Liang Feng, Greg Heinrich, Jack Huang, Peter Karkus, Boyi Li, Pinyi Li, Tsung-Yi Lin, Dongran Liu, Ming-Yu Liu, Langechuan Liu, Zhijian Liu, Jason Lu, Yunxiang Mao, Pavlo Molchanov, Lindsey Pavao, Zhenghao Peng, Mike Ranzinger, Ed Schmerling, Shida Shen, Yunfei Shi, Sarah Tariq, Ran Tian, Tilman Wekel, Xinshuo Weng, Tianjun Xiao, Eric Yang, Xiaodong Yang, Yurong You, Xiaohui Zeng, Wenyuan Zhang, Boris Ivanovic, and Marco Pavone. Alpamayo-r1: Bridging reasoning and action prediction for generalizable autonomous driving in the long tail, 2025. URL https://arxiv.org/abs/2511.00088.

[88] Milan Ganai, Rohan Sinha, Christopher Agia, Daniel Morton, Luigi Di Lillo, and Marco Pavone. Real-time out-of-distribution failure prevention via multi-modal reasoning. In Joseph Lim, Shuran Song, and Hae-Won Park, editors, *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pages 283–308. PMLR, 27–30 Sep 2025.

[89] Zhenghao Peng, Wenhao Ding, Yurong You, Yuxiao Chen, Wenjie Luo, Thomas Tian, Yulong Cao, Apoorva Sharma, Danfei Xu, Boris Ivanovic, Boyi Li, Bolei Zhou, Yan Wang, and Marco Pavone. Counterfactual vla: Self-reflective vision-language-action model with adaptive reasoning. *arXiv preprint arXiv:2512.24426*, 2025.

[90] Catherine Glossop, William Chen, Arjun Bhorkar, Dhruv Shah, and Sergey Levine. Cast: Counterfactual labels improve instruction following in vision-language-action models. *arXiv preprint arXiv:2508.13446*, 2025.

[91] Tuan Van Vo, Tan Quang Nguyen, Khang Minh Nguyen, Duy Ho Minh Nguyen, and Minh Nhat Vu. Refinevla: Reasoning-aware teacher-guided transfer fine-tuning. *arXiv preprint arXiv:2505.19080*, 2025.

[92] Jaden Clark, Suvir Mirchandani, Dorsa Sadigh, and Suneel Belkhale. Action-free reasoning for policy generalization. *arXiv preprint arXiv:2502.03729*, 2025.

[93] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

[94] Yuri Burda, Roger B. Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016. URL http://arxiv.org/abs/1509.00519.

[95] Chris Cremer, Quaid Morris, and David Duvenaud. Reinterpreting importance-weighted autoencoders. *arXiv preprint arXiv:1704.02916*, 2017.

[96] Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36: 44776–44791, 2023.

[97] Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

[98] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=a68SUt6zFt. Featured Certification.

[99] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023.

[100] Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023. doi: 10.15607/RSS.2023.XIX.016.

[101] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[102] Jacky Kwok, Christopher Agia, Rohan Sinha, Matt Foutter, Shulu Li, Ion Stoica, Azalia Mirhoseini, and Marco Pavone. Robomonkey: Scaling test-time sampling and verification for vision-language-action models. In Joseph Lim, Shuran Song, and Hae-Won Park, editors, *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pages 3200–3217. PMLR, 27–30 Sep 2025.

[103] Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Livia Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E Gonzalez, Clark Barrett, and Ying Sheng. Sglang: Efficient execution of structured language model programs. *Advances in neural information processing systems*, 37: 62557–62583, 2024.

[104] Tim Windecker, Manthan Patel, Moritz Reuss, Richard Schwarzkopf, Cesar Cadena, Rudolf Lioutikov, Marco Hutter, and Jonas Frey. Navitrace: Evaluating embodied navigation of vision-language models. *arXiv preprint arXiv:2510.26909*, 2025.

[105] Zeyi Liu, Arpit Bahety, and Shuran Song. Reflect: Summarizing robot experiences for failure explanation and correction. *arXiv preprint arXiv:2306.15724*, 2023.

[106] Chi-Pin Huang, Yueh-Hua Wu, Min-Hung Chen, Yu-Chiang Frank Wang, and Fu-En Yang. Thinkact: Vision-language-action reasoning via reinforced visual latent planning. *arXiv preprint arXiv:2507.16815*, 2025.

[107] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, Lewei Lu, Xiaosong Jia, Qiang Liu, Jifeng Dai, Yu Qiao, and Hongyang Li. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17853–17862, 2023.

[108] Jyh-Jing Hwang, Runsheng Xu, Hubert Lin, Wei-Chih Hung, Jingwei Ji, Kristy Choi, Di Huang, Tong He, Paul Covington, Benjamin Sapp, Yin Zhou, James Guo, Dragomir Anguelov, and Mingxing Tan. Emma: End-

to-end multimodal model for autonomous driving. *arXiv preprint arXiv:2410.23262*, 2024.

[109] Craig Kapfer, Kurt Stine, Balasubramanian Narasimhan, Christopher Mentzel, and Emmanuel Candes. Marlowe: Stanford's gpu-based computational instrument, 2025.

[110] A Gemini Team. Family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

CONTENTS

I Introduction 1

II Related Works 2

III Preliminaries on Variational Inference 3
- III-A Latent Variable Models and the Variational Autoencoders . . . 3
- III-B Importance Weighted Autoencoders (IWAE) . . . 3

IV R&B-EnCoRe Framework 4
- IV-A Warmstarting Strategy Hypotheses via Reasoning Dropout . . . 4
- IV-B Jointly Training Prior and Posterior . . . 4
- IV-C Refining and Bootstrapping via Importance Sampling . . . 4

V Experiments 5
- V-A LIBERO-90 Franka Panda Manipulation . . . 5
- V-B Bridge WidowX Hardware . . . 6
- V-C Legged Robots Navigation . . . 8
- V-D Autonomous Vehicles . . . 9

VI Discussion and Conclusion 9

Appendix A: Notation Table 18

Appendix B: Proposition 19
- B-A Empirical Support for Assumption . . . 20

Appendix C: Importance-Weighted Variational Inference with Categorical Resampling 20
- C-A From IWAE to SIR IWAE . . . 20
- C-B Practical Advantages of SIR IWAE . . . 21
- C-C Adapting SIR IWAE in R&B-EnCoRe . . . 21
  - C-C1 The Surrogate Posterior as Proposal Distribution . . . 21
  - C-C2 Importance Weight Computation . . . 21
  - C-C3 Categorical Resampling for Data Refinement . . . 21
- C-D Contextualizing Information Benefit in SIR IWAE . . . 22
  - C-D1 How SIR Reweights Reasoning Strategies . . . 22
  - C-D2 The Large-K Limit: Relating SIR to Expected Importance Weights . . . 22
  - C-D3 Quantifying Strategy Preference Through Probability Ratios . . . 23
  - C-D4 Information Benefit Drives Strategy Selection . . . 23
  - C-D5 Relating to R&B-EnCoRe . . . 24
- C-E Scaling with Inference Compute . . . 24

Appendix D: Experimental Details 25
- D-A LIBERO-90 Manipulation . . . 25
  - D-A1 Model Architecture and Training . . . 25
  - D-A2 Reasoning Primitives . . . 25
  - D-A3 Inferencing . . . 25
- D-B Bridgev2 WidowX Hardware Manipulation . . . 25
  - D-B1 Model Architecture and Training . . . 25
  - D-B2 Reasoning Primitives . . . 26
  - D-B3 Evaluation . . . 26
- D-C Legged Robot Navigation . . . 26
  - D-C1 Model Architecture and Training . . . 26
  - D-C2 Self-Generated Reasoning Primitives . . . 26
  - D-C3 Inference Protocol . . . 29

D-D Autonomous Driving 30
D-D1 Model Architecture and Representation 30
D-D2 Reasoning Primitives 30
D-D3 Data Transformation and Evaluation 30

Appendix E: Hardware Details with Results 31

Appendix F: Additional Ablation Experiments 32

Appendix G: Example Reasoning Traces Across Embodiments 34

APPENDIX A
NOTATION TABLE

TABLE V: Notation and Symbols used in paper

Symbol	Description
Core Variables	
C	Context (observation image and task description)
A	Action (discrete action tokens, 2D waypoints, etc)
Z	Latent reasoning trace (unobserved strategy)
R	Reasoning primitve (e.g., Affordance reasoning)
ρ	Number of reasoning primitive types
$\mathcal{R}$	Set of reasoning primitives $\{R_1, \ldots, R_\rho\}$
$\mathbf{R}$	Reasoning Strategy, subset of reasoning primitives
z_R	Textual content for reasoning primitive $R \in \mathcal{R}$
Probabilistic Models	
$p(A \mid C)$	Marginal action distribution given context
$p(Z, A \mid C)$	Prior distribution (joint over reasoning and action)
$p(A \mid C, Z)$	Action likelihood given context and reasoning
$p(Z \mid C)$	Prior reasoning distribution
$q(Z \mid C, A)$	Posterior distribution (reasoning given action)
$p_{\text{data}}(A \mid C)$	Expert action distribution
Variational Inference	
K	Number of posterior samples in IWAE
$Z_1, \ldots, Z_K$	K i.i.d. samples from posterior $q(Z \mid C, A)$
$w_k, w(Z_k)$	Importance weight $\frac{p(Z_k, A \mid C)}{q(Z_k \mid C, A)}$
$\mathcal{L}_K$	K-sample importance-weighted lower bound
ELBO_{VAE}	Evidence lower bound for standard VAE
$D_{\text{KL}}(\cdot \Vert \cdot)$	Kullback-Leibler divergence
Datasets and Training	
$\mathcal{D}$	Original dataset: $\{(C^i, A^i)\}_{i=1}^N$
N	Number of demonstrations in dataset
$\mathcal{D}_{\text{warm}}$	Warmstart dataset with diverse reasoning traces
$\mathcal{D}_{\text{refined}}$	Refined dataset with action-predictive reasoning
M	Number of warmstarting traces per demonstration
d	Reasoning dropout rate
Z_j^i	j-th synthetic reasoning trace for demonstration i
Z^{i*}	Refined, importance-sampled reasoning trace
Models	
$\mathcal{M}$	Base vision-language model
$\mathcal{M}_{pq}$	Jointly trained prior-posterior model
$\mathcal{M}_{\text{VLA}}$	Final vision-language-action model
FM	Foundation model for generated reasoning content
Information Benefit	
ΔI_R	Information benefit of reasoning strategy R
$\mathcal{Z}_R$	Set of reasoning traces containing strategy R
$\mathcal{Z}_{\not R}$	Set of reasoning traces not containing strategy R
$\mathbb{E}_{Z \sim q}[w(Z) \mid Z \in \mathcal{Z}_{\mathbf{R}}]$	Expected importance weight of reasoning traces in set $\mathcal{Z}_R$

APPENDIX B
PROPOSITION

Proposition (Importance Weights Capture Information Gain)**.** *Let $A \sim p_{data}(A \mid C)$ be the observed ground-truth action and Z be a latent reasoning variable. We partition the reasoning space into disjoint sets $\mathcal{Z}_{\mathbf{R}}$ and $\mathcal{Z}_{\not\mathbf{R}}$.*

We assume that we construct our latent reasoning dataset $\mathcal{Z} = \mathcal{Z}_{\mathbf{R}} \cup \mathcal{Z}_{\not\mathbf{R}}$ so the latents $Z_{\not\mathbf{R}}$ without reasoning primitive or more generally strategy $\mathbf{R}$ are sampled with constant "dropout probability" d and each latent $Z_{\mathbf{R}}$ with reasoning $\mathbf{R}$ with constant probability $1 - d$, and then we train both the prior latent generation $p(Z \mid C)$ and posterior $q(Z \mid C, A)$ on the constructed dataset. Crucially, because Alg. 1 applies reasoning dropout independently of the action A, the conditional probability of the reasoning partition in the ground-truth data is constant. Therefore, assuming the models p and q converge to the underlying data distribution during training in Alg. 2, we have the point-wise equality for all A:

$$p(\mathcal{Z}_{\mathbf{R}} \mid C) = q(\mathcal{Z}_{\mathbf{R}} \mid C, A) = (1 - d),$$
$$p(\mathcal{Z}_{\not\mathbf{R}} \mid C) = q(\mathcal{Z}_{\not\mathbf{R}} \mid C, A) = d.$$

We define the information benefit of of a reasoning strategy $\mathbf{R}$ as how much it reduces the divergence between our model's action distribution and the expert's distribution $p_{data}(A \mid C)$:

$$\Delta\mathcal{I}_{\mathbf{R}} \doteq D_{KL}(p_{data} \| p(A \mid C, \mathcal{Z}_{\not\mathbf{R}})) - D_{KL}(p_{data} \| p(A \mid C, \mathcal{Z}_{\mathbf{R}}))$$

Then, the expected log-ratio of the importance weights (defined as $w(Z) \doteq \frac{p(A,Z|C)}{q(Z|C,A)}$) assigned to reasoning traces $\mathcal{Z}_{\mathbf{R}}$ with the strategy versus reasoning traces $\mathcal{Z}_{\not\mathbf{R}}$ without the strategy is:

$$\mathop{\mathbb{E}}_{A \sim p_{data}} \log \frac{\mathbb{E}_{Z \sim q}[w(Z) \mid Z \in \mathcal{Z}_{\mathbf{R}}]}{\mathbb{E}_{Z \sim q}[w(Z) \mid Z \in \mathcal{Z}_{\not\mathbf{R}}]} = \Delta\mathcal{I}_{\mathbf{R}}.$$

Proof: The expected importance weight for reasoning traces from the partition $\mathcal{Z}_{\mathbf{R}}$ is:

$$\begin{aligned} W_{\mathbf{R}} &\doteq \mathop{\mathbb{E}}_{Z \sim q}[w(Z) \mid Z \in \mathcal{Z}_{\mathbf{R}}] \\ &= \int_{\mathcal{Z}_{\mathbf{R}}} \frac{p(A \mid C, Z)p(Z \mid C)}{q(Z \mid C, A)} q(Z \mid C, A, \mathcal{Z}_{\mathbf{R}}) dZ \end{aligned}$$

Using identity $q(Z \mid C, A) = q(Z \mid C, A, \mathcal{Z}_{\mathbf{R}})q(\mathcal{Z}_{\mathbf{R}} \mid C, A)$ for $Z \in \mathcal{Z}_{\mathbf{R}}$:

$$\begin{aligned} W_{\mathbf{R}} &= \int_{\mathcal{Z}_{\mathbf{R}}} \frac{p(A \mid C, Z)p(Z \mid C)}{q(Z \mid C, A, \mathcal{Z}_{\mathbf{R}})q(\mathcal{Z}_{\mathbf{R}} \mid C, A)} q(Z \mid C, A, \mathcal{Z}_{\mathbf{R}}) dZ \\ &= \frac{1}{q(\mathcal{Z}_{\mathbf{R}} \mid C, A)} \int_{\mathcal{Z}_{\mathbf{R}}} p(A \mid C, Z)p(Z \mid C) dZ \end{aligned}$$

This integral is the joint probability of the data and the partition, $p(A, \mathcal{Z}_{\mathbf{R}} \mid C)$.

$$W_{\mathbf{R}} = \frac{p(A, \mathcal{Z}_{\mathbf{R}} \mid C)}{q(\mathcal{Z}_{\mathbf{R}} \mid C, A)} = \frac{p(A \mid C, \mathcal{Z}_{\mathbf{R}})p(\mathcal{Z}_{\mathbf{R}} \mid C)}{q(\mathcal{Z}_{\mathbf{R}} \mid C, A)}$$

Similarly for the $\mathcal{Z}_{\not\mathbf{R}}$ partition:

$$W_{\not\mathbf{R}} = \frac{p(A \mid C, \mathcal{Z}_{\not\mathbf{R}})p(\mathcal{Z}_{\not\mathbf{R}} \mid C)}{q(\mathcal{Z}_{\not\mathbf{R}} \mid C, A)}$$

Now we look at the ratio $W_{\mathbf{R}}/W_{\not\mathbf{R}}$. This allows us to separate the Likelihood term (predictive power) from the Prior/Posterior reasoning distribution.

$$\frac{W_{\mathbf{R}}}{W_{\not\mathbf{R}}} = \underbrace{\frac{p(A \mid C, \mathcal{Z}_{\mathbf{R}})}{p(A \mid C, \mathcal{Z}_{\not\mathbf{R}})}}_{\text{Likelihood Ratio}} \times \underbrace{\left(\frac{p(\mathcal{Z}_{\mathbf{R}} \mid C)/q(\mathcal{Z}_{\mathbf{R}} \mid C, A)}{p(\mathcal{Z}_{\not\mathbf{R}} \mid C)/q(\mathcal{Z}_{\not\mathbf{R}} \mid C, A)}\right)}_{= 1 \text{ (from assumption)}}$$

Taking the log of the ratio and expectation over $A \sim p_{\text{data}}$, and recalling the definition of cross-entropy and KL $\mathbb{E}_p[\log q] = -H(p) - D_{KL}(p \| q)$, where $H(\cdot)$ measures entropy of a distribution:

$$
\begin{aligned}
\mathop{\mathbb{E}}_{A\sim p_{\text{data}}} \log \frac{\mathbb{E}_{Z\sim q}[w(Z) \mid Z \in \mathcal{Z}_{\mathbf{R}}]}{\mathbb{E}_{Z\sim q}[w(Z) \mid Z \in \mathcal{Z}_{\not\mathbf{R}}]} &= \mathop{\mathbb{E}}_{A\sim p_{\text{data}}} \left[\log \frac{W_{\mathbf{R}}}{W_{\not\mathbf{R}}}\right] \\
&= \mathop{\mathbb{E}}_{A\sim p_{\text{data}}} \left[\log \frac{p(A \mid C, \mathcal{Z}_{\mathbf{R}})}{p(A \mid C, \mathcal{Z}_{\not\mathbf{R}})}\right] \\
&= \mathop{\mathbb{E}}_{A\sim p_{\text{data}}} [\log p(A \mid C, \mathcal{Z}_{\mathbf{R}})] - \mathop{\mathbb{E}}_{A\sim p_{\text{data}}} [\log p(A \mid C, \mathcal{Z}_{\not\mathbf{R}})] \\
&= (-H(p_{\text{data}}) - D_{KL}(p_{\text{data}} \| p(A \mid C, \mathcal{Z}_{\mathbf{R}}))) \\
&\quad - (-H(p_{\text{data}}) - D_{KL}(p_{\text{data}} \| p(A \mid C, \mathcal{Z}_{\not\mathbf{R}}))) \\
&= D_{KL}(p_{\text{data}} \| p(A \mid C, \mathcal{Z}_{\not\mathbf{R}})) - D_{KL}(p_{\text{data}} \| p(A \mid C, \mathcal{Z}_{\mathbf{R}})) \\
&= \Delta\mathcal{I}_{\mathbf{R}}
\end{aligned}
$$

■

A. Empirical Support for Assumption

In the proposition, one assumption we make is that the probability of a reasoning traces with strategy $\mathbf{R}$ from the trained prior $p(\mathcal{Z}_{\mathbf{R}} \mid C)$ and posterior $q(\mathcal{Z}_{\mathbf{R}} \mid C, A)$ is $(1-d)$, which is 1 minus the dropout rate. We make this assumption because of how we train in Alg. 1 the prior and posterior models: specifically they both are trained with the same underlying distribution of reasoning data where the primitives are dropped out with probability d. We empirically validate this assumption by investigating the reasoning primitive frequencies from the raw prior and posterior distributions in Fig. 13 for the LIBERO-90 reasoning setting where the dropout rate $d = 0.2$. Overall we observe that the reasoning primitive distributions are quite close to $1 - d = 0.8$.

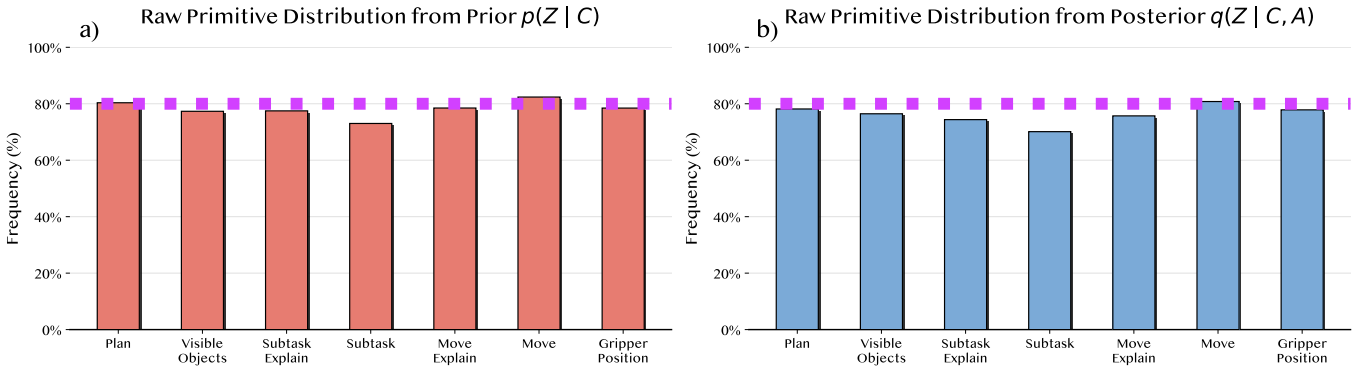


Fig. 13: Reasoning primitive distributions from the **raw poseterior and prior** distributions. Note this is before the reweighting and importance sampling step (minor gaps due to warmstarting sampling noise and potential base model prior bias).

APPENDIX C
IMPORTANCE-WEIGHTED VARIATIONAL INFERENCE WITH CATEGORICAL RESAMPLING

In the main text, we introduced the Importance Weighted Autoencoder (IWAE) framework [94], which provides a tighter lower bound on the log-evidence through multiple importance-weighted samples. Our algorithm `R&B-EnCoRe` employs an improved variant of this approach that uses categorical resampling from importance weights, following the theoretical developments in Cremer et al. [95]. This section provides the theoretical justification for this approach and relates it to our method based on sampling-importance-resampling (SIR).

A. From IWAE to SIR IWAE

Recall from Section III that the K-sample IWAE bound is defined as:

$$
\mathcal{L}_K \doteq \mathbb{E}_{Z_1,\ldots,Z_K\sim q(Z|A)} \left[\log\left(\frac{1}{K}\sum_{k=1}^{K} w_k\right)\right], \tag{1}
$$

where $w_k = \frac{p(Z_k,A)}{q(Z_k|A)}$ are the importance weights.

A closer look at this bound shows we can apply sampling-importance-resampling (SIR) [93]. Consider the following procedure:

1) Sample K candidates $\{Z_k\}_{k=1}^K$ from the proposal distribution $q(Z|A)$
2) Compute importance weights w_k for each candidate
3) Resample a single latent Z^* from a categorical distribution proportional to the weights: $Z^* \sim \text{Cat}(\{w_k\}_{k=1}^K)$

This resampling procedure induces an improved proposal distribution $\tilde{q}(Z|A)$ that is defined implicitly through the SIR process. Critically, Cremer et al. [95] showed that this categorical resampling can be understood as sampling from a distribution that achieves a tighter IWAE bound:

Proposition (Categorical Resampling Bound Adapted from [95])**.** *Let $\tilde{q}(Z \mid A)$ denote the distribution induced by the categorical resampling procedure described above. Then, as $K \to \infty$, we have that $\tilde{q}(Z \mid A) \to p(Z \mid A)$, the true posterior. Furthermore, for the importance weighted lower bound:*

$$\tilde{\mathcal{L}}_K \doteq \mathbb{E}_{Z\sim\tilde{q}(Z|A)}\left[\log \frac{p(Z,A)}{\tilde{q}(Z \mid A)}\right],$$

and for $K \geq 1$, the bounds satisfy $\log p(A) \geq \tilde{\mathcal{L}}_K \geq \mathcal{L}_K \geq \mathcal{L}_1 = \textit{ELBO}_{\textit{VAE}}$.

This interpretation reveals that the IWAE bound can be viewed as training with an improved proposal distribution $\tilde{q}(Z|A)$ that is obtained through the resampling procedure. Importantly, this improved proposal $\tilde{q}(Z \mid A)$ is provably closer to the true posterior $p(Z \mid A)$ than the original $q(Z \mid A)$.

B. Practical Advantages of SIR IWAE

The categorical resampling interpretation provides several practical advantages for our embodied reasoning application:
Policy Improvement Guarantee The resampling procedure acts as a non-parametric policy improvement operator. Each resampled trace Z^* is theoretically guaranteed in expectation to come from a distribution that provides a theoretically tighter bound on $\log p(A)$ than the original proposal $q(Z|A)$.
Interpretable Refined Training Data Rather than computing expectations over the continuous distribution defined by the log-sum-exp of importance weights, we obtain samples from the improved posterior estimate $\tilde{q}(Z|A)$. This is important for our setting where we generate concrete reasoning traces Z^* to refining the original reasoning distribution and augment the training data. These reasoning traces can provide better interpretability and insight into what action-predictive embodied reasoning is.
Computational Efficiency For synthetic data generation at scale, generating K samples and resampling once is more efficient than computing gradients through the log-sum-exp operation required for standard IWAE training.
Data Augmentation Compatibility The discrete resampling naturally produces a refined dataset where each demonstration (C, A) is paired with a single high-quality reasoning trace Z^*, making it directly compatible with standard reasoning VLA model training pipelines.

C. Adapting SIR IWAE in `R&B-EnCoRe`

In our embodied reasoning setting, we apply this categorical resampling framework as follows:

1) The Surrogate Posterior as Proposal Distribution: During the Warmstarting training in Alg. 1, we have trained a model $\mathcal{M}_{pq}$ that parameterizes both:

- The joint distribution: $p(Z, A|C)$
- The approximate posterior: $q(Z|C, A)$

For each demonstration (C^i, A^i) in our dataset, we use $q(\cdot|C^i, A^i)$ as the proposal distribution to generate K candidate reasoning traces $\{Z_k^i\}_{k=1}^K$ (Alg. 1 Line 3).

2) Importance Weight Computation: For each candidate Z_k^i, we compute the importance weight (Alg. 1 Line 4-6):

$$w(Z_k^i) = \frac{p(Z_k^i, A^i|C^i)}{q(Z_k^i|C^i, A^i)}. \tag{2}$$

These weights can be computed efficiently using the model's log-probabilities for the respective conditioning patterns.

3) Categorical Resampling for Data Refinement: We then resample a single trace Z_i^* from the categorical distribution (Alg. 1 Line 7):

$$Z^{i*} \sim \text{Cat}\left(\left\{\frac{w(Z_k^i)}{\sum_{k'=1}^K w(Z_{k'}^i)}\right\}_{k=1}^K\right). \tag{3}$$

By Proposition C-A, this trace Z^{i*} is sampled from the improved distribution $\tilde{q}(Z|C^i, A^i)$ that provides a tighter bound on the true posterior than the original $q(Z|C^i, A^i)$.

D. Contextualizing Information Benefit in SIR IWAE

Our algorithm leverages the sampling-importance-resampling (SIR) procedure to provide a rigorous framework for quantifying the value of reasoning in terms of its information benefit for expert action prediction. To understand this connection, we relate the improved posterior distribution $\tilde{q}(Z|A)$ induced by SIR IWAE to the specific reasoning strategies selected during resampling. We demonstrate that this categorical resampling step naturally shifts the distribution of reasoning strategies by exactly the amount they improve action prediction; specifically, the model's "Refined Preference" for a strategy mathematically decomposes into its "Information Benefit" plus its initial "Warmstart Preference." Consequently, reasoning strategies that reduce divergence from the expert action distribution are automatically amplified, while distracting strategies are suppressed, allowing the model to autonomously filter for action-predictive reasoning without external supervision.

Recall our definition of information benefit for a reasoning strategy $\mathbf{R}$:

$$\Delta\mathcal{I}_{\mathbf{R}} \doteq D_{\text{KL}}(p_{\text{data}} \| p(A|C, \mathcal{Z}_{\not\mathbf{R}})) - D_{\text{KL}}(p_{\text{data}} \| p(A|C, \mathcal{Z}_{\mathbf{R}})),$$

where $\mathcal{Z}_{\mathbf{R}}$ denotes the set of reasoning traces that include strategy $\mathbf{R}$, and $\mathcal{Z}_{\not\mathbf{R}}$ denotes traces that exclude it. A positive $\Delta\mathcal{I}_{\mathbf{R}}$ indicates that reasoning with strategy $\mathbf{R}$ brings the model's action distribution closer to the expert's, while a negative value indicates that the strategy is distracting.

1) How SIR Reweights Reasoning Strategies: The improved posterior $\tilde{q}(Z|A)$ induced by categorical resampling does not uniformly upweight all reasoning traces. Instead, it selectively amplifies traces based on their importance weights $w(Z) = \frac{p(Z,A|C)}{q(Z|C,A)}$, which measure how well each trace aligns with both the model's generative distribution and the expert action.

To formalize this, consider partitioning the reasoning space into $\mathcal{Z}_{\mathbf{R}}$ and $\mathcal{Z}_{\not\mathbf{R}}$. When we draw K candidate samples $\{Z_k\}_{k=1}^K \sim q(Z|C, A)$ and resample according to normalized importance weights, the probability that the resampled trace $Z^* \sim \tilde{q}(Z|C, A)$ belongs to $\mathcal{Z}_{\mathbf{R}}$ is:

$$\begin{aligned}\tilde{q}(\mathcal{Z}_{\mathbf{R}} \mid C, A) &= \mathop{\mathbb{E}}_{Z_1,\ldots,Z_K \sim q(Z|C,A)} \left[\sum_{k=1}^{K} \frac{w(Z_k)}{\sum_{k'=1}^{K} w(Z_{k'})} \cdot \mathbf{1}_{Z_k \in \mathcal{Z}_{\mathbf{R}}} \right] \\ &= \mathop{\mathbb{E}}_{Z_1,\ldots,Z_K \sim q(Z|C,A)} \left[\frac{\frac{1}{K}\sum_{k=1}^{K} w(Z_k) \cdot \mathbf{1}_{Z_k \in \mathcal{Z}_{\mathbf{R}}}}{\frac{1}{K}\sum_{k'=1}^{K} w(Z_{k'})} \right]. \end{aligned} \tag{4}$$

This expression captures the stochastic resampling process: each candidate Z_k is selected with probability proportional to its importance weight $w(Z_k)$, so this is the probability that the selected candidate belongs to $\mathcal{Z}_{\mathbf{R}}$.

2) The Large-K Limit: Relating SIR to Expected Importance Weights: To understand how SIR affects strategy selection, we analyze the large-K limit where the law of large numbers provides intuition. As $K \to \infty$, the sample averages converge to expectations:

$$\frac{1}{K}\sum_{k=1}^{K} w(Z_k) \cdot \mathbf{1}_{Z_k \in \mathcal{Z}_{\mathbf{R}}} \xrightarrow{K\to\infty} \mathop{\mathbb{E}}_{Z\sim q}[w(Z) \cdot \mathbf{1}_{Z \in \mathcal{Z}_{\mathbf{R}}}]. \tag{5}$$

We can decompose this expectation by conditioning on the set membership:

$$\begin{aligned}\mathop{\mathbb{E}}_{Z\sim q}[w(Z) \cdot \mathbf{1}_{Z \in \mathcal{Z}_{\mathbf{R}}}] &= \int_{\mathcal{Z}_{\mathbf{R}}} w(z)\, q(z|C, A)\, dz \\ &= \underbrace{\left(\int_{\mathcal{Z}_{\mathbf{R}}} q(z|C, A)\, dz \right)}_{q(\mathcal{Z}_{\mathbf{R}}|C,A)} \cdot \underbrace{\frac{\int_{\mathcal{Z}_{\mathbf{R}}} w(z)\, q(z \mid C, A)\, dz}{\int_{\mathcal{Z}_{\mathbf{R}}} q(z|C, A)\, dz}}_{\mathbb{E}_{Z\sim q}[w(Z)|Z\in\mathcal{Z}_{\mathbf{R}}]} \\ &= q(\mathcal{Z}_{\mathbf{R}}|C, A) \cdot \mathop{\mathbb{E}}_{Z\sim q}[w(Z)|Z \in \mathcal{Z}_{\mathbf{R}}]. \end{aligned} \tag{6}$$

Similarly, the denominator converges:

$$\frac{1}{K}\sum_{k'=1}^{K} w(Z_{k'}) \xrightarrow{K\to\infty} \mathop{\mathbb{E}}_{Z\sim q}[w(Z)]. \tag{7}$$

Therefore, in the large-K limit, the improved posterior assigns the following probability to $\mathcal{Z}_{\mathbf{R}}$:

$$\tilde{q}(\mathcal{Z}_{\mathbf{R}}|C, A) = \frac{q(\mathcal{Z}_{\mathbf{R}}|C, A) \cdot \mathbb{E}_{Z\sim q}[w(Z)|Z \in \mathcal{Z}_{\mathbf{R}}]}{\mathbb{E}_{Z\sim q}[w(Z)]}. \tag{8}$$

This demonstrates the reweighting mechanism in the new categorical importance sampled distribution: the improved posterior $\tilde{q}$ modifies the original probability $q(\mathcal{Z}_{\mathbf{R}}|C, A)$ by a factor equal to the ratio of the conditional expected importance weight (within $\mathcal{Z}_{\mathbf{R}}$) to the overall expected importance weight.

3) Quantifying Strategy Preference Through Probability Ratios: To understand how SIR shifts preference between reasoning strategies, we examine the ratio of probabilities assigned to $\mathcal{Z}_{\mathbf{R}}$ versus $\mathcal{Z}_{\not\mathbf{R}}$. For the improved posterior:

$$\frac{\tilde{q}(\mathcal{Z}_{\mathbf{R}}|C,A)}{\tilde{q}(\mathcal{Z}_{\not\mathbf{R}}|C,A)} = \frac{\mathbb{E}_{Z\sim q}[w(Z)|Z\in\mathcal{Z}_{\mathbf{R}}]}{\mathbb{E}_{Z\sim q}[w(Z)|Z\in\mathcal{Z}_{\not\mathbf{R}}]} \cdot \frac{q(\mathcal{Z}_{\mathbf{R}}|C,A)}{q(\mathcal{Z}_{\not\mathbf{R}}|C,A)} \tag{9}$$

We consider two components:

- **Warmstart Preference for R:** $\log \frac{q(\mathcal{Z}_{\mathbf{R}}|C,A)}{q(\mathcal{Z}_{\not\mathbf{R}}|C,A)}$ represents the (log) prior preference for strategy $\mathbf{R}$ in the original proposal distribution q, which is induced by our warmstart training with reasoning dropout rate d.
- **Importance Weight Ratio:** The term $\frac{\mathbb{E}_{Z\sim q}[w(Z)|Z\in\mathcal{Z}_{\mathbf{R}}]}{\mathbb{E}_{Z\sim q}[w(Z)|Z\in\mathcal{Z}_{\not\mathbf{R}}]}$ quantifies how much more (or less) the model values traces with strategy $\mathbf{R}$ in terms of their alignment with the joint distribution $p(Z, A|C)$.

From our warmstart training procedure (Alg. 1), we construct the latent dataset with reasoning dropout such that each strategy $\mathbf{R}$ is independently included with probability $1-d$ and excluded with probability d. So, as in the assumptions from Section B, this induces:

$$\frac{q(\mathcal{Z}_{\mathbf{R}}|C,A)}{q(\mathcal{Z}_{\not\mathbf{R}}|C,A)} = \frac{1-d}{d}. \tag{10}$$

Substituting Equation (10) into Equation (9), we obtain:

$$\frac{\tilde{q}(\mathcal{Z}_{\mathbf{R}}|C,A)}{\tilde{q}(\mathcal{Z}_{\not\mathbf{R}}|C,A)} = \frac{\mathbb{E}_{Z\sim q}[w(Z)|Z\in\mathcal{Z}_{\mathbf{R}}]}{\mathbb{E}_{Z\sim q}[w(Z)|Z\in\mathcal{Z}_{\not\mathbf{R}}]} \cdot \frac{1-d}{d}. \tag{11}$$

We define the **Refined Preference for $\mathbf{R}$** as the log-ratio of probabilities of reasoning with strategy $\mathbf{R}$ and without strategy $\mathbf{R}$ in the improved posterior, averaged over the expert action distribution:

$$\textbf{Refined Preference for } \mathbf{R} := \mathop{\mathbb{E}}_{A\sim p_{\text{data}}}\left[\log \frac{\tilde{q}(\mathcal{Z}_{\mathbf{R}}|C,A)}{\tilde{q}(\mathcal{Z}_{\not\mathbf{R}}|C,A)}\right].$$

Note that the improved posterior distrution $\tilde{q}(\cdot \mid C, A)$ is the source for the refined reasoning training data for the VLA model in Alg. 2.

Expanding this expectation:

$$\begin{aligned}
\mathop{\mathbb{E}}_{A\sim p_{\text{data}}}\left[\log \frac{\tilde{q}(\mathcal{Z}_{\mathbf{R}}|C,A)}{\tilde{q}(\mathcal{Z}_{\not\mathbf{R}}|C,A)}\right] &= \mathop{\mathbb{E}}_{A\sim p_{\text{data}}}\left[\log \frac{\mathbb{E}_{Z\sim q}[w(Z)|Z\in\mathcal{Z}_{\mathbf{R}}]}{\mathbb{E}_{Z\sim q}[w(Z)|Z\in\mathcal{Z}_{\not\mathbf{R}}]} + \log\frac{1-d}{d}\right] \\
&= \underbrace{\mathop{\mathbb{E}}_{A\sim p_{\text{data}}}\left[\log \frac{\mathbb{E}_{Z\sim q}[w(Z)|Z\in\mathcal{Z}_{\mathbf{R}}]}{\mathbb{E}_{Z\sim q}[w(Z)|Z\in\mathcal{Z}_{\not\mathbf{R}}]}\right]}_{\Delta\mathcal{I}_{\mathbf{R}} \text{ (by Proposition of Sec IV-C)}} + \underbrace{\log\frac{1-d}{d}}_{\text{Warmstart Preference}} \\
&= \Delta\mathcal{I}_{\mathbf{R}} + \log\frac{1-d}{d}.
\end{aligned} \tag{12}$$

4) Information Benefit Drives Strategy Selection: Equation (12) reveals the fundamental insight that the refined preference for a reasoning strategy $\mathbf{R}$, which characterizes the new refined reasoning training dataset, decomposes into two terms:

$$\textbf{Refined Preference for } \mathbf{R} = \textbf{Information Benefit of } \mathbf{R} + \textbf{Warmstart Preference for } \mathbf{R}$$

This decomposition has a clear interpretation:

- The **Warmstart Preference** ($\log \frac{1-d}{d}$) is a constant baseline that reflects the prior probability of including strategy $\mathbf{R}$ during warmstart training.
- The **Information Benefit** ($\Delta\mathcal{I}_{\mathbf{R}}$) represents the additional preference shift induced by categorical resampling based on importance weights. This term is strategy-specific and captures how much the strategy improves action prediction.

This demonstrates that the `R&B-EnCoRe`'s SIR IWAE procedure systematically adjusts the distribution of reasoning strategies based on their utility for action prediction:

a) Positive Information Benefit ($\Delta\mathcal{I}_{\mathbf{R}} > 0$): When strategy $\mathbf{R}$ improves action prediction, traces in $\mathcal{Z}_{\mathbf{R}}$ receive higher importance weights on average than traces in $\mathcal{Z}_{\not\mathbf{R}}$. Categorical resampling amplifies the probability of selecting traces with $\mathbf{R}$, increasing its representation in the refined dataset beyond the warmstart baseline.

b) Negative Information Benefit ($\Delta\mathcal{I}_{\mathbf{R}} < 0$): When strategy $\mathbf{R}$ distracts from action prediction, traces in $\mathcal{Z}_{\mathbf{R}}$ receive lower importance weights on average. Categorical resampling suppresses the probability of selecting traces with $\mathbf{R}$, decreasing its representation below the warmstart baseline.

c) Zero Information Benefit ($\Delta\mathcal{I}_{\mathbf{R}} = 0$): When strategy $\mathbf{R}$ provides no information about actions, the importance weight ratio is unity (in expectation), and the refined preference equals the warmstart preference. The strategy's representation remains unchanged.

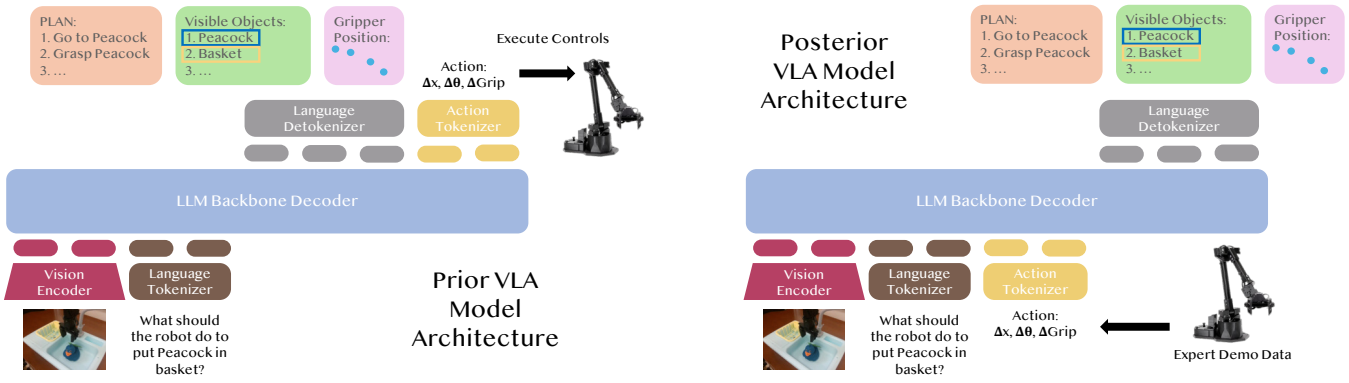


Fig. 14: **Prior and Posterior architecture.** The prior architecture is the same as the standard generative VLA that takes as input the task context (scene and task) and outputs textual reasoning followed by action tokens. The posterior architecture takes as input the context *and action* and outputs only the reasoning tokens.

TABLE VI: Experimental Configuration Details Across Embodiment Domains

Configuration	LIBERO-90	WidowX Hardware	Legged Robot	Autonomous Vehicle
M (warmstart traces)	10	4	8	32
K (posterior samples)	8	4	4	8,12,16,32
ρ (reasoning primitives)	7	7	7,8 (w/ or w/o Weather)	6
Dropout rate d	0.2	0.2	0.1,0.3,0.5.0.7,0.9	0.5
Posterior sampling temp.	1.0	1.0	0.7	1.0
Expert demo source	LIBERO-90 [96]	Bridgev2 [30]	NaviTrace [104]	nuScenes [32]
Primary reasoning primitive content source	Llama 2 [15], Molmo [16]	Gemini 1.0 [110]	Qwen3-VL 30B MoE [17]	Human Annotations [85, 86]
VLA architecture	MiniVLA [6]: Qwen2.5 0.5B LLM [97], DINOv2 [98]+SigLIP [99],VQ-VAE [101]	OpenVLA [5]: Llama 2 7B LLM [15], DINOv2 [98]+SigLIP [99]	Qwen3-VL 30B MoE [17]	Qwen3-VL 4B Dense [17]

5) Relating to `R&B-EnCoRe`*:* By employing SIR IWAE to generate refined reasoning traces, `R&B-EnCoRe` implements a principled, self-supervised filtering process that:

1) Starts with diverse reasoning strategies sampled uniformly via reasoning dropout (warmstart)
2) Measures each strategy's alignment with expert actions through importance weights
3) Automatically concentrates probability mass on action-predictive strategies through categorical resampling
4) Produces a refined training dataset where strategy prevalence correlates with information benefit

Importantly, this entire process requires no external rewards, verifiers, or heuristics—only the generative probabilities of the model itself. The improved posterior $\tilde{q}(Z|C, A)$ discovers embodiment-specific reasoning distributions (Fig. 4) by maximizing information benefit for action prediction. Through this mechanism, `R&B-EnCoRe` can identify distinct reasoning strategies across manipulation, legged navigation, and autonomous driving domains, with each embodiment benefiting from a unique distribution of reasoning primitives tailored to its specific embodiment and task challenges.

E. Scaling with Inference Compute

A key advantage of the categorical resampling approach is that performance improves monotonically with the number of samples K. From the IWAE literature [94, 95], we know that:

$$\tilde{\mathcal{L}}_{K+1} \geq \tilde{\mathcal{L}}_K, \tag{13}$$

for $K \geq 1$, with the bound approaching $\log p(A)$ as $K \to \infty$ when $\tilde{q}(Z|A)$ converges to $p(Z|A)$.

In our experiments on scaling sampling for AV VLA models (Fig. 12), we empirically validate this theoretical prediction, showing that increasing K from 8, 12, 16 to 32 consistently improves downstream performance on metrics like collision rate. This demonstrates a practical avenue for scaling inference compute to improve data efficiency during pretraining—by investing more compute in the posterior sampling step to generate higher-quality reasoning traces, we empirically observe better model performance.

APPENDIX D
EXPERIMENTAL DETAILS

This section provides comprehensive details on the training procedures, inference protocols, and reasoning primitive specifications for all experimental domains evaluated in this work.

A. LIBERO-90 Manipulation

1) Model Architecture and Training: We employ the MiniVLA architecture [6] with a 1B-parameter configuration, consisting of a 0.5B Qwen2.5 language model backbone [97] combined with DINOv2 [98] and SigLIP [99] vision encoders. Actions are represented as seven discrete tokens encoding a 10-step action chunk [6, 37] via VQ-VAE tokenization [101] with codebook size of 256.

Implementation. We adapt the codebase from [11] to implement MiniVLA with VQ-VAE action tokenization. Note that while we follow the architectural principles from [12], their code, models, and detailed hyperparameters were not publicly available at the time of our experiments, so we could not directly verify or reproduce their specific implementation details.

Warmstart Training. The prior-posterior model $\mathcal{M}_{pq}$ is trained on 64 NVIDIA A100 GPUs (40GB) with a total batch size of 512. Training continues until the model achieves 95% action token prediction accuracy on the training set. We apply reasoning dropout with rate $d = 0.2$ to generate diverse reasoning strategy combinations as described in Alg. 1.

Reasoning VLA Training. Following the refinement procedure (Alg. 2), we train the final VLA model $\mathcal{M}_{\text{VLA}}$ from the base model initialization to ensure fair comparison with baseline methods. This model is trained on the refined dataset $\mathcal{D}_{\text{refined}}$ with the same hardware configuration and stopping criterion (95% action token accuracy).

Posterior Sampling. During the refinement stage, we sample $K = 8$ reasoning candidates from the posterior distribution for each demonstration.

2) Reasoning Primitives: We utilize seven reasoning primitives for manipulation tasks, with content generated from the foundation models Llama 2 [15], Molmo [16] following the data from previous work [12]:

- **Plan**: High-level task decomposition into sequential subtasks.
- **Visible Objects**: Enumeration of objects detected in the scene using 2D bounding boxes.
- **Subtask**: Current subtask or immediate goal the robot should accomplish.
- **Subtask Explain**: Detailed justification for why the current subtask is necessary. In [12] this was called Subtask Reasoning, and we trained using the phrase Subtask Reasoning, but for the sake of clarity and differentiating the usage of the word Reasoning we call this Subtask Explain in the main paper.
- **Move**: Meta-action language description of the intended motion (e.g., "move gripper to object").
- **Move Explain**: Detailed reasoning explaining why the proposed movement is appropriate. Like Subtask Explain, this primitive was called Move Reasoning in the [12] and we trained our traces using the phrase Move Reasoning.
- **Gripper Position**: 2D waypoint representation of the end-effector's spatial coordinates.

3) Inferencing: At test time, the model receives a scene image and task description as context C. The model generates reasoning traces at each action step before predicting the action tokens. This step-by-step reasoning allows the policy to adapt its chain-of-thought to the evolving scene state throughout task execution.

Evaluation Protocol. We perform 20 rollout trials per task. Success is determined by task-specific completion criteria defined in the LIBERO-90 benchmark.

B. Bridgev2 WidowX Hardware Manipulation

1) Model Architecture and Training: We employ the OpenVLA architecture [5] with a 7B-parameter Llama 2 [15] language backbone with DINOv2 [98] and SigLIP [99] vision encoders. Actions are represented as seven discrete tokens following the tokenization scheme from [5, 37].

Implementation. We adapt the codebase from [11] for training the OpenVLA model with out algorithm on the Bridgev2 dataset [30].

Warmstart Training. The prior-posterior model $\mathcal{M}_{pq}$ is trained on 64 NVIDIA A100 GPUs (40GB) with a global batch size of 640, continuing until 95% action token accuracy is achieved. Reasoning dropout rate is set to $d = 0.2$.

Reasoning VLA Training. The final VLA model $\mathcal{M}_{\text{VLA}}$ is trained from base model initialization on the refined dataset $\mathcal{D}_{\text{refined}}$ with identical hardware and convergence criteria.

Posterior Sampling. During refinement, we sample $K = 4$ reasoning candidates from the posterior distribution for each demonstration. We adapted SGLang-VLA [102, 103] to support high-throughput batch inferencing of posterior distribution VLA architecture along with extraction of log probabilities of prior VLA architecture (needed for computation of importance weights).

Post-Training All Primitives Model for Action Forcing. For the baseline model trained with all reasoning primitives, we perform additional post-training epochs with reasoning-free examples to enable the action-forcing capability, following [12]. Models trained with `R&B-EnCoRe` and even random primitives acquire this capability naturally through dropout.

2) Reasoning Primitives: We utilize the same seven reasoning primitives like in LIBERO-90 (Plan, Visible Objects, Subtask, Subtask Explain, Move, Move Explain, Gripper Position), with reasoning content generated by the Gemini 1.0 model [110] from the data of [11]. The demonstration data is sourced from the Bridgev2 dataset [30].

3) Evaluation: During evaluation, we use a SGLang-VLA [102, 103] serving engine to rapidly inference the VLAs during test-time.

Evaluation with Suppressed Test-Time Reasoning (i.e., Action Forcing). In this evaluation mode (Fig. 15 and Table III), we suppress reasoning generation at test time appending the phrase `Action:` right at the end of the standard openvla prompt. This forces the model to directly produce action tokens without generating intermediate reasoning traces. We evaluate across 9 tasks (3 per category) totaling 468 trials, categorized into:

- **In-Distribution**: Tasks similar to training data
- **OOD Target**: Tasks with novel target grasp objects not seen during training
- **OOD Scene with Distractions**: Cluttered environments containing task-irrelevant objects

This evaluation protocol tests whether reasoning during training improves the learned policy representation, even when reasoning is not explicitly generated at deployment.

Evaluation with Test-Time Reasoning Enabled. In this evaluation mode (Fig. 7 and Table VII), models generate complete chain-of-thought reasoning traces before predicting actions. We evaluate on an in-distribution task and OOD scenes with distractions task to measure both task success and inference latency. Models trained on all primitives exhibit > 5 second per-step latency, while R&B-EnCoRe models achieve $\sim$3 second latency due to more concise reasoning traces.

C. Legged Robot Navigation

1) Model Architecture and Training: We finetune a Qwen3-VL-30B-A3B-Instruct [17] Mixture-of-Experts (MoE) model to process scene images and task specifications, outputting 2D waypoint coordinates for navigation. The model uses the NaviTrace dataset [104] containing approximately 1,000 tasks across 500 unique scenes for four embodiments: bipedal, wheeled robot, bicycle, and quadruped. Evaluation is performed on a 20% holdout subset of the scenes (to ensure no evaluation image data contamination in training data).

Training Infrastructure. We utilize the Thinking Machines Lab Tinker API for both inference and finetuning of the Qwen3-VL-30B MoE model. Training is performed with a batch size of 16 for 8 epochs.

Reasoning Dropout. Training employs reasoning dropout with rate $d = 0.5$ to generate diverse reasoning strategy combinations. We additionally ablate the dropout values as seen in Fig. 16.

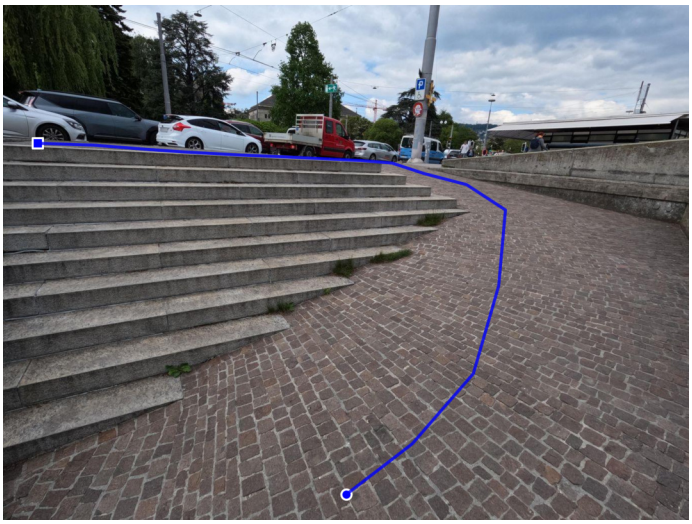
Posterior Sampling. During refinement, we sample $K = 4$ reasoning candidates from the posterior distribution for each demonstration.

2) Self-Generated Reasoning Primitives: Unlike manipulation tasks that rely on foundation models external to/different from the base VLM, we generate reasoning primitives by querying the base Qwen3-VL-30B model itself via visual question answering (VQA). For each demonstration, we provide the scene image *with the ground truth waypoint path overlaid* along with the task description.

Data Curation Protocol. For each scene in the NaviTrace dataset, we query the base Qwen3-VL-30B model with:

- The scene image with ground truth waypoint trajectory overlaid
- The task description
- Individual prompts for each reasoning primitive

An example image and prompt for generating content for the eight reasoning primitives (Terrain, Obstacles, Affordances, Plan, Move, Social, Counterfactuals, and Weather) is:



```
You are a navigation expert for various embodiments including robots and humans. Given a
    first-person image view of the current scenario with a planned path overlaid on the
    image, the coordinates of these 2D points in normalized image coordinates (ranging from
    0 to 1, where [0,0] is the top-left and [1,1] is the bottom-right), a specified
    embodiment (e.g., legged robot, wheeled robot, human, or bike), and a navigation task (e
    .g., "Go down the road"), you will provide a comprehensive, step-by-step, detailed
    reasoning trace in full sentences about how the overlaid path on the image solves the
    given task. Do not explicitly refer to the trajectory during the reasoning as future
    readers of your reasoning trace will not see the overlaid path.
Your reasoning must be provided within the specified XML-style tags for each distinct
    category.

###Navigation Reasoning Components
Terrain: A detailed analysis of the physical characteristics of the ground surfaces in the
    scene, including the types of stepping, rolling, or contact considerations required (e.g
    ., grass vs. sidewalk, uneven cobblestones, smooth asphalt).
Obstacles: An identification of specific objects, obstacles, or areas anywhere in the scene
    that the embodiment must actively avoid colliding with or entering.
Affordances: Description of specific action capabilities a region of the environment offers
    to the robot--such as steppable, jumpable, or gap-crossable--determined by the interplay
     between the terrain's physical properties and the robot's kinematic and dynamic limits.
Plan: High-level language description of the overall navigation plan.
Move: Description of primary direction and type of motion required to follow the path (e.g.,
     move forward, turn left 45 degrees, slow down).
Social: Description of expected or required social behaviors and conventions that the
    depicted trajectory does to maintain safe and cooperative movement around other entities
     (e.g., move left for bikers, right for humans or yield to oncoming traffic).
Counterfactuals: Listing of potential alternative navigation plans and their likely
    consequences, presented as pairs of Action and Consequence.
Weather: An analysis of the weather in the environment using subjective language.

###Output Format
You must format your final output using the following tags for each corresponding type of
    reasoning:
<Terrain>
Description of physical properties of surface contact patches.
</Terrain>

<Obstacles>
Identify objects in scene to not hit.
</Obstacles>

<Affordances>
Region: [Environment region/terrain 1]
```

```
Affordances: [Action capabilities of region/terrain 1]

Region: [Environment region/terrain 2]
Affordances: [Action capabilities of region/terrain 2]
...
</Affordances>

<Plan>
Step 1: [First step in plan]
Step 2: [Second step in plan]
...
</Plan>

<Move>
Directions of motions.
</Move>

<Social>
Applicable social rules for movement.
</Social>

<Counterfactuals>
Action: [Alternative action 1]
Consequence: [Likely result of alternative action 1]

Action: [Alternative action 2]
Consequence: [Likely result of alternative action 2]
...
</Counterfactuals>

<Weather>
Description of the weather.
</Weather>

###Additional Notes about the Image:
  - The image shows a first-person view of the navigation scenario
  - The trajectory starts near the bottom center of the image, which corresponds
    approximately to normalized coordinate [0.5, 0.95] (representing the current position of
     the embodiment)
  - The trajectory is adapted to the embodiment's abilities and limitations
  - The trajectory extends all the way to the goal if the path is visible. If the path is
    occluded, the trajectory ends where the path becomes fully obscured, unless the path can
     be reasonably inferred from the visible context.

### Information on General Embodiment Movement Characteristics
  - **Human**: A standard pedestrian. Can navigate stairs and ramps but cannot climb tall
    obstacles.
  - **Wheeled Robot**: A wheeled delivery robot. Behaves like a wheelchair, preferring
    smooth surfaces such as walkways and ramps. It cannot use stairs or escalators.
  - **Bicycle**: A standard cyclist. Follows traffic regulations and prefers bike lanes or
    streets. Cannot navigate stairs.
  - **Legged Robot**: A quadruped like ANYmal. Behaves similarly to a human, but it is
    shorter. It can handle stairs and escalators.

### Your Task Specific Information
**Trajectory**: [[0.50, 0.95], [0.60, 0.85], [0.69, 0.72],[0.72, 0.55],[0.73, 0.40], [0.68,
    0.35], [0.61, 0.33], [0.57, 0.31], [0.52, 0.30], [0.47, 0.30], [0.41, 0.29], [0.34,
    0.29], [0.26, 0.28], [0.21, 0.28], [0.13, 0.27], [0.05 , 0.27]]
**Your current Embodiment**: Wheeled Robot
**Task**: Follow the road to the left
```

and the output from the Qwen3-VL-30b Instruct model was:

```
<Terrain>
The terrain consists of two main surfaces: a set of wide, stone steps on the left and a
    cobblestone path on the right. The steps are made of large, flat stone slabs and are
```

```
    designed for walking. The cobblestone path is composed of small, irregularly shaped
    stones set in a pattern, creating a textured but generally stable surface. For a wheeled
     robot, the cobblestone path is navigable but may require careful control due to the
    unevenness, while the steps are completely impassable.
</Terrain>

<Obstacles>
The primary obstacles are the stone steps on the left, which the robot cannot traverse.
    Additionally, there are parked cars and a red truck in the background, which are static
    obstacles that must be avoided. The presence of a person in the distance on the right
    side of the path is also noted, although they appear to be stationary and not in the
    immediate path of the robot.
</Obstacles>

<Affordances>
Region: Cobblestone path
Affordances: The cobblestone path is suitable for a wheeled robot to move across, although
    it may require adjustments in speed and steering to handle the uneven surface.

Region: Stone steps
Affordances: The stone steps are not suitable for a wheeled robot to traverse and must be
    avoided.
</Affordances>

<Plan>
Step 1: Begin moving forward along the cobblestone path, staying to the right of the steps
    to avoid them.
Step 2: Continue moving forward, maintaining a steady course along the path as it curves
    slightly to the left.
Step 3: Ensure to navigate around any potential obstacles, such as the parked cars and the
    person, by adjusting the path slightly if necessary.
Step 4: Continue until the path is fully traversed or the goal is reached.
</Plan>

<Move>
Move forward along the cobblestone path, navigating the slight curve to the left. Maintain a
     steady pace and ensure to stay clear of the steps on the left and any obstacles on the
    path.
</Move>

<Social>
The robot should maintain a safe distance from the parked cars and the person on the right
    side of the path. It should not obstruct the path of any pedestrians and should move at
    a pace that allows for smooth and safe navigation.
</Social>

<Counterfactuals>
Action: Attempt to move up the stone steps
Consequence: The wheeled robot cannot traverse the steps and would become stuck or damaged.

Action: Move directly towards the parked cars
Consequence: The robot would collide with the vehicles, causing damage and potentially
    obstructing traffic.
</Counterfactuals>

<Weather>
The weather appears to be overcast, with a cloudy sky and no direct sunlight visible. The
    clouds are thick and gray, suggesting a cool and possibly damp environment. There are no
     signs of rain or strong winds, and the overall lighting is diffused, creating a soft
    and even illumination across the scene.
</Weather>
```

We parsed out the individual contents for reasoning primitives from the corresponding XML synthetic textual data to get the corresponding reasoning.

3) Inference Protocol: At test time, the model receives the scene image and task description, generating 2D waypoint coordinates. Evaluation uses the NaviTrace metric [104] incorporating Dynamic Time Warping distance, goal endpoint error,

and semantic penalties aligned with human preferences.

D. Autonomous Driving

1) Model Architecture and Representation: We finetune a Qwen3-VL-4B-Instruct [17] dense (non-MoE) model on the nuScenes dataset [32] which consists of almost 1000 driving scenes of 20 seconds each, consisting of images and trajectory data sampled at 2Hz, collected from Boston and Singapore and we use the official training and validation split. We follow [85, 86] and use 2 seconds of history to predict 3-second of future trajectories at a 2Hz frequency. We rely on a natural language representation of the trajectory, formatted as pairs of (x,y) in nuScenes ego-centric coordinates, for a total of 6 numbers. See an example below. Different from prior works, we process the front-camera RGB image only in order to remain within the computational budget. We find that empirically, we perform similarly to methods using a surround-view setup [44, 85], suggesting that future work can improve VLMs' understanding of multiple frames.

Implementation. We employed the Qwen3-VL 4B [17] dense model to initialize the autonomous driving VLA. Training employs reasoning dropout with rate $d = 0.5$.

Posterior Sampling. During refinement, we sample $K = 16$ reasoning candidates from the posterior distribution for each demonstration (results reported in Table III use this configuration). We ablate the effect of K in Fig. 12, finding that performance saturates around $K = 16$ samples.

2) Reasoning Primitives: We utilize reasoning traces from human-annotated data for the nuScenes dataset, originally curated for agent-based planning methods [85, 86]. For this work, we re-distributed them into reasoning primitives that more evenly break down the original annotations. The new primitives, based on the annotations, include:

- **Mission Goal**: High-level navigation objective (e.g., "turn left", "change lane right"). This information is extracted from the "ego" field of the annotation from [86], repurposed for driving VLA reasoning.
- **Perception**: Detailed scene understanding including detected objects, their types, IDs, and predicted future waypoints. This field corresponds to the "perception" annotations.
- **Notable Objects**: Salient objects requiring attention. Notable objects are extracted from original "reasoning" field of the annotations, and are objects that a language model identified as notable.
- **Collidable Objects**: Objects with collision risk. These correspond to the notable objects in the "chain_of_thoughts" field of the annotations, and are objects the ego-vehicle will collide with if the path is kept.
- **Driving Plan**: Concrete action plan with speed modulation (e.g., "change lane right with constant speed"). This is extracted from the driving plan in the "chain_of_thoughts" field of the annotations, and is heuristically generated.
- **Experience**: Retrieved similar past driving scenarios from memory with confidence scores. This corresponds exactly to the "experiences" field of the annotations.

3) Data Transformation and Evaluation: The original nuScenes dataset was matched to the reasoning annotations from [86], and details corresponding to the ego-vehicle were extracted to provide to the driving VLA. This information corresponds to the "ego" field in the annotations, and include current state (velocities, acceleration, can bus information, etc.) as well as the historical trajectory (2-second past information). The mission goal is not included, and is instead used for the reasoning. All reasoning is headed by a question, formatted such that the reasoning type is the answer, and in tags, e.g.,:

```
<What is the mission goal?> FORWARD

<What do you perceive in the scene?>

Distance to both sides of road shoulders of current ego-vehicle location:
Current ego-vehicle's distance to left shoulder is 5.5m and right shoulder is 1.5m

<What is a similar past experience?>
Most similar driving experience from memory with confidence score: 1.00:
The planned trajectory in this experience for your reference:
[(-0.00,0.00), (-0.00,0.00), (-0.00,-0.00), (-0.00,0.16), (-0.01,0.60), (0.01,1.40)]
<What are the collidable objects?>
 - Notable Objects: None
   Potential Effects: None<What is the driving plan?> STOP
<What are the notable objects?>
 - Notable Objects: None
   Potential Effects: None

Planned Trajectory:
[(0.05,4.55), (0.09,9.14), (0.12,12.84), (0.19,17.51), (0.22,22.14), (0.25,26.60)]
```

Training Data Details. During training, the model takes as input the front camera image, ego annotation, and the task. The model optionally predicts a reasoning trace in the above format, and a planned trajectory. We train on 4 NVIDIA H100 GPUs with a global batch size of 32, for a fixed 30 epoch training run, using the provided finetuning infrastructure.

Evaluation Details. We follow the UniAD evaluation [107] for open-loop evaluation of the planned trajectory. The predicted trajectory is parsed into the evaluation format. If there is an error in parsing—which occurs <0.5% of the time—it is replaced with a stationary trajectory to not bias the evaluation results.

APPENDIX E
HARDWARE DETAILS WITH RESULTS

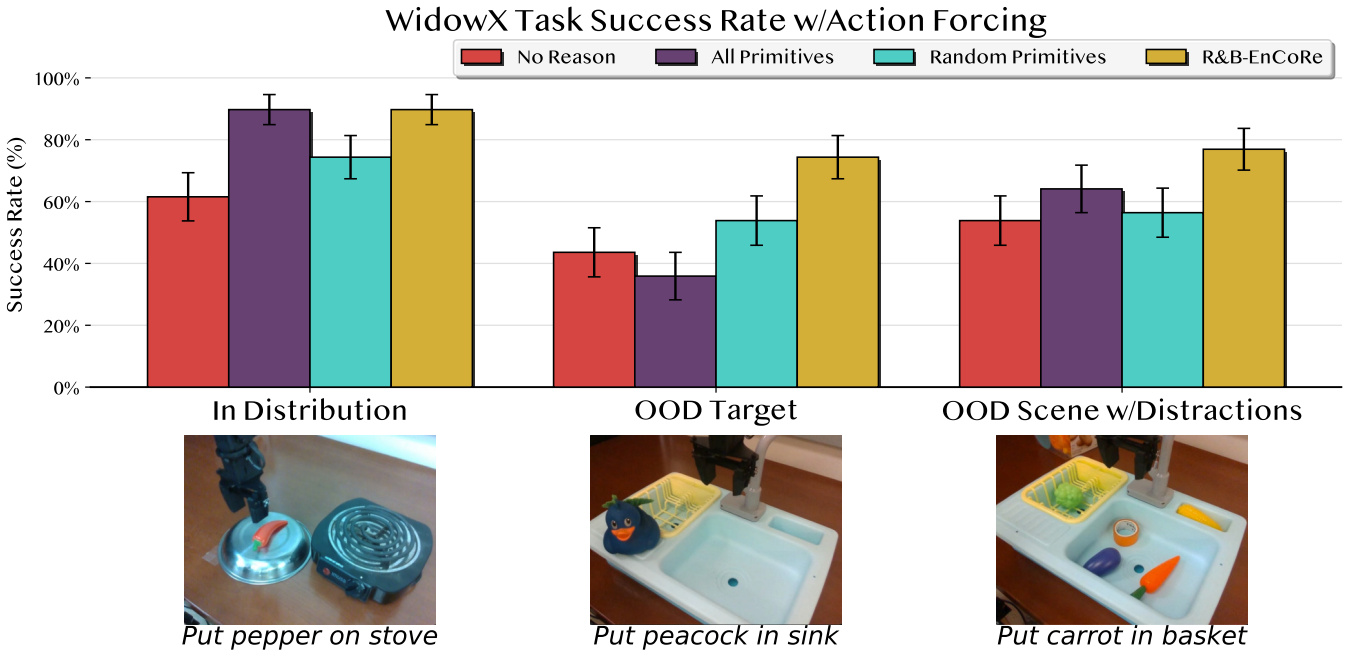


Fig. 15: Success rates on WidowX hardware Bridgev2 setup. Reasoning Models are prompted and/or trained with Action forcing for reduced latency [12]. Evaluated on a total of 9 different tasks (3 per category), and 468 total trials. We show 1 example task per category.

TABLE VII: WidowX Task Success Rate with Test-Time Reasoning Enabled. Evaluation includes 13 trials per task and model to compare performance when the model generates full chain-of-thought traces.

Category	**Task**	**All Primitives**	**R&B-EnCoRe**
In Distrib.	put red pepper in yellow basket	84.6%	**92.3%**
OOD Scene w/Distr.	put red pepper in yellow basket *(include distraction objects in basket, sink)*	53.8%	**84.6%**

APPENDIX F
ADDITIONAL ABLATION EXPERIMENTS

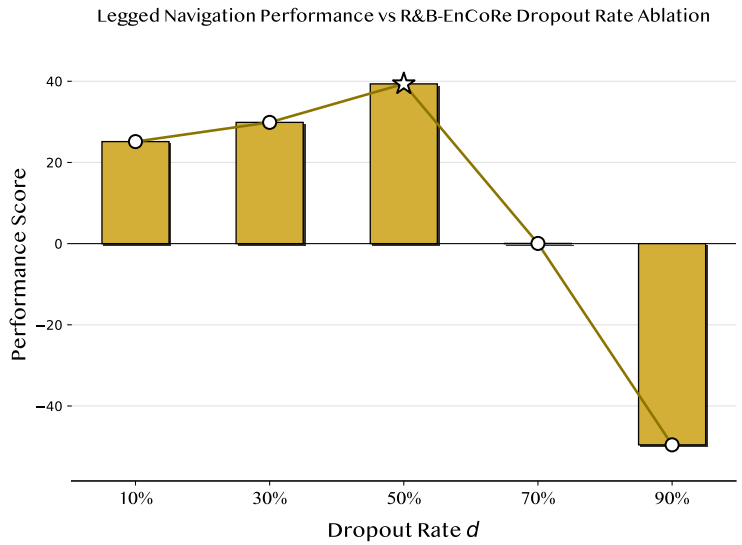


Fig. 16: For our `R&B-EnCoRe` algorithm applied to the Legged navigation embodiments, we perform an ablation study on varying the dropout rate parameter d that affects the initial warmstart reasoning strategy training distribution. We find that 50% dropout provides best downstream performance. This dropout rate encourages the prior and posterior model to see a diverse set of reasoning strategies with overall minimal warmstarting preference bias for any strategy (i.e., $\log \frac{1-d}{d} = \log \frac{1-0.5}{0.5} = 0$).

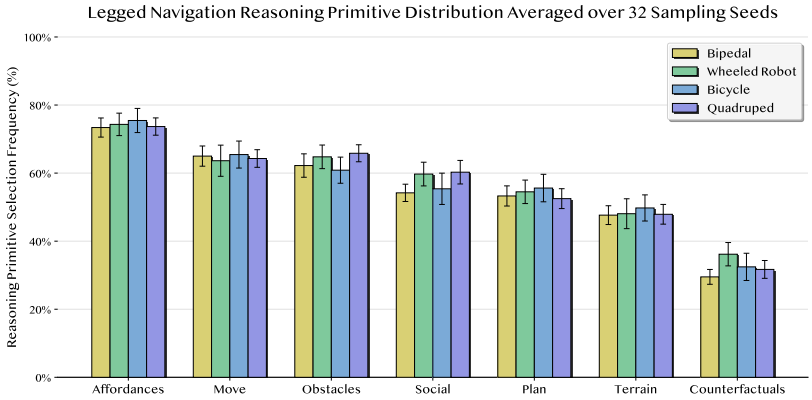


Fig. 17: For the Legged Navigation Dataset we perform an ablation on performing posterior sampling (from Alg. 2) across 32 different sampling seed to validate whether the refined reasoning primitive distribution remains consistent. This plot confirms the generally consistency of the reasoning primitive frequencies (note the error bars and compare with the result of a single sample seed of Fig. 4b).

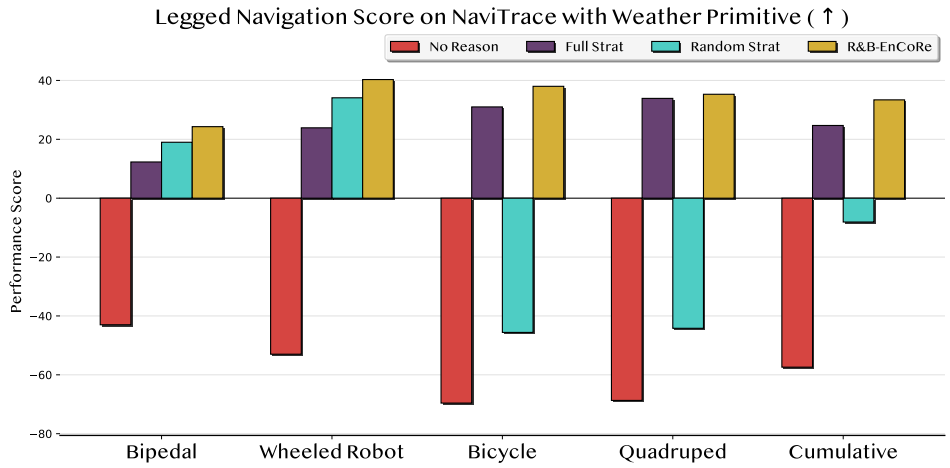


Fig. 18: NaviTrace scores on the various VLA models with the additional Weather Reasoning primitive. `R&B-EnCoRe` refines the traces to remove irrelevant Weather reasoning primitive scores as seen in Fig. 10, and results in best performance across all embodiments on the navigation metric from [104]. Recalls the NaviTrace score is a normalized metric so 100 is perfect path alignment with expert, and 0 is the score for the naïve straight line path down the middle of the scene.

APPENDIX G
EXAMPLE REASONING TRACES ACROSS EMBODIMENTS

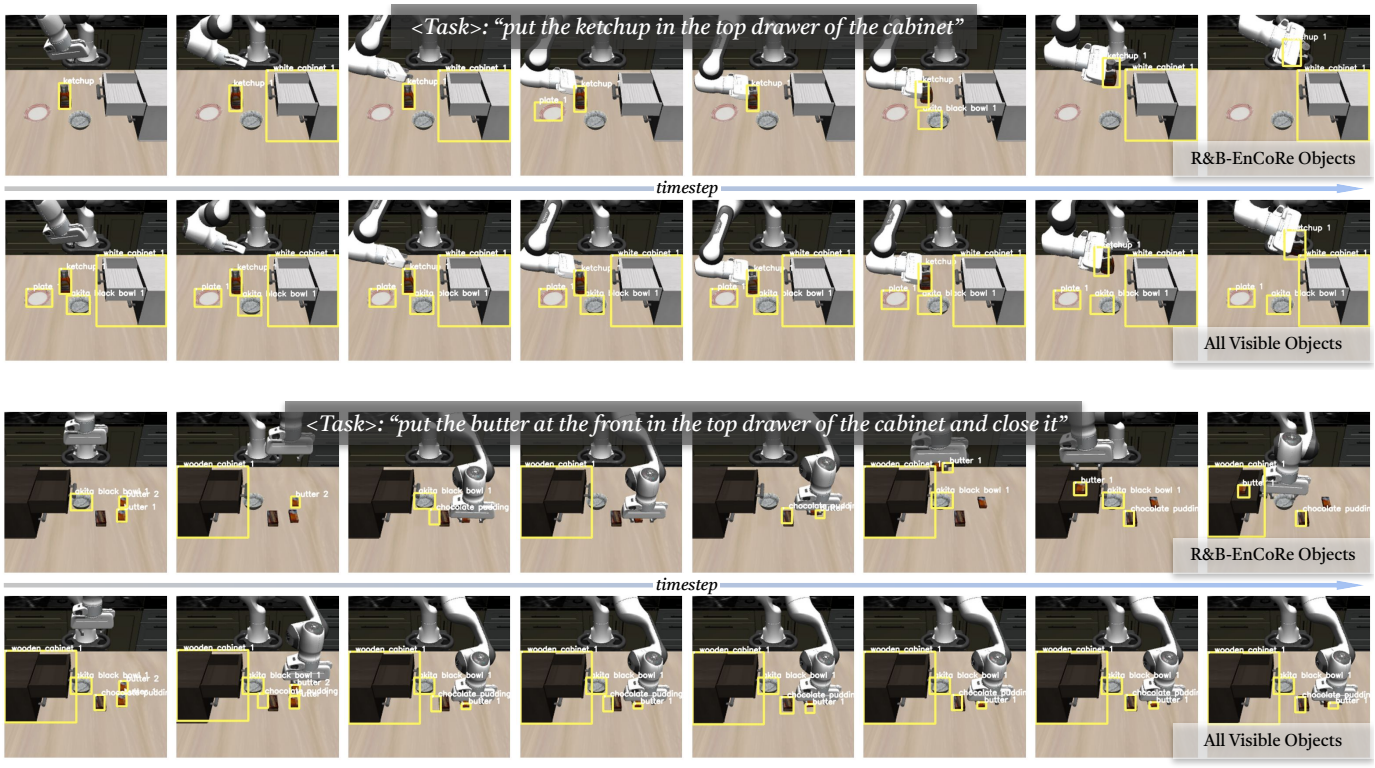


Fig. 19: Visible Object-only reasoning (Section V-A-Q1) in LIBERO-90 across steps in episode. Notice that the generated object bounding boxes for R&B-EnCoRe's reasoning model generally attend to primarily task salient objects, while reasoning with all visible objects attends to all objects (including distracting/irrelevant ones) at every frame.

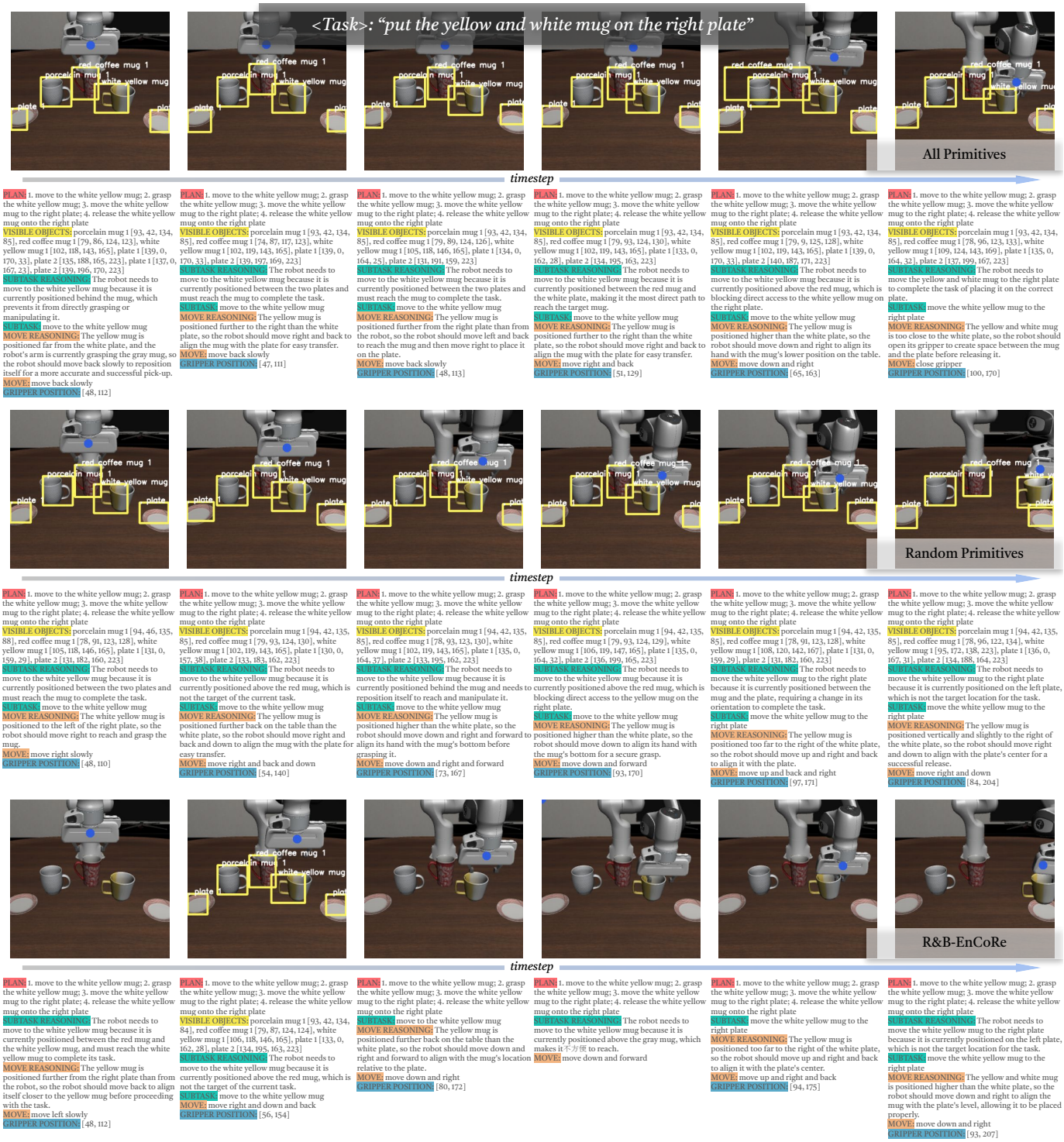


Fig. 20: Reasoning Traces with wider set of primitives (Section V-A-Q2) from the reasoning VLAs for LIBERO-90 across episode. Notice how R&B-EnCoRe reasons less frequently about visible objects compared with the other two models.

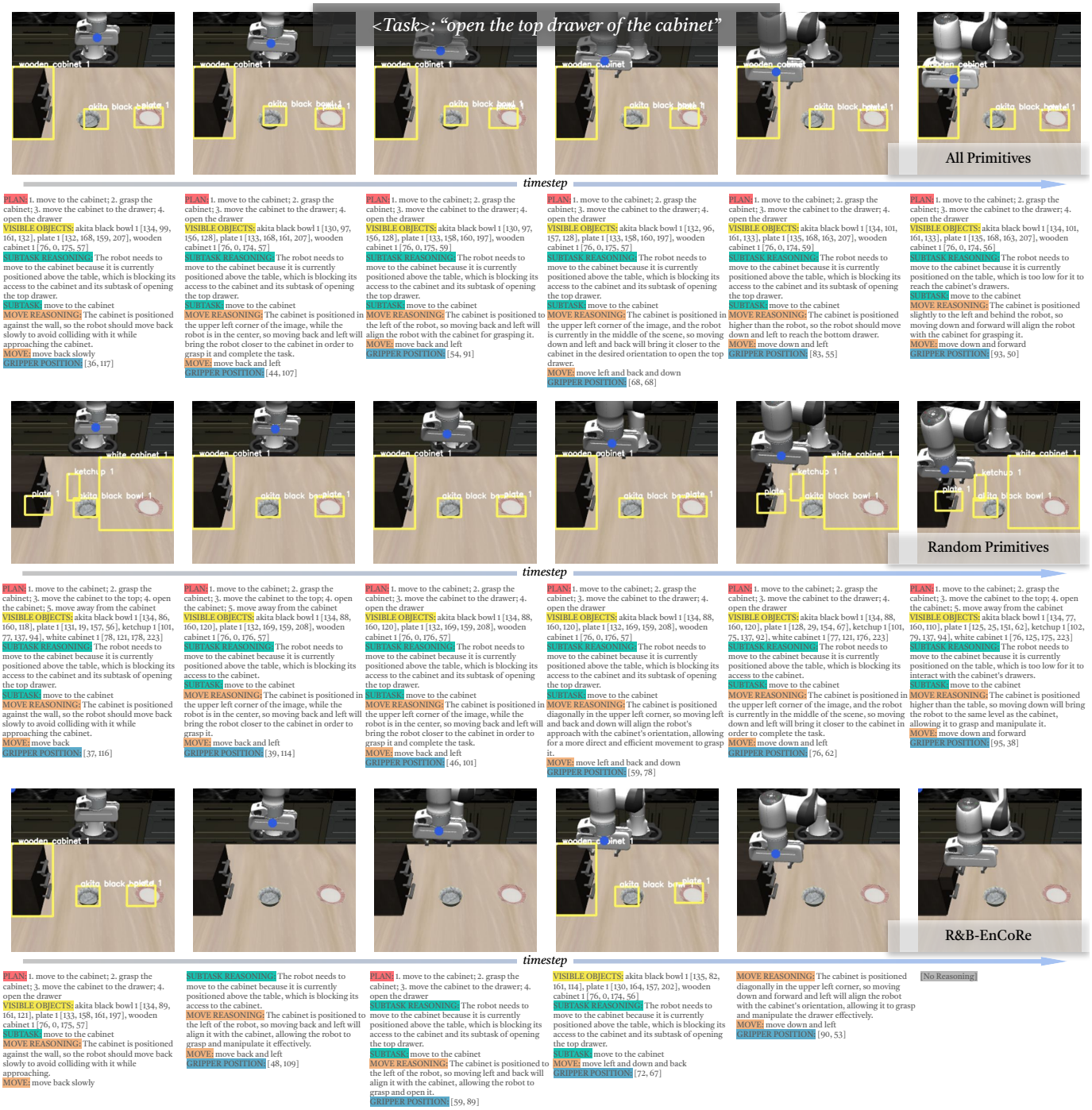


Fig. 21: Reasoning Traces with wider set of primitives (Section V-A-Q2) from the reasoning VLAs for LIBERO-90 across episode. Notice how R&B-EnCoRe reasons less frequently about visible objects compared with the other two models.

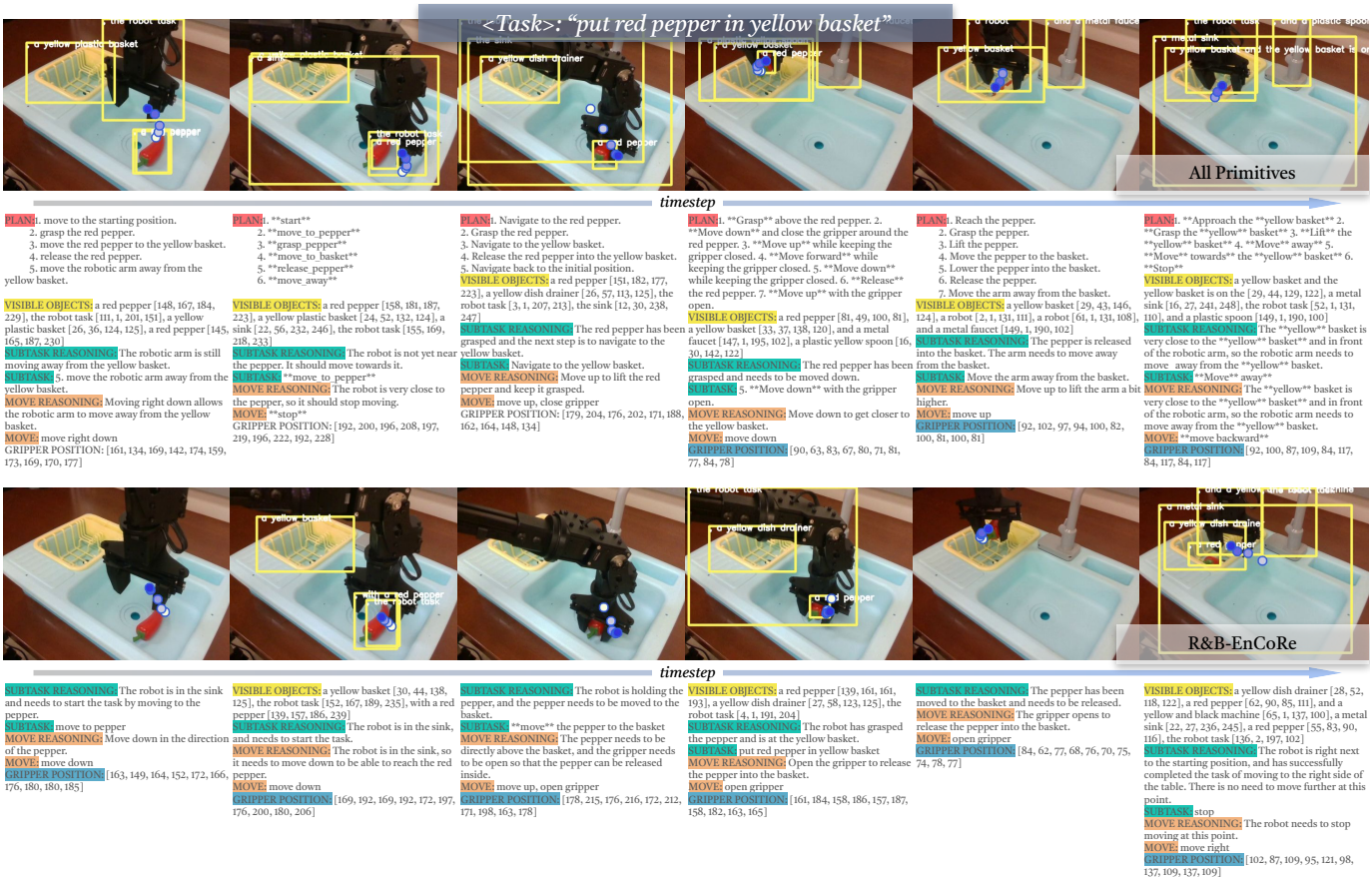


Fig. 22: Reasoning Traces (Section V-B-Q2) from the reasoning VLAs in Bridgev2 setup on WidowX hardware with Test-Time Reasoning enabled across episode.

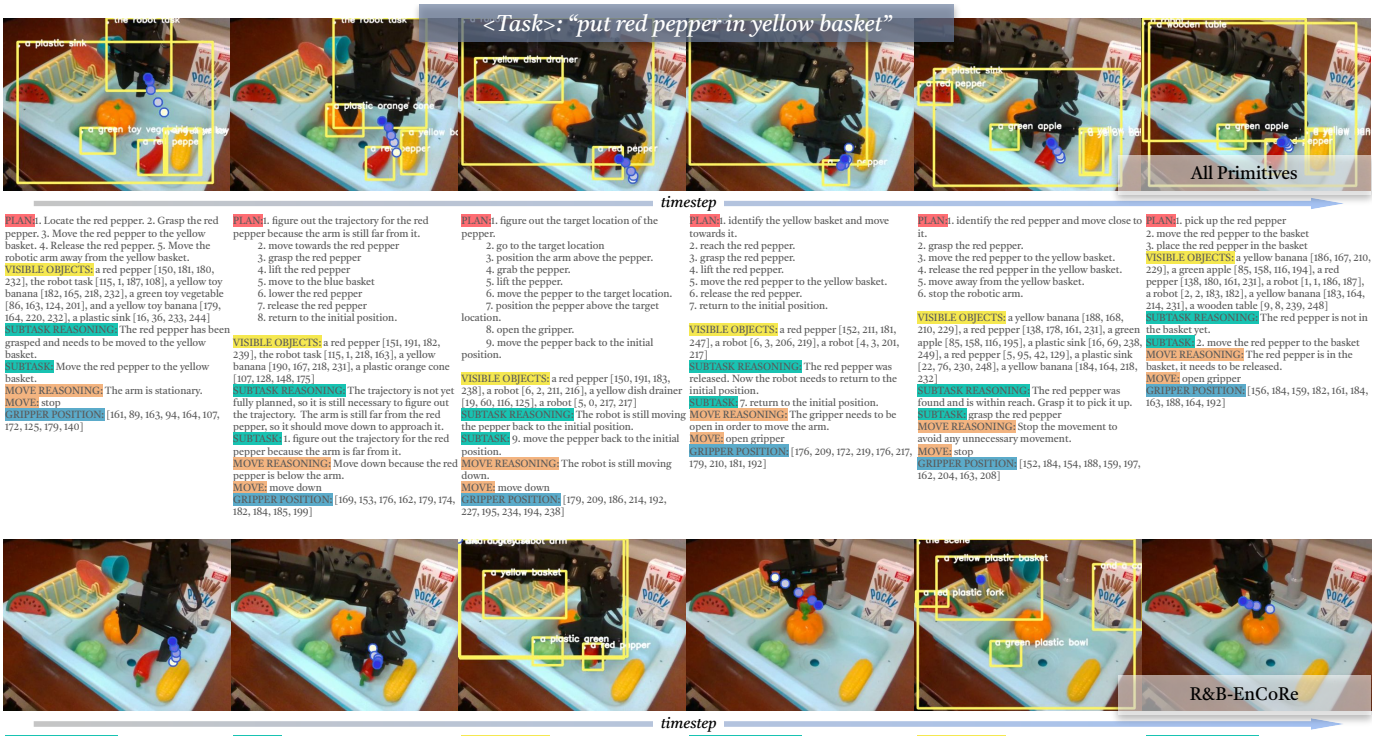


Fig. 23: Reasoning Traces (Section V-B-Q2) from the reasoning VLAs in Bridgev2 setup on WidowX hardware with Test-Time Reasoning enabled across episode.

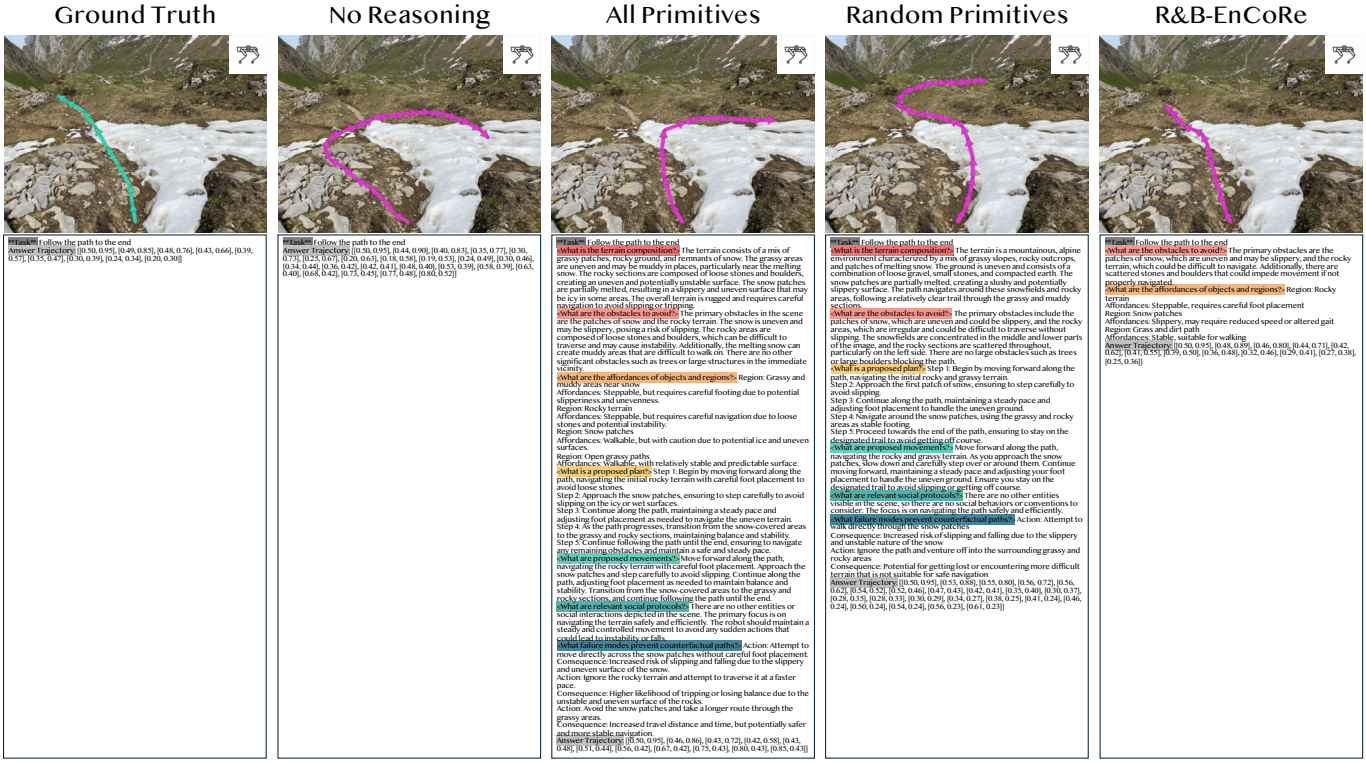


Fig. 24: Reasoning Traces for NaviTrace dataset with Quadruped embodiment (expanded version of Fig. 8).

Planned Trajectory:
[(0.07,4.87), (0.18,9.69), (0.33,14.43), (0.51,19.12), (0.71,23.76), (0.93,28.30)]

<What is the mission goal?> RIGHT

<What do you perceive in the scene?>

Future trajectories for specific objects:
Object type: car, object id: 1, future waypoint coordinates in 3s: [(9.53, -32.84), (9.55, -32.84), (9.55, -32.83), (9.57, -32.83), (9.58, -32.82), (9.58, -32.81)]
Object type: car, object id: 2, future waypoint coordinates in 3s: [(12.07, -40.02), (12.07, -40.01), (12.06, -40.01), (12.06, -40.00), (12.06, -39.99), (12.05, -39.98)]
Object type: car, object id: 3, future waypoint coordinates in 3s: [(10.90, -27.42), (10.91, -27.41), (10.91, -27.40), (10.93, -27.39), (10.94, -27.37), (10.94, -27.35)]

Distance to both sides of road shoulders of current ego-vehicle location:
Current ego-vehicle's distance to left shoulder is 1.0m and right shoulder is 2.0m

<What is a similar past experience?>
Most similar driving experience from memory with confidence score: 0.47:
The planned trajectory in this experience for your reference: [(0.27,4.98), (0.84,9.87), (1.74,14.66), (2.95,19.42), (4.38,24.25), (5.99,29.12)]
<What are the collidable objects?>
- Notable Objects: None
Potential Effects: None<What is the driving plan?> CHANGE LANE TO RIGHT WITH A CONSTANT SPEED
<What are the notable objects?>
- Notable Objects: None
Potential Effects: None

Planned Trajectory:
[(0.07,4.82), (0.22,9.62), (0.46,14.38), (0.84,19.13), (1.31,23.85), (1.91,28.53)]

<What is the mission goal?> FORWARD

<What do you perceive in the scene?>

Distance to both sides of road shoulders of current ego-vehicle location:
Current ego-vehicle's distance to left shoulder is 1.0m and right shoulder is 4.0m

<What is a similar past experience?>
Most similar driving experience from memory with confidence score: 0.72:
The planned trajectory in this experience for your reference: [(0.12,4.90), (0.39,9.79), (0.70,14.28), (1.06,19.27), (1.46,24.21), (1.85,29.05)]
<What are the collidable objects?>
- Notable Objects: None
Potential Effects: None<What is the driving plan?> MOVE FORWARD WITH A CONSTANT SPEED
<What are the notable objects?>
- Notable Objects: None
Potential Effects: None

Planned Trajectory:
[(0.06,4.83), (0.21,9.15), (0.51,14.36), (0.96,19.10), (1.54,23.82), (2.22,28.48)]

<What is the mission goal?> LEFT

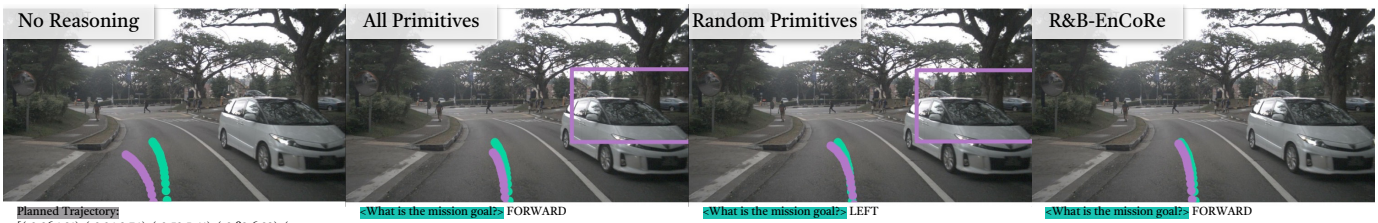
<What do you perceive in the scene?>

Future trajectories for specific objects:
Object type: truck, object id: 1, future waypoint coordinates in 3s: [(16.85, -2.72), (16.85, -2.71), (16.85, -2.71), (16.84, -2.71), (16.84, -2.71), (16.83, -2.71)]
Object type: car, object id: 2, future waypoint coordinates in 3s: [(16.77, 2.81), (16.76, 2.82), (16.75, 2.84), (16.75, 2.85), (16.75, 2.86), (16.74, 2.87)]

Distance to both sides of road shoulders of current ego-vehicle location:
Current ego-vehicle's distance to left shoulder is 1.5m and right shoulder is 4.0m

<What are the collidable objects?>
- Notable Objects: None
Potential Effects: None<What is the driving plan?> CHANE LANE TO LEFT WITH A DECELERATION

Planned Trajectory:
[(0.07,4.85), (0.15,9.58), (0.10,14.12), (-0.12,18.40), (-0.52,22.46), (-1.21,26.24)]



Planned Trajectory:
[(-0.06,1.91), (-0.24,3.74), (-0.52,5.41), (-0.89,6.93), (-1.30,8.23), (-1.74,9.37)]

<What is the mission goal?> FORWARD

<What do you perceive in the scene?>
Front object detections:
Front object detected, object type: car, object id: 0, position: (2.78, 8.66), size: (1.86, 4.45)

Future trajectories for specific objects:
Object type: car, object id: 0, future waypoint coordinates in 3s: [(2.98, 6.32), (3.12, 3.70), (3.18, 0.73), (3.18, -2.37), (3.16, -5.59), (3.02, -8.99)]

Distance to both sides of road shoulders of current ego-vehicle location:
Current ego-vehicle's distance to left shoulder is 1.5m and right shoulder is 4.0m

<What is a similar past experience?>
Most similar driving experience from memory with confidence score: 0.72:
The planned trajectory in this experience for your reference: [(-0.03,1.91), (-0.13,3.85), (-0.26,5.74), (-0.41,7.54), (-0.58,9.18), (-0.76,10.61)]
<What are the collidable objects?>
- Notable Objects: car at (2.78,8.66), moving to (2.98,6.32) at 0.5 second
Potential Effects: within the safe zone of the ego-vehicle at 0.5 second<What is the driving plan?> MOVE FORWARD WITH A_TUNING
<What are the notable objects?>
- Notable Objects: car at (2.78,8.66), moving to (2.98,6.32) at 0.5 second
Potential Effects: within the safe zone of the ego-vehicle at 0.5 second

Planned Trajectory:
[(-0.06,2.05), (-0.18,3.95), (-0.34,5.75), (-0.54,7.40), (-0.76,8.84), (-1.01,10.10)]

<What is the mission goal?> LEFT

<What do you perceive in the scene?>
Front object detections:
Front object detected, object type: car, object id: 2, position: (2.82, 8.69), size: (1.82, 4.27)

Future trajectories for specific objects:
Object type: car, object id: 2, future waypoint coordinates in 3s: [(2.77, 6.60), (2.64, 4.50), (2.44, 2.37), (2.17, 0.23), (1.85, -1.91), (1.44, -4.03)]

Distance to both sides of road shoulders of current ego-vehicle location:
Current ego-vehicle's distance to left shoulder is 1.5m and right shoulder is 3.5m

<What is a similar past experience?>
Most similar driving experience from memory with confidence score: 0.41:
The planned trajectory in this experience for your reference: [(-0.09,1.93), (-0.29,3.90), (-0.61,5.84), (-1.03,7.72), (-1.59,9.58), (-2.19,11.41)]
<What are the collidable objects?>
- Notable Objects: None
Potential Effects: None<What is the driving plan?> CHANE LANE TO LEFT WITH AN ACCELERATION

Planned Trajectory:
[(-0.02,1.91), (-0.11,3.92), (-0.29,6.01), (-0.58,8.22), (-1.00,10.47), (-1.53,12.61)]

<What is the mission goal?> FORWARD

<What is a similar past experience?>
Most similar driving experience from memory with confidence score: 0.77:
The planned trajectory in this experience for your reference: [(-0.02,1.94), (-0.08,3.90), (-0.18,5.85), (-0.34,7.80), (-0.53,9.74), (-0.78,11.62)]
<What are the collidable objects?>
- Notable Objects: None
Potential Effects: None<What is the driving plan?> MOVE FORWARD WITH AN ACCELERATION
<What are the notable objects?>
- Notable Objects: None
Potential Effects: None

Planned Trajectory:
[(-0.05,1.99), (-0.18,4.11), (-0.38,6.38), (-0.64,8.71), (-0.99,11.09), (-1.40,13.59)]

Fig. 25: Reasoning Traces (Section V-D-Q7) from the driving VLAs. We visualize predictions across models on two samples from the nuScenes dataset. Observe that using R&B-EnCoRe improves performance and yields concise reasoning traces that is more informative than not reasoning at all. Reasoning types are colored for visualization purposes.

Planned Trajectory:
[(0.02,3.70), (0.03,7.24), (0.04,10.61), (0.05,13.79), (0.06,16.79), (0.07,19.61)]

<What is the mission goal?> FORWARD

<What do you perceive in the scene?>
Front object detections:
Front object detected, object type: car, object id: 0, position: (0.59, 15.46), size: (1.93, 4.55)
Front object detected, object type: bus, object id: 6, position: (4.40, 18.95), size: (2.63, 8.31)

Future trajectories for specific objects:
Object type: car, object id: 0, future waypoint coordinates in 3s: [(0.67, 18.36), (0.71, 21.03), (0.78, 23.53), (0.82, 25.82), (0.81, 27.91), (0.87, 30.01)]
Object type: bus, object id: 6, future waypoint coordinates in 3s: [(4.45, 22.80), (4.48, 26.67), (4.54, 30.63), (4.57, 34.59), (4.56, 38.60), (4.57, 42.59)]

Distance to both sides of road shoulders of current ego-vehicle location:
Current ego-vehicle's distance to left shoulder is 2.0m and right shoulder is 4.5m

<What is a similar past experience?>
Most similar driving experience from memory with confidence score: 0.87:
The planned trajectory in this experience for your reference:
[(0.04,3.62), (0.07,7.06), (0.10,10.36), (0.12,13.43), (0.14,16.37), (0.18,19.16)]
<What are the collidable objects?>
- Notable Objects: None
Potential Effects: None<What is the driving plan?> MOVE FORWARD WITH A CONSTANT SPEED
<What are the notable objects?>
- Notable Objects: car at (0.59,15.46), moving to (0.78,23.53) at 1.5 second
Potential Effects: within the safe zone of the ego-vehicle at 1.5 second

Planned Trajectory:
[(0.02,3.75), (0.03,7.70), (0.03,11.65), (0.04,15.76), (0.02,19.86), (0.02,24.03)]

<What is the mission goal?> FORWARD

<What do you perceive in the scene?>
Front object detections:
Front object detected, object type: car, object id: 0, position: (0.56, 15.95), size: (1.92, 4.53)
Front object detected, object type: bus, object id: 7, position: (4.36, 18.91), size: (2.81, 11.65)

Future trajectories for specific objects:
Object type: car, object id: 0, future waypoint coordinates in 3s: [(0.58, 19.45), (0.59, 22.80), (0.64, 26.09), (0.67, 29.27), (0.66, 32.36), (0.73, 35.40)]
Object type: bus, object id: 7, future waypoint coordinates in 3s: [(4.37, 18.91), (4.37, 18.91), (4.37, 18.91), (4.37, 18.91), (4.37, 18.91), (4.38, 18.91)]

Distance to both sides of road shoulders of current ego-vehicle location:
Current ego-vehicle's distance to left shoulder is 4.0m and right shoulder is 4.5m

<What is a similar past experience?>
Most similar driving experience from memory with confidence score: 0.89:
The planned trajectory in this experience for your reference:
[(0.03,3.52), (0.06,6.79), (0.09,9.76), (0.12,12.17), (0.13,14.56), (0.14,16.90)]
<What are the collidable objects?>
- Notable Objects: None
Potential Effects: None<What is the driving plan?> MOVE FORWARD WITH A CONSTANT SPEED
<What are the notable objects?>
- Notable Objects: car at (0.56,15.95), moving to (0.64,26.09) at 1.5 second
Potential Effects: within the safe zone of the ego-vehicle at 1.5 second

Planned Trajectory:
[(0.00,3.72), (0.01,7.31), (0.03,10.79), (0.06,14.13), (0.10,17.32), (0.16,20.30)]

<What is the mission goal?> FORWARD

<What do you perceive in the scene?>
Front object detections:
Front object detected, object type: car, object id: 4, position: (0.56, 15.90), size: (1.90, 4.51)

Future trajectories for specific objects:
Object type: car, object id: 4, future waypoint coordinates in 3s: [(0.59, 19.30), (0.60, 22.89), (0.65, 26.67), (0.67, 30.56), (0.61, 34.62), (0.64, 38.77)]

Distance to both sides of road shoulders of current ego-vehicle location:
Current ego-vehicle's distance to left shoulder is 4.5m and right shoulder is 4.5m

<What is the driving plan?> MOVE FORWARD WITH A CONSTANT SPEED

Planned Trajectory:
[(0.03,3.65), (0.04,7.11), (0.05,10.35), (0.06,13.49), (0.08,16.32), (0.09,18.89)]

Planned Trajectory:
[(0.05,3.60), (0.11,7.49), (0.16,11.60), (0.21,15.90), (0.25,20.34), (0.29,24.90)]

<What is the mission goal?> FORWARD

<What do you perceive in the scene?>
None.<What is a similar past experience?>
Most similar driving experience from memory with confidence score: 0.79:
The planned trajectory in this experience for your reference:
[(0.08,3.66), (0.13,7.52), (0.20,11.56), (0.26,15.68), (0.30,19.95), (0.34,24.34)]
<What are the collidable objects?>
- Notable Objects: None
Potential Effects: None<What is the driving plan?> MOVE FORWARD WITH AN ACCELERATION
<What are the notable objects?>
- Notable Objects: None
Potential Effects: None

Planned Trajectory:
[(0.04,3.60), (0.10,7.09), (0.19,10.47), (0.27,13.77), (0.31,17.06), (0.33,20.30)]

<What is the mission goal?> FORWARD

<What do you perceive in the scene?>
None.<What is a similar past experience?>
Most similar driving experience from memory with confidence score: 0.64:
The planned trajectory in this experience for your reference:
[(0.01,3.27), (0.04,6.78), (0.06,10.58), (0.03,14.58), (0.01,18.72), (-0.02,23.05)]
<What are the collidable objects?>
- Notable Objects: None
Potential Effects: None<What is the driving plan?> MOVE FORWARD WITH AN ACCELERATION
<What are the notable objects?>
- Notable Objects: None
Potential Effects: None

Planned Trajectory:
[(0.05,3.60), (0.09,7.44), (0.12,11.46), (0.15,15.61), (0.20,19.89), (0.25,24.24)]

<What is the mission goal?> FORWARD

<What do you perceive in the scene?>
Front object detections:
Front object detected, object type: pedestrian, object id: 10, position: (-4.59, 13.85), size: (0.63, 0.67)

Future trajectories for specific objects:
Object type: pedestrian, object id: 10, future waypoint coordinates in 3s: [(-4.67, 13.31), (-4.73, 12.79), (-4.80, 12.27), (-4.87, 11.75), (-4.95, 11.24), (-4.98, 10.72)]

Distance to both sides of road shoulders of selected locations:
Location (-4.59, 13.85) distance to left shoulder is 0.5m and right shoulder is 0.0m

<What are the collidable objects?>
- Notable Objects: None
Potential Effects: None<What is the driving plan?> MOVE FORWARD WITH AN ACCELERATION

Planned Trajectory:
[(0.03,3.63), (0.09,7.52), (0.17,11.74), (0.26,16.20), (0.37,20.72), (0.48,25.39)]

Fig. 26: Reasoning Traces (Section V-D-Q7) from the driving VLAs. We visualize results on two more samples from the nuScenes dataset.