HIERARCHICAL VARIATIONAL CHAIN OF THOUGHTS FOR VISION AND LANGUAGE REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Chain-of-thought (CoT) vectors offer a compact way to steer reasoning in frozen large models, but representing reasoning with a single deterministic vector shared across inputs limits adaptation to individual problems. We propose Hierarchical Variational Chain of Thoughts (HIERACOT) by modeling the CoT trace as a distribution over latent states. A global state is defined to capture reasoning patterns shared across tasks, while a local state adapts these patterns to the current input. This separation is encouraged by an input-independent global prior and a local prior conditioned on both the input and the global state. During training, posterior networks use CoT traces and target answers as auxiliary information, and a variational objective aligns these posteriors with the corresponding priors. At inference, states sampled from the priors are projected into additive hidden-state shifts, enabling answer generation without explicit rationales or updates to the backbone parameters. Because the intervention operates in hidden space, the same formulation supports both vision-language and language-only reasoning. Experiments across four frozen backbones show that HIERACOT outperforms direct prompting, CoT prompting, and deterministic CoT-vector baselines, improving average accuracy over CoT prompting by 2.32–5.15 percentage points. Ablations support the benefits of the global–local hierarchy, while cross-domain evaluations provide evidence that the learned states transfer beyond their training domains.

1 INTRODUCTION

Reasoning is a central capability of large models, enabling them to solve complex tasks that require multi-step inference, abstraction, and decision making (Zhao et al., 2023; Huang & Chang, 2023; Jaech et al., 2024; Guo et al., 2025). A dominant approach to eliciting reasoning is to make the intermediate process explicit. Chain-of-thought prompting asks models to generate natural-language rationales before producing the final answer (Wei et al., 2022; Gao et al., 2023; Xu et al., 2025), and has led to strong improvements on mathematical, commonsense, and symbolic reasoning tasks (Wang et al., 2026b; Yu et al., 2024; Yax et al., 2024). However, explicit reasoning also introduces additional token cost, exposes intermediate traces that may be verbose or unreliable, and ties the reasoning process to a particular textual form.

To reduce the reliance on visible rationales, recent work has explored implicit or latent reasoning, where intermediate computation is moved from generated text into hidden states or continuous representations (M-A-P, 2025; Hao et al., 2024; Shen et al., 2025; Wei et al., 2026). These methods retain the intuition that reasoning unfolds through a sequence of intermediate steps, but replace textual steps with continuous latent states. In parallel, another line of work suggests a more compact view: reasoning-related behavior can be transferred or steered through hidden-state vectors (Ilharco et al., 2023; Ortiz-Jimenez et al., 2023; Li et al., 2026b; Cao et al., 2026). In particular, CoT Vectors show that chain-of-thought reasoning can be compressed into additive directions in the activation space of a frozen model, providing an efficient alternative to generating explicit rationales.

Despite this progress, existing CoT-vector method Li et al. (2026b) typically represent reasoning with a single deterministic vector, often shared across a task or support set. This design is efficient, but it imposes a strong bottleneck. A single vector must simultaneously encode reasoning tendencies that are broadly useful across problems and adjustments that are specific to each input. As a result, it cannot naturally distinguish what should be shared across reasoning problems from what should vary

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

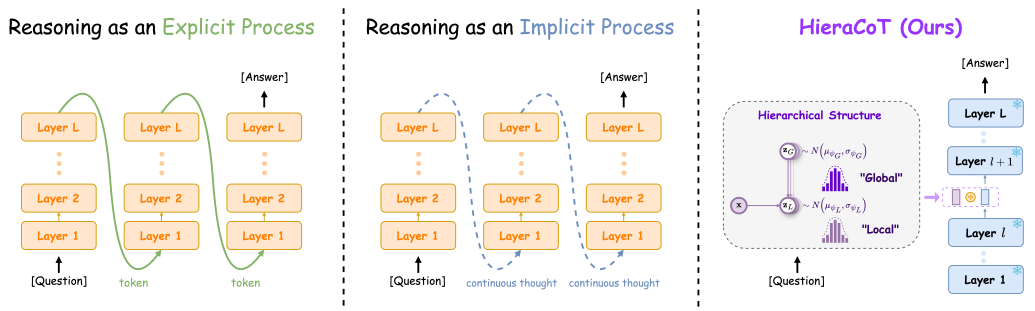


Figure 1: **From reasoning processes to Chain-of-Thought states.** *Left*: explicit reasoning generates intermediate natural-language steps before the final answer. *Middle*: implicit reasoning moves the sequential process into continuous hidden states. *Right* (**HIERACOT**): we extend CoT vectors into hierarchical variational Chain-of-Thought states. A global state captures shared reasoning tendencies, while a local state adapts them to the current input. The sampled state is injected into the hidden states of a frozen model as an additive shift, without generating rationales or unfolding iterative latent steps.

with the current question, image, or answer space. This limitation becomes especially important in vision-language reasoning, where different inputs may require distinct forms of visual grounding, textual inference, or commonsense abstraction.

In this work, we introduce Hierarchical Variational Chain of Thought (HIERACOT) for vision and language reasoning. Instead of using a single fixed CoT vector, HIERACOT models the CoT trace by distributions over hierarchical latent states. The method keeps the pretrained model frozen and learns lightweight inference networks around it. During training, a posterior network observes auxiliary information from the full training example, including the input, chain-of-thought trace, and target answer. A prior network observes only the input and is trained, through a variational objective, to approximate the posterior distribution. At inference time, only the prior is used: it samples a CoT state from the input, and the sampled state is injected into the hidden states of the frozen model as a simple additive shift.

The key design of HIERACOT is a two-level reasoning hierarchy: a global state is defined to capture reasoning tendencies shared across tasks, while the local state adapts these tendencies to the current input. We implement this separation through a structured prior: the global prior is input-independent, encouraging it to represent reusable reasoning information, whereas the local prior is conditioned on the input and the global state, allowing it to encode instance-specific variation. This hierarchy turns CoT vectors from a single deterministic direction into a global-local distribution over reasoning states. Since the state is applied in hidden space rather than to modality-specific tokens, visual regions, or generated rationales, the same formulation naturally applies to both vision-language and language-only reasoning.

We evaluate HIERACOT on diverse reasoning tasks across multiple modalities. For vision-language reasoning, we use Qwen2.5-VL-7B and InternVL2.5-8B as frozen models and evaluate on ScienceQA, Visual-CoT-GQA, TextVQA, and A-OKVQA. For language reasoning, we use Qwen2.5-Math-7B and Llama3.1-8B-Instruct and evaluate on GSM8K, MATH-E, MATH-H, and MMLU-Pro. Across both settings, HIERACOT consistently improves over direct prompting, chain-of-thought prompting, and deterministic CoT-vector baselines. The hierarchical variant further outperforms the single-state variant on average. Additional analyses show that the global and local states exhibit the intended functional separation, and cross-domain experiments indicate that the learned states capture transferable reasoning patterns rather than merely memorizing domain-specific answer behavior.

2 RELATED WORK

Explicit and Implicit Reasoning. Explicit reasoning has become a standard way to improve the performance of large language models on complex tasks (Wei et al., 2022; Kojima et al., 2022). Chain-of-thought prompting elicits intermediate natural-language steps before the final answer, and later methods further improve this paradigm by decomposing problems, aggregating multiple rea-

soning paths, or searching over candidate thoughts (Zhou et al., 2023; Xia et al., 2025; Wang et al., 2023; Yao et al., 2023). Other work strengthens explicit reasoning through supervised rationales, bootstrapped reasoning traces, process supervision, or hierarchical planning and execution (Zelikman et al., 2022; Phan et al., 2023; Lightman et al., 2023; Wang et al., 2025a;b; Huang et al., 2026). These methods demonstrate the value of reasoning traces, but they require models to generate additional tokens and make reasoning explicitly tied to natural-language rationales. A parallel line of work moves reasoning from visible text into hidden states or continuous representations. Some methods replace explicit token generation with continuous thought steps (M-A-P, 2025; Hao et al., 2024; Zhang et al., 2025b; Geiping et al., 2025), while others compress or internalize chain-of-thought traces into latent representations through distillation or self-distillation (Deng et al., 2024; Shen et al., 2025; Tan et al., 2025; Wei et al., 2026; Zhu et al., 2025). Recent variational approaches further treat reasoning traces or latent reasoning states as latent variables (Zhou et al., 2026; Wang et al., 2026a), with extensions to multimodal settings (Li et al., 2026a; Pham & Ngo, 2025; Zhang et al., 2026). These studies reduce the dependence on explicit rationales, but many of them still model reasoning as a sequence of textual or continuous intermediate steps. Our work follows a different motivation: instead of unfolding additional reasoning steps, we build on CoT-vector methods and learn a structured hidden-state shift that can be injected into a frozen model.

Reasoning in Vision-Language Models. Vision-language reasoning has also benefited from explicit intermediate reasoning. Early work trains models to generate textual explanations or multimodal rationales for science question answering and visual question answering (Lu et al., 2022; Zhang et al., 2023; Wang et al., 2024a). More recent work makes the intermediate reasoning process more grounded or structured, for example by incorporating compositional rationales, visually grounded reasoning steps, or region-level evidence (Mitra et al., 2024; Li et al., 2024; Wu et al., 2025). Another group of methods introduces tool programs, visual manipulations, perception tokens, or long-chain visual reasoning supervision (Hu et al., 2024; Qi et al., 2025; Bigverdi et al., 2025; Dong et al., 2024). These methods improve multimodal reasoning by adding modality-specific intermediate objects, such as textual rationales, visual regions, perception tokens, or program traces. In contrast, our method does not introduce additional visual modules or generate multimodal rationales at inference time. It operates directly in the hidden space of a frozen vision-language model, which allows the same formulation to be used for both vision-language and language reasoning.

CoT Vectors and Hidden-State Interventions. A related direction studies whether model behavior can be modified through compact directions in parameter space or activation space (Ilharco et al., 2023; Todd et al., 2024; Zhang et al., 2025a; Yang et al., 2024b). Task-vector methods show that parameter-space directions can encode transferable behavioral changes, while hidden-state steering methods show that activation-space directions can control model behavior without updating the full model (Ortiz-Jimenez et al., 2023; Yadav et al., 2023; Cheng et al., 2025; Hendel et al., 2023). This idea has recently been applied to reasoning. CoT Vectors show that reasoning mechanisms induced by chain-of-thought supervision can be transferred and probed through compact additive shifts in hidden space (Li et al., 2026b). Thought injection and related activation-steering methods similarly suggest that reasoning-related behavior can be influenced by adding learned or extracted vectors to internal representations (Cao et al., 2026; He et al., 2026). This line is closest to ours, but existing CoT-vector methods usually rely on a single deterministic vector shared across a task or behavior. We instead learn HIERACOT, a hierarchical variational distribution over Chain-of-Thought states, where a global state captures shared reasoning tendencies and a local state provides input-specific adaptation. This preserves the efficiency of hidden-state intervention while making CoT-induced reasoning states probabilistic, input-conditioned, and structured.

3 METHODOLOGY

3.1 PROBLEM SETUP

Consider a supervised reasoning setting with a training set $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{c}_i, \mathbf{y}_i)\}_{i=1}^{|\mathcal{D}|}$. Here, $\mathbf{x}_i$ denotes the input problem. In the vision-language setting, $\mathbf{x}_i$ includes an image and a question. The target answer sequence is $\mathbf{y}_i = (y_{i,1}, \ldots, y_{i,T_i})$. The variable $\mathbf{c}_i$ denotes the chain-of-thought trace observed only during training and serves only as auxiliary information for the inference networks introduced in the following. Our goal is to learn the conditional distribution $p(\mathbf{y}|\mathbf{x})$. At test time, only the input $\mathbf{x}$ is observed.

We build on a pre-trained autoregressive model with frozen parameters θ, which is either a language model or a vision-language model. Let $\mathbf{H}_i^{(l)} \in \mathbb{R}^{L_i \times d}$ denote the hidden representation at layer l for input i, where L_i is the sequence length and d is the hidden dimension. Inside the frozen model, the Chain-of-Thought state is realized as an additive shift at an internal layer:

$$\tilde{\mathbf{H}}_i^{(l)} = \mathbf{H}_i^{(l)} + \mathbf{1}_{L_i}\mathbf{s}^\top, \tag{1}$$

where $\mathbf{s} \in \mathbb{R}^d$ is the injected state vector and $\mathbf{1}_{L_i}$ denotes the vector of ones of length L_i. This shift conditions the frozen model on the Chain-of-Thought state during answer generation. Because the shift operates in the hidden space rather than on input tokens or visual regions, the same formulation applies to both language and vision-language models.

The remaining question is how to learn this state. We first formalize it as a single latent variable trained via variational inference, and then extend the formulation to a two-level hierarchy.

3.2 A Variational Chain-of-Thought Reasoning

Unlike a single deterministic vector shared across all inputs, we formalize the Chain-of-Thought state as a latent variable $\mathbf{z} \in \mathbb{R}^{d_z}$ conditioned on the input $\mathbf{x}$, and learn its distribution through variational inference. This input-conditioned formulation lets the state adapt to each problem, recovering input-specific reasoning behavior that a deterministic vector cannot represent.

The answer distribution is obtained by marginalizing over $\mathbf{z}$:

$$p(\mathbf{y}|\mathbf{x}) = \int p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{z})\, p_\psi(\mathbf{z}|\mathbf{x})\, \mathrm{d}\mathbf{z}, \tag{2}$$

where the prior network $p_\psi(\mathbf{z}|\mathbf{x})$ specifies the distribution of $\mathbf{z}$ given the input, and $p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{z})$ is the conditional distribution over $\mathbf{y}$ produced by the frozen model with $\mathbf{z}$ injected at layer l via Equation 1. The injected state takes the form $\mathbf{s} = \mathbf{W}_z\mathbf{z}$, where $\mathbf{W}_z \in \mathbb{R}^{d \times d_z}$ is a learned projection from the latent space to the hidden space.

Since the true posterior $p(\mathbf{z}|\mathbf{x}, \mathbf{y})$ is intractable, we introduce a posterior network $q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{r})$ to approximate it, where $\mathbf{r}$ denotes auxiliary information available only during training. The posterior network has access to $\mathbf{r}$ in addition to $\mathbf{x}$, while the prior network conditions only on $\mathbf{x}$. This asymmetry lets the posterior incorporate richer training information than the prior.

The auxiliary information $\mathbf{r}$ is realized as a feature vector extracted from the full training sample:

$$\mathbf{r} = \mathcal{R}_\theta^{(l)}(\mathbf{x}, \mathbf{c}, \mathbf{y}). \tag{3}$$

The operator $\mathcal{R}_\theta^{(l)}$ pools the hidden states at layer l when the triple is fed in. The resulting $\mathbf{r}$ therefore carries information from both the chain-of-thought trace $\mathbf{c}$ and the target $\mathbf{y}$. We examine the separate contributions of $\mathbf{c}$ and $\mathbf{y}$ to the posterior input in Appendix B.4.

We then obtain the following evidence lower bound (ELBO):

$$\log p(\mathbf{y}|\mathbf{x}) \geq \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{r})}\left[\log p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{z})\right] - \mathbb{D}_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{r}) \parallel p_\psi(\mathbf{z}|\mathbf{x}))\,. \tag{4}$$

The likelihood term encourages $\mathbf{z}$ to capture information useful for predicting the answer, while the KL term aligns the prior with the posterior. At test time, $\mathbf{z}$ is sampled from the prior $p_\psi(\mathbf{z}|\mathbf{x})$, which depends only on the input. The trained model therefore does not require auxiliary information at inference time.

3.3 Hierarchical Chain-of-Thought Reasoning

The single-state formulation in Section 3.2 represents the Chain-of-Thought state with one latent variable, but a single variable mixes two kinds of information together. Some patterns generalize

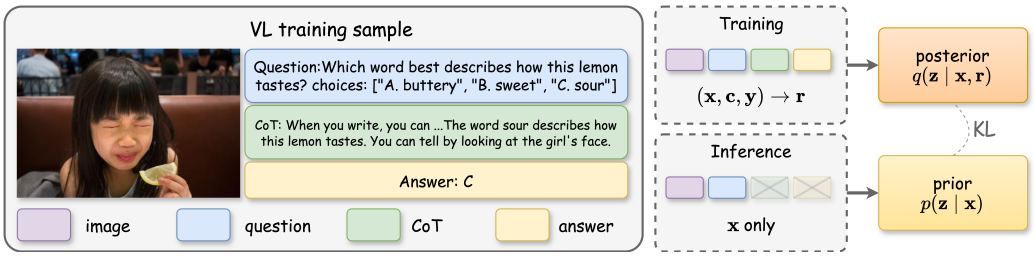


Figure 2: **Information flow during training and inference.** The figure shows the single-state case for clarity. The posterior network is conditioned on the auxiliary information $\mathbf{r}$ derived from the full training sample, while the prior network is conditioned only on the input $\mathbf{x}$.

across problems, while others depend on the current input. We introduce a two-level hierarchy with a global state $\mathbf{g}$ and a local state $\mathbf{z}$ to handle these two kinds separately.

The joint distribution over the answer and the two latent variables factorizes as:

$$p(\mathbf{y}, \mathbf{g}, \mathbf{z}|\mathbf{x}) = p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{g}, \mathbf{z})\, p_{\psi_g}(\mathbf{g})\, p_{\psi_z}(\mathbf{z}|\mathbf{x}, \mathbf{g}), \tag{5}$$

where $p_{\psi_g}(\mathbf{g})$ is the global prior, which does not depend on $\mathbf{x}$, and $p_{\psi_z}(\mathbf{z}|\mathbf{x}, \mathbf{g})$ is the local prior.

Mirroring the prior, we factorize the variational posterior as:

$$q_\phi(\mathbf{g}, \mathbf{z}|\mathbf{x}, \mathbf{r}) = q_{\phi_g}(\mathbf{g}|\mathbf{x}, \mathbf{r})\, q_{\phi_z}(\mathbf{z}|\mathbf{x}, \mathbf{r}, \mathbf{g}), \tag{6}$$

where $\mathbf{r}$ is the auxiliary information defined in equation 3. Both posteriors have access to $\mathbf{r}$ during training, while the corresponding priors do not.

Following the same derivation, we obtain the hierarchical evidence lower bound:

$$\begin{aligned} \log p(\mathbf{y}|\mathbf{x}) \geq\ & \mathbb{E}_{q_\phi(\mathbf{g},\mathbf{z}|\mathbf{x},\mathbf{r})} \left[\log p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{g}, \mathbf{z})\right] \\ & - \mathbb{D}_{\mathrm{KL}} \left(q_{\phi_g}(\mathbf{g}|\mathbf{x}, \mathbf{r}) \parallel p_{\psi_g}(\mathbf{g})\right) \\ & - \mathbb{E}_{q_{\phi_g}(\mathbf{g}|\mathbf{x},\mathbf{r})} \mathbb{D}_{\mathrm{KL}} \left(q_{\phi_z}(\mathbf{z}|\mathbf{x}, \mathbf{r}, \mathbf{g}) \parallel p_{\psi_z}(\mathbf{z}|\mathbf{x}, \mathbf{g})\right). \end{aligned} \tag{7}$$

The likelihood term encourages $\mathbf{g}$ and $\mathbf{z}$ to jointly capture information useful for predicting the answer. The two KL terms align each posterior with its corresponding prior. Because the global prior does not depend on the input while the local prior does, the global state $\mathbf{g}$ is pushed toward patterns that generalize across inputs, while the local state $\mathbf{z}$ captures input-specific variation. This separation arises from the prior structure alone, without additional supervision.

At inference time, only the priors are used. The global state $\mathbf{g}$ is sampled from $p_{\psi_g}(\mathbf{g})$, and the local state $\mathbf{z}$ is sampled from $p_{\psi_z}(\mathbf{z}|\mathbf{x}, \mathbf{g})$. The two states are projected to the hidden space and injected at layer l as:

$$\tilde{\mathbf{H}}_i^{(l)} = \mathbf{H}_i^{(l)} + \mathbf{1}_{L_i} \left(\mathbf{W}_g \mathbf{g} + \mathbf{W}_z \mathbf{z}\right)^\top, \tag{8}$$

where $\mathbf{W}_g \in \mathbb{R}^{d \times d_g}$ and $\mathbf{W}_z \in \mathbb{R}^{d \times d_z}$ are learned projections from the global and local latent spaces to the hidden space.

During training, $\mathbf{g}$ and $\mathbf{z}$ are drawn from the posteriors via the reparameterization trick so that gradients can flow through the sampling step. The two posteriors, the two priors, and the projections $\mathbf{W}_g, \mathbf{W}_z$ are jointly optimized to maximize Equation 7, while θ remains frozen. Figure 2 illustrates the information flow during training and inference using a vision-language example. Further implementation details are provided in Appendix B.

Table 1: **Main results on vision-language reasoning benchmarks.** The best and second-best results are shown in **bold** and underline, respectively.

Model	Method	ScienceQA	GQA	TextVQA	A-OKVQA	Avg.
Qwen2.5-VL-7B	Direct	75.31	41.60	69.00	80.70	66.65
	CoT (Kojima et al., 2022)	77.54	57.90	70.10	83.10	72.16
	Extracted (Li et al., 2026b)	77.65	58.00	71.20	84.00	72.71
	Learnable (Li et al., 2026b)	77.92	57.90	71.30	84.20	72.83
	HIERACOT-S	78.94	58.10	72.90	85.50	73.86
	HIERACOT-H	**79.12**	**58.40**	**74.50**	**85.90**	**74.48**
InternVL2.5-8B	Direct	85.63	40.00	69.50	83.50	69.66
	CoT (Kojima et al., 2022)	87.80	47.00	70.30	85.70	72.70
	Extracted (Li et al., 2026b)	88.97	48.20	72.10	85.90	73.79
	Learnable (Li et al., 2026b)	89.94	48.90	71.50	85.90	74.06
	HIERACOT-S	90.22	54.70	73.90	86.20	76.26
	HIERACOT-H	**90.51**	**55.10**	**75.20**	**86.40**	**76.80**

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

For vision-language reasoning, we use Qwen2.5-VL-7B (Wang et al., 2024b) and InternVL2.5-8B (Chen et al., 2024) as frozen models. We evaluate all methods on ScienceQA (Lu et al., 2022), Visual-CoT-GQA (GQA) (Shao et al., 2024), TextVQA (Singh et al., 2019), and A-OKVQA (Schwenk et al., 2022) using exact-match accuracy. We compare against four baselines: (1) *Direct*, which uses the frozen model without chain-of-thought prompting or hidden-state injection; (2) *CoT* (Kojima et al., 2022), which uses chain-of-thought prompting without injection; (3) *Extracted* (Li et al., 2026b), which computes a fixed CoT vector from the mean activation difference between teacher and student passes; and (4) *Learnable* (Li et al., 2026b), which optimizes a single CoT vector with a teacher-student objective. The latter two are deterministic CoT-vector baselines and are injected as additive hidden-state shifts under the same frozen-model setting as our method. We evaluate two variants of our approach: HIERACOT-S, a single variational CoT state without hierarchy, and HIERACOT-H, the hierarchical model with global and local CoT states. Additional details are provided in Appendix B.1.

For language reasoning, we use Qwen2.5-Math-7B (Yang et al., 2024a) and Llama3.1-8B-Instruct (Grattafiori et al., 2024) as frozen models. We evaluate all methods on GSM8K (Cobbe et al., 2021), MATH-E, MATH-H, and MMLU-Pro (MMLU-P) (Wang et al., 2024c), again using exact-match accuracy. MATH-E and MATH-H follow the difficulty split definitions of CoT Vectors (Li et al., 2026b). We evaluate the same baselines and method variants as in the vision-language setting using a shared evaluation pipeline. Appendix B.2 gives the benchmark details, and Appendix B.3 specifies the evaluation protocol. All experiments are conducted on a single NVIDIA H100 80GB GPU. Full implementation details and hyperparameter settings are provided in Appendix B.

4.2 RESULTS ON VISION-LANGUAGE REASONING

Table 1 reports the main results on vision-language reasoning. Across both Qwen2.5-VL-7B and InternVL2.5-8B, HIERACOT-H achieves the best average accuracy among all methods. Compared with CoT prompting, it improves the average score by 2.32 points on Qwen2.5-VL-7B and 4.10 points on InternVL2.5-8B. It also consistently outperforms the deterministic CoT-vector baselines, suggesting that a structured, input-conditioned CoT state is more effective than a single fixed hidden-state vector.

The improvements are consistent across benchmarks rather than being driven by one isolated task. On Qwen2.5-VL-7B, the most notable gain appears on TextVQA, where HIERACOT-H improves over CoT by 4.40 points. On InternVL2.5-8B, the gains are especially clear on Visual-CoT-GQA and TextVQA, indicating that the proposed states are useful for tasks that require stronger visual

Table 2: **Main results on language reasoning benchmarks.** The best and second-best results are shown in **bold** and underline, respectively.

Model	Method	GSM8K	MATH-E	MATH-H	MMLU-P	Avg.
Qwen2.5-Math-7B	Direct	87.20	61.70	30.10	25.00	51.00
	CoT (Kojima et al., 2022)	87.30	63.80	32.30	25.00	52.10
	Extracted (Li et al., 2026b)	87.50	68.10	32.40	25.50	53.38
	Learnable (Li et al., 2026b)	90.70	67.70	33.60	26.10	54.53
	HIERACOT-S	**91.30**	68.50	33.80	26.10	54.93
	HIERACOT-H	91.20	**69.40**	**35.50**	**32.90**	**57.25**
Llama3.1-8B-Instruct	Direct	84.20	62.50	32.00	29.30	52.00
	CoT (Kojima et al., 2022)	85.60	62.70	32.30	36.30	54.23
	Extracted (Li et al., 2026b)	86.70	63.20	32.50	36.80	54.80
	Learnable (Li et al., 2026b)	86.20	63.80	32.90	36.60	54.88
	HIERACOT-S	87.30	63.40	33.10	36.90	55.18
	HIERACOT-H	**88.00**	**65.90**	**34.30**	**38.40**	**56.65**

grounding and multimodal evidence integration. The method also improves on higher-accuracy datasets such as ScienceQA and A-OKVQA, although the margins are smaller when the baseline is already strong.

The hierarchical design further improves over the single-state variant. While HIERACOT-S already provides clear gains over CoT prompting and deterministic CoT vectors, HIERACOT-H achieves higher average accuracy on both models and obtains the best result on every vision-language benchmark. This supports the motivation for decomposing CoT-induced reasoning behavior into a shared global state and an input-specific local state.

4.3 RESULTS ON LANGUAGE REASONING

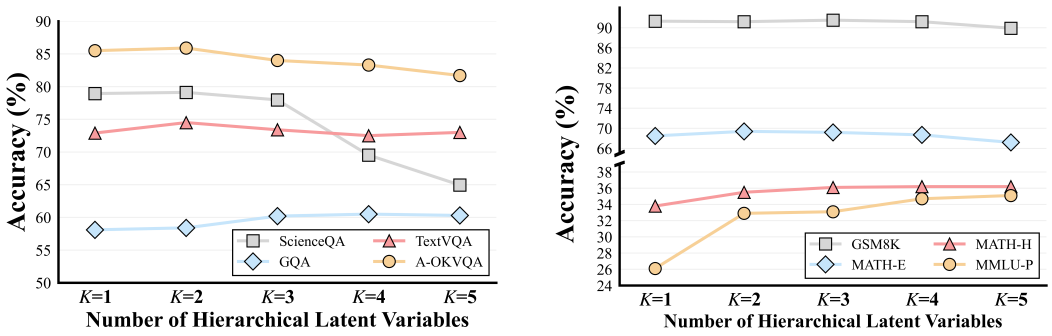
Table 2 shows the results on language reasoning benchmarks. The overall trend is consistent with the vision-language setting. HIERACOT-H achieves the best average performance on both Qwen2.5-Math-7B and Llama3.1-8B-Instruct. Compared with CoT prompting, it improves the average score by 5.15 points on Qwen2.5-Math-7B and 2.42 points on Llama3.1-8B-Instruct. The method also outperforms both Extracted and Learnable CoT-vector baselines, showing that the benefit of hierarchical variational CoT states is not limited to multimodal reasoning.

The performance gains are more pronounced on challenging benchmarks. For Qwen2.5-Math-7B, the largest improvement appears on MMLU-Pro, where HIERACOT-H improves over CoT prompting by 7.90 points. The hierarchical model also provides clear gains on MATH-H, suggesting that the global-local decomposition matters most when the task is harder and requires stronger input-specific adaptation. On GSM8K, however, HIERACOT-S slightly outperforms the hierarchical variant. This exception suggests that a single variational CoT state may already be sufficient for tasks where the baseline model already performs well.

On Llama3.1-8B-Instruct, HIERACOT-H ranks first on all four benchmarks, although the gains are smaller than those on Qwen2.5-Math-7B. This indicates that the method is not tied to a math-specialized model, but the magnitude of improvement depends on how much room the base model has for reasoning adaptation. Overall, the language results confirm that moving from deterministic CoT vectors to hierarchical variational CoT states improves average reasoning performance, with the hierarchy being most beneficial on more difficult tasks.

4.4 ABLATION AND ANALYSIS

Posterior input. We vary the information used to compute the posterior features in HIERACOT-H. Removing the separate gold answer reduces accuracy by at most 0.8 percentage points, whereas removing the CoT trace reduces accuracy by 2.2 to 7.7 points across four evaluated benchmarks. These results support the contribution of CoT information beyond the separate answer input. Appendix B.4 provides the settings, results, and scope of this comparison.



(a) Vision-language reasoning

(b) Language reasoning

Figure 3: **Effect of the number of latent levels K on per-benchmark accuracy.** $K = 1$ corresponds to HIERACOT-S, $K = 2$ to HIERACOT-H, and $K > 2$ adds further hierarchy levels under the same prior conditioning scheme.

Number of hierarchical levels. We vary the number of latent levels K from 1 to 5 on Qwen2.5-VL-7B and Qwen2.5-Math-7B. For $K > 1$, we keep the highest-level prior unconditional and condition each lower-level prior on the input and all higher-level latents, generalizing the two-level design of HIERACOT-H to deeper hierarchies. Figure 3 reports per-benchmark accuracy on vision-language and language-only reasoning tasks.

On vision-language tasks (Figure 3(a)), the two-level hierarchy ($K = 2$) reaches the best or near-best accuracy on every benchmark. Beyond $K = 2$, performance degrades on the higher-accuracy datasets ScienceQA and A-OKVQA, while GQA and TextVQA remain stable without showing additional gains. This suggests that adding levels beyond the global-local separation tends to hurt rather than help generalization on vision-language reasoning.

On language tasks (Figure 3(b)), the trend is less uniform. Adding the second level brings substantial gains on MMLU-P and MATH-H, while easier datasets such as GSM8K and MATH-E remain at high accuracy throughout. Beyond $K = 2$, MATH-H and MMLU-P continue to improve mildly and plateau around four levels, suggesting that harder reasoning tasks may benefit from finer decomposition of the latent state. The additional gain is small relative to the added model complexity, and we therefore adopt the two-level hierarchy ($K = 2$) in all main experiments.

Effect of the global prior structure. A key design choice in the hierarchical model is whether the prior over the global state should depend on the input. We compare an unconditional prior $p_{\psi_g}(\mathbf{g})$ with a conditional prior $p_{\psi_g}(\mathbf{g}|\mathbf{x})$, keeping all other settings fixed. As shown in Figure 4, the unconditional prior consistently outperforms the conditional one across all eight benchmarks. The improvements are uniformly positive on both vision-language and language reasoning, with the largest gain of 8.5 points on MMLU-P, followed by MATH-H, MATH-E, and TextVQA. This pattern supports the rationale behind the hierarchical structure. When the global prior is conditioned on the input, it is free to encode input-specific information, which overlaps with the role of the local state $\mathbf{z}$ and weakens the separation between the two levels. Removing this dependence encourages the global state $\mathbf{g}$ to focus on patterns that generalize across inputs, sharpening the division of roles between the two levels.

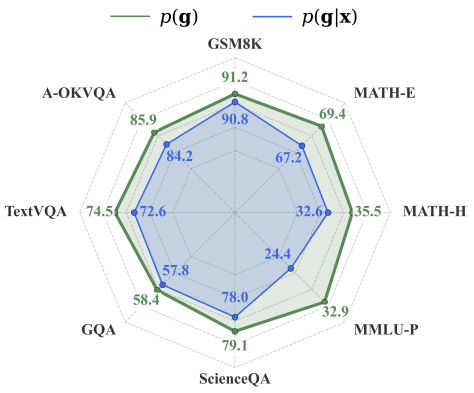


Figure 4: **Effect of the global prior.**

Cross-domain transferability. We further test whether the Chain-of-Thought state learned by HIERACOT captures domain-specific answer patterns or encodes a more general reasoning ability that transfers across domains. We construct two cross-domain transfer directions on each of two datasets. ScienceQA is split into Natural Science and Social Science, and MMLU-Pro is split into STEM and Humanities. For each direction, we train HIERACOT on the source domain and evaluate on the target domain. The model is kept frozen throughout.

Table 3: **Cross-domain transferability of HIERACOT.** The CoT column reports the baseline accuracy of the frozen backbone on the target domain. $\Delta_{\text{HieraCoT-S}}$ and $\Delta_{\text{HieraCoT-H}}$ report the absolute change in accuracy (in percentage points) over the CoT baseline.

Dataset	Model	Train Domain	Test Domain	CoT	$\Delta_{\text{HieraCoT-S}}$	$\Delta_{\text{HieraCoT-H}}$
ScienceQA	Qwen2.5-VL-7B	Natural	Social	88.40	−10.40	+0.60
		Social	Natural	70.20	+0.20	+0.40
MMLU-Pro	Qwen2.5-Math-7B	STEM	Humanities	23.60	+4.40	+6.40
		Humanities	STEM	27.00	+7.00	+5.20

As shown in Table 3, HIERACOT-H delivers positive or near-zero gains over the baseline in all four transfer directions. On MMLU-Pro, where the two domains are more distant in content, HIERACOT-H improves target-domain accuracy by 6.40 (STEM → Humanities) and 5.20 points (Humanities → STEM). This suggests that the CoT state learned by HIERACOT-H encodes a reasoning ability that transfers across domains, rather than merely capturing domain-specific answer patterns.

In contrast, HIERACOT-S exhibits a clear negative transfer of -10.40 points on ScienceQA when transferring from Natural Science to Social Science. This contrast suggests that the single-state model can be more vulnerable to overfitting to the training domain. The hierarchical design more cleanly separates shared reasoning patterns from input-specific adaptation, which keeps the global state $\mathbf{g}$ from absorbing domain-specific information. These results indicate that HIERACOT-H captures domain-invariant reasoning patterns rather than memorizing domain-specific answer behavior. Additional ablation on the injection layer is provided in Appendix C.

Functional separation between $\mathbf{g}$ and $\mathbf{z}$. To examine whether the two states in HIERACOT realize the intended functional separation, we draw 1,000 validation samples from each of the four vision-language benchmarks, obtain the posterior mean of $\mathbf{g}$ and the prior mean of $\mathbf{z}$ for every sample, and compute the standard deviation of each latent dimension across samples. We use the posterior mean for $\mathbf{g}$ to test whether the global state remains stable even under full training information, and the prior mean for $\mathbf{z}$ to measure the local state used at inference time. As shown in Figure 5, the standard deviations of $\mathbf{g}$ remain close to zero across all four datasets, while the standard deviations of $\mathbf{z}$ are substantially larger, with median ratios ranging from 7.8 to 20.2. Although the global posterior q_{ϕ_g} is free to output a different $\mathbf{g}$ for each input, it produces nearly the same value across samples, while $\mathbf{z}$ adjusts the CoT state to each individual input. This separation is not imposed by additional supervision. It arises from the ELBO together with the prior structure, and provides empirical support for the intended division of labor in which the global level remains stable across samples, while the local level captures input-specific variation.

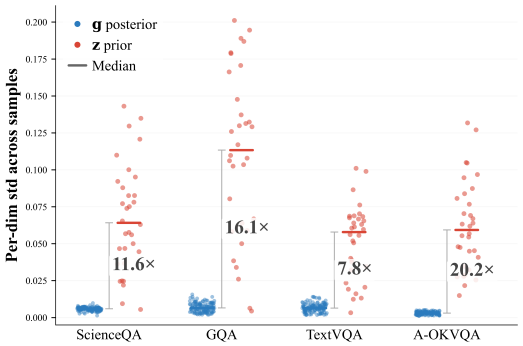


Figure 5: **Per-dimension standard deviation of $\mathbf{g}$ and $\mathbf{z}$ across samples.** Horizontal bars mark the median of each group, and the gray annotations report the median ratio between $\mathbf{z}$ and $\mathbf{g}$.

5 CONCLUSION

In this paper, we introduced HIERACOT, a hierarchical variational extension of CoT vectors for frozen large models. Instead of representing CoT-induced reasoning behavior with a single deterministic vector, HIERACOT models it as a structured distribution over Chain-of-Thought states. The global state captures reasoning tendencies shared across problems, while the local state adapts these tendencies to each input. By learning a posterior from training-time CoT and answer information and distilling it into an input-conditioned prior, HIERACOT enables lightweight hidden-state intervention at inference time without updating the parameters of the model or generating explicit rationales. Experiments on both vision-language and language-only reasoning show consistent improvements over direct prompting, chain-of-thought prompting, and deterministic CoT-vector baselines. Further analyses demonstrate the benefit of the hierarchical design, the functional separation between global and local states, and the transferability of the learned states across domains.

AI USE STATEMENT

Generative AI tools were used to assist with language editing and software development, including code debugging and refinement. All AI-assisted text and code were reviewed and verified by the authors. The authors take full responsibility for the final content of the paper and the correctness of the implementation.

REPRODUCIBILITY STATEMENT

Section 3 specifies the single-state and hierarchical latent-variable models, the variational objectives, and the hidden-state injection mechanism. Appendix A provides the derivation of the hierarchical evidence lower bound. Section 4 identifies the evaluated backbones, benchmarks, baselines, and hardware. Appendices B.1 and B.2 document model checkpoints, dataset preparation and sampling, latent-network architectures, optimization settings, and decoding and evaluation protocols for the vision-language and language experiments, respectively. Section 4.4 describes the hierarchy-depth, prior-structure, cross-domain transfer, and latent-variation analyses; Appendix C reports the injection-layer ablation and describes the held-out selection procedure.

REFERENCES

Mahtab Bigverdi, Zelun Luo, Cheng-Yu Hsieh, Ethan Shen, Dongping Chen, Linda G Shapiro, and Ranjay Krishna. Perception tokens enhance visual reasoning in multimodal language models. In Proceedings of the Computer Vision and Pattern Recognition Conference, pp. 3836–3845, 2025.

Yi Cao, Weijie Shi, Wei-Jie Xu, Yucheng Shen, Yue Cui, Hanghui Guo, Shimin Di, Ziyi Liu, Jiaming Li, Alexander Zhou, Jia Zhu, and Jiajie Xu. Tiv: Thought injection via vectors for efficient reasoning in large reasoning models. Proceedings of the AAAI Conference on Artificial Intelligence, 40(36):30148–30155, 2026.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 24185–24198, 2024.

Runxi Cheng, Feng Xiong, Yongxian Wei, Wanyun Zhu, and Chun Yuan. Whoever started the interference should end it: Guiding data-free model merging via task vectors. In Forty-second International Conference on Machine Learning, 2025.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. arXiv preprint arXiv:2110.14168, 2021.

Yuntian Deng, Yejin Choi, and Stuart Shieber. From explicit cot to implicit cot: Learning to internalize cot step by step. arXiv preprint arXiv:2405.14838, 2024.

Yuhao Dong, Zuyan Liu, Hai-Long Sun, Jingkang Yang, Winston Hu, Yongming Rao, and Ziwei Liu. Insight-v: Exploring long-chain visual reasoning with multimodal large language models. arXiv preprint arXiv:2411.14432, 2024.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. Pal: Program-aided language models. In International conference on machine learning, pp. 10764–10799. PMLR, 2023.

Jonas Geiping, Sean Michael McLeish, Neel Jain, John Kirchenbauer, Siddharth Singh, Brian R. Bartoldson, Bhavya Kailkhura, Abhinav Bhatele, and Tom Goldstein. Scaling up test-time compute with latent reasoning: A recurrent depth approach. In The Thirty-ninth Annual Conference on Neural Information Processing Systems, 2025.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. arXiv preprint arXiv:2407.21783, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. Nature, 645(8081):633–638, 2025.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. arXiv preprint arXiv:2412.06769, 2024.

Zhenghao He, Guangzhi Xiong, Bohan Liu, Sanchit Sinha, and Aidong Zhang. Reasoning beyond chain-of-thought: A latent computational mode in large language models, 2026.

Roee Hendel, Mor Geva, and Amir Globerson. In-context learning creates task vectors. In Findings of the Association for Computational Linguistics: EMNLP 2023, pp. 9318–9333, 2023.

Yushi Hu, Otilia Stretcu, Chun-Ta Lu, Krishnamurthy Viswanathan, Kenji Hata, Enming Luo, Ranjay Krishna, and Ariel Fuxman. Visual program distillation: Distilling tools and programmatic reasoning into vision-language models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9590–9601, 2024.

Jie Huang and Kevin Chen-Chuan Chang. Towards reasoning in large language models: A survey. In Findings of the association for computational linguistics: ACL 2023, pp. 1049–1065, 2023.

Xingshuai Huang, Derek Li, Bahareh Nikpour, and Parsa Omidi. Hierarchical chain-of-thought prompting: Enhancing llm reasoning performance and efficiency, 2026.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In The Eleventh International Conference on Learning Representations, 2023.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. arXiv preprint arXiv:2412.16720, 2024.

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.

Bangzheng Li, Ximeng Sun, Jiang Liu, Ze Wang, Jialian Wu, Xiaodong Yu, Emad Barsoum, Muhao Chen, and Zicheng Liu. Latent visual reasoning. In The Fourteenth International Conference on Learning Representations, 2026a.

Li Li, Ziyi Wang, Yongliang Wu, Jianfei Cai, and Xu Yang. Cot vectors: Transferring and probing the reasoning mechanisms of LLMs. In The Fourteenth International Conference on Learning Representations, 2026b.

Zejun Li, Ruipu Luo, Jiwen Zhang, Minghui Qiu, and Zhongyu Wei. Vocot: Unleashing visually grounded multi-step reasoning in large multi-modal models. arXiv preprint arXiv:2405.16919, 2024.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In The twelfth international conference on learning representations, 2023.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In International Conference on Learning Representations, 2019.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. Advances in neural information processing systems, 35:2507–2521, 2022.

M-A-P. A survey on latent reasoning. arxiv, 2025.

Chancharik Mitra, Brandon Huang, Trevor Darrell, and Roei Herzig. Compositional chain of thought prompting for large multimodal models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2024.

Guillermo Ortiz-Jimenez, Alessandro Favero, and Pascal Frossard. Task arithmetic in the tangent space: Improved editing of pre-trained models. Advances in Neural Information Processing Systems, 36:66727–66754, 2023.

Tan-Hanh Pham and Chris Ngo. Multimodal chain of continuous thought for latent-space reasoning in vision-language models. arXiv preprint arXiv:2508.12587, 2025.

Du Phan, Matthew Douglas Hoffman, 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. Advances in Neural Information Processing Systems, 36:72819–72841, 2023.

Ji Qi, Ming Ding, Weihan Wang, Yushi Bai, Qingsong Lv, Wenyi Hong, Bin Xu, Lei Hou, Juanzi Li, Yuxiao Dong, and Jie Tang. Cogcom: A visual language model with chain-of-manipulations reasoning. In The Thirteenth International Conference on Learning Representations, 2025.

Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge. In European conference on computer vision, pp. 146–162. Springer, 2022.

Hao Shao, Shengju Qian, Han Xiao, Guanglu Song, Zhuofan Zong, Letian Wang, Yu Liu, and Hongsheng Li. Visual cot: Advancing multi-modal language models with a comprehensive dataset and benchmark for chain-of-thought reasoning. Advances in Neural Information Processing Systems, 37:8612–8642, 2024.

Zhenyi Shen, Hanqi Yan, Linhai Zhang, Zhanghao Hu, Yali Du, and Yulan He. Codi: Compressing chain-of-thought into continuous space via self-distillation. In Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pp. 677–693, 2025.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 8317–8326, 2019.

Wenhui Tan, Jiaze Li, Jianzhong Ju, Zhenbo Luo, Ruihua Song, and Jian Luan. Think silently, think fast: Dynamic latent compression of LLM reasoning chains. In The Thirty-ninth Annual Conference on Neural Information Processing Systems, 2025.

Eric Todd, Millicent Li, Arnab Sen Sharma, Aaron Mueller, Byron C Wallace, and David Bau. Function vectors in large language models. In The Twelfth International Conference on Learning Representations, 2024.

Fanmeng Wang, Haotian Liu, Guojiang Zhao, Hongteng Xu, and Zhifeng Gao. Regular: Variational latent reasoning guided by rendered chain-of-thought. arXiv preprint arXiv:2601.23184, 2026a.

Guan Wang, Jin Li, Yuhao Sun, Xing Chen, Changling Liu, Yue Wu, Meng Lu, Sen Song, and Yasin Abbasi Yadkori. Hierarchical reasoning model, 2025a. URL https://arxiv.org/abs/2506.21734.

Lei Wang, Yi Hu, Jiabang He, Xing Xu, Ning Liu, Hui Liu, and Heng Tao Shen. T-sciq: Teaching multimodal chain-of-thought reasoning via large language model signals for science question answering. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 38, pp. 19162–19170, 2024a.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. arXiv preprint arXiv:2409.12191, 2024b.

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, 58(8):1–35, 2026b.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In The Eleventh International Conference on Learning Representations, 2023.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. Advances in Neural Information Processing Systems, 37:95266–95290, 2024c.

Zhichang Wang, Xianwei Zhuang, Zhihong Zhu, and Yuexian Zou. Hcott: Hierarchical chain-of-thought distillation. In ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1–5, 2025b.

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.

Xilin Wei, Xiaoran Liu, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Jiaqi Wang, Xipeng Qiu, and Dahua Lin. SIM-COT: Supervised implicit chain-of-thought. In International Conference on Learning Representations, 2026.

Qiong Wu, Xiangcong Yang, Yiyi Zhou, Chenxin Fang, Baiyang Song, Xiaoshuai Sun, and Rongrong Ji. Grounded chain-of-thought for multimodal large language models. arXiv preprint arXiv:2503.12799, 2025.

Yu Xia, Rui Wang, Xu Liu, Mingyan Li, Tong Yu, Xiang Chen, Julian McAuley, and Shuai Li. Beyond chain-of-thought: A survey of chain-of-x paradigms for llms. In Proceedings of the 31st International Conference on Computational Linguistics, pp. 10795–10809, 2025.

Fengli Xu, Qianyue Hao, Chenyang Shao, Zefang Zong, Yu Li, Jingwei Wang, Yunke Zhang, Jingyi Wang, Xiaochong Lan, Jiahui Gong, et al. Toward large reasoning models: A survey of reinforced reasoning with large language models. Patterns, 6(10), 2025.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. Advances in neural information processing systems, 36:7093–7115, 2023.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, et al. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. arXiv preprint arXiv:2409.12122, 2024a.

Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. Adamerging: Adaptive model merging for multi-task learning. The Twelfth International Conference on Learning Representations, 2024b.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. Advances in neural information processing systems, 36:11809–11822, 2023.

Nicolas Yax, Hernán Anlló, and Stefano Palminteri. Studying and improving reasoning in humans and machines. Communications Psychology, 2(1):51, 2024.

Fei Yu, Hongbo Zhang, Prayag Tiwari, and Benyou Wang. Natural language reasoning, a survey. ACM Computing Surveys, 56(12):1–39, 2024.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. Star: Bootstrapping reasoning with reasoning. Advances in Neural Information Processing Systems, 35:15476–15488, 2022.

Boyuan Zhang, Yingjun Du, Xiantong Zhen, and Ling Shao. Variational task vector composition. In The Thirty-ninth Annual Conference on Neural Information Processing Systems, 2025a.

Yiming Zhang, Qiangyu Yan, Borui Jiang, and Kai Han. Multimodal latent reasoning via hierarchical visual cues injection, 2026.

Zhen Zhang, Xuehai He, Weixiang Yan, Ao Shen, Chenyang Zhao, Shuohang Wang, Yelong Shen, and Xin Eric Wang. Soft thinking: Unlocking the reasoning potential of llms in continuous concept space. arXiv preprint arXiv:2505.15778, 2025b.

Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola. Multimodal chain-of-thought reasoning in language models. arXiv preprint arXiv:2302.00923, 2023.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. arXiv preprint arXiv:2303.18223, 1(2):1–124, 2023.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc V Le, and Ed H. Chi. Least-to-most prompting enables complex reasoning in large language models. In The Eleventh International Conference on Learning Representations, 2023.

Xiangxin Zhou, Zichen Liu, Haonan Wang, Chao Du, Min Lin, Chongxuan Li, Liang Wang, and Tianyu Pang. Variational reasoning for language models. In The Fourteenth International Conference on Learning Representations, 2026.

Hanlin Zhu, Shibo Hao, Zhiting Hu, Jiantao Jiao, Stuart Russell, and Yuandong Tian. Reasoning by superposition: A theoretical perspective on chain of continuous thought. In ICML 2025 Workshop on Methods and Opportunities at Small Scale, 2025.

A Derivation of the Hierarchical ELBO

This appendix provides the full derivation of the hierarchical evidence lower bound in equation 7 of Section 3.3.

Starting from the marginal likelihood:

$$\log p(\mathbf{y}|\mathbf{x}) = \log \int p(\mathbf{y}, \mathbf{g}, \mathbf{z}|\mathbf{x})\, d\mathbf{g}\, d\mathbf{z}. \tag{9}$$

Introducing the variational posterior $q_\phi(\mathbf{g}, \mathbf{z}|\mathbf{x}, \mathbf{r})$, where $\mathbf{r}$ is the auxiliary information defined in equation 3, and applying Jensen's inequality:

$$\begin{aligned} \log p(\mathbf{y}|\mathbf{x}) &= \log \int q_\phi(\mathbf{g}, \mathbf{z}|\mathbf{x}, \mathbf{r})\, \frac{p(\mathbf{y}, \mathbf{g}, \mathbf{z}|\mathbf{x})}{q_\phi(\mathbf{g}, \mathbf{z}|\mathbf{x}, \mathbf{r})}\, d\mathbf{g}\, d\mathbf{z} \\ &\geq \mathbb{E}_{q_\phi(\mathbf{g},\mathbf{z}|\mathbf{x},\mathbf{r})} \left[\log p(\mathbf{y}, \mathbf{g}, \mathbf{z}|\mathbf{x}) - \log q_\phi(\mathbf{g}, \mathbf{z}|\mathbf{x}, \mathbf{r})\right]. \end{aligned} \tag{10}$$

Using the factorization of the joint distribution:

$$p(\mathbf{y}, \mathbf{g}, \mathbf{z}|\mathbf{x}) = p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{g}, \mathbf{z})\, p_{\psi_g}(\mathbf{g})\, p_{\psi_z}(\mathbf{z}|\mathbf{x}, \mathbf{g}), \tag{11}$$

with the factorization of the variational posterior:

$$q_\phi(\mathbf{g}, \mathbf{z}|\mathbf{x}, \mathbf{r}) = q_{\phi_g}(\mathbf{g}|\mathbf{x}, \mathbf{r})\, q_{\phi_z}(\mathbf{z}|\mathbf{x}, \mathbf{r}, \mathbf{g}). \tag{12}$$

Substituting these factorizations into equation 10 and grouping terms by their corresponding factors yields:

$$\begin{aligned} \mathcal{L}_{\text{ELBO}} = \mathbb{E}_{q_\phi(\mathbf{g},\mathbf{z}|\mathbf{x},\mathbf{r})} &\log p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{g}, \mathbf{z}) \\ &+ \mathbb{E}_{q_\phi(\mathbf{g},\mathbf{z}|\mathbf{x},\mathbf{r})} \left[\log p_{\psi_g}(\mathbf{g}) - \log q_{\phi_g}(\mathbf{g}|\mathbf{x}, \mathbf{r})\right] \\ &+ \mathbb{E}_{q_\phi(\mathbf{g},\mathbf{z}|\mathbf{x},\mathbf{r})} \left[\log p_{\psi_z}(\mathbf{z}|\mathbf{x}, \mathbf{g}) - \log q_{\phi_z}(\mathbf{z}|\mathbf{x}, \mathbf{r}, \mathbf{g})\right]. \end{aligned} \tag{13}$$

For the global term, the integrand depends only on $\mathbf{g}$, so marginalizing $\mathbf{z}$ out of $q_\phi(\mathbf{g}, \mathbf{z}|\mathbf{x}, \mathbf{r})$ leaves $q_{\phi_g}(\mathbf{g}|\mathbf{x}, \mathbf{r})$:

$$\begin{aligned} &\mathbb{E}_{q_\phi(\mathbf{g},\mathbf{z}|\mathbf{x},\mathbf{r})} \left[\log p_{\psi_g}(\mathbf{g}) - \log q_{\phi_g}(\mathbf{g}|\mathbf{x}, \mathbf{r})\right] \\ &= \mathbb{E}_{q_{\phi_g}(\mathbf{g}|\mathbf{x},\mathbf{r})} \left[\log p_{\psi_g}(\mathbf{g}) - \log q_{\phi_g}(\mathbf{g}|\mathbf{x}, \mathbf{r})\right] \\ &= -\mathbb{D}_{\text{KL}}\big(q_{\phi_g}(\mathbf{g}|\mathbf{x}, \mathbf{r}) \,\|\, p_{\psi_g}(\mathbf{g})\big). \end{aligned} \tag{14}$$

For the local term, applying the tower property to the nested factorization $q_\phi(\mathbf{g}, \mathbf{z}|\mathbf{x}, \mathbf{r}) = q_{\phi_g}(\mathbf{g}|\mathbf{x}, \mathbf{r})\, q_{\phi_z}(\mathbf{z}|\mathbf{x}, \mathbf{r}, \mathbf{g})$ yields:

$$\begin{aligned} &\mathbb{E}_{q_\phi(\mathbf{g},\mathbf{z}|\mathbf{x},\mathbf{r})} \left[\log p_{\psi_z}(\mathbf{z}|\mathbf{x}, \mathbf{g}) - \log q_{\phi_z}(\mathbf{z}|\mathbf{x}, \mathbf{r}, \mathbf{g})\right] \\ &= \mathbb{E}_{q_{\phi_g}(\mathbf{g}|\mathbf{x},\mathbf{r})} \mathbb{E}_{q_{\phi_z}(\mathbf{z}|\mathbf{x},\mathbf{r},\mathbf{g})} \left[\log p_{\psi_z}(\mathbf{z}|\mathbf{x}, \mathbf{g}) - \log q_{\phi_z}(\mathbf{z}|\mathbf{x}, \mathbf{r}, \mathbf{g})\right] \\ &= -\mathbb{E}_{q_{\phi_g}(\mathbf{g}|\mathbf{x},\mathbf{r})} \mathbb{D}_{\text{KL}}(q_{\phi_z}(\mathbf{z}|\mathbf{x}, \mathbf{r}, \mathbf{g}) \,\|\, p_{\psi_z}(\mathbf{z}|\mathbf{x}, \mathbf{g})). \end{aligned} \tag{15}$$

Substituting equation 14 and equation 15 into equation 13 gives the hierarchical evidence lower bound:

$$\begin{aligned} \log p(\mathbf{y}|\mathbf{x}) \geq \mathbb{E}_{q_\phi(\mathbf{g},\mathbf{z}|\mathbf{x},\mathbf{r})} &\left[\log p_\theta(\mathbf{y}|\mathbf{x}, \mathbf{g}, \mathbf{z})\right] \\ &- \mathbb{D}_{\text{KL}}\big(q_{\phi_g}(\mathbf{g}|\mathbf{x}, \mathbf{r}) \,\|\, p_{\psi_g}(\mathbf{g})\big) \\ &- \mathbb{E}_{q_{\phi_g}(\mathbf{g}|\mathbf{x},\mathbf{r})} \mathbb{D}_{\text{KL}}(q_{\phi_z}(\mathbf{z}|\mathbf{x}, \mathbf{r}, \mathbf{g}) \,\|\, p_{\psi_z}(\mathbf{z}|\mathbf{x}, \mathbf{g})), \end{aligned} \tag{16}$$

which recovers equation 7 in the main text.

Summing over the dataset yields the training objective:

$$\sum_{i=1}^{|\mathcal{D}|} \log p(\mathbf{y}_i|\mathbf{x}_i) \;\geq\; \sum_{i=1}^{|\mathcal{D}|} \mathcal{L}_{\text{ELBO}}^{(i)}. \tag{17}$$

B Implementation Details and Additional Results

This appendix provides full implementation details for vision-language reasoning (Section B.1) and language reasoning (Section B.2), and an additional ablation on the injection layer (Section C).

B.1 Vision-Language Reasoning

Models. We use two frozen vision-language models. **Qwen2.5-VL-7B-Instruct**[1] (Wang et al., 2024b) has a language model with hidden size $d = 3{,}584$ and 28 transformer layers. **InternVL2.5-8B**[2] (Chen et al., 2024) has a language model with hidden size $d = 4{,}096$ and 32 transformer layers. Both models are loaded in `bfloat16`, and all model parameters θ remain frozen throughout training and inference.

Datasets. We evaluate on four vision-language reasoning benchmarks: ScienceQA (Lu et al., 2022), Visual-CoT-GQA (Shao et al., 2024), TextVQA (Singh et al., 2019), and A-OKVQA (Schwenk et al., 2022). ScienceQA is restricted to the image-bearing subset. For Visual-CoT-GQA, we retain only the GQA-derived portion. All datasets are converted into a unified JSONL format containing image path, question, gold answer, and chain-of-thought annotations where available. For multiple-choice datasets, answer choices are formatted as option lists, and the target is the corresponding option letter.

Training setup. For each experiment, we sample up to $1{,}000$ training instances and evaluate on $1{,}000$ test instances for Visual-CoT-GQA, TextVQA, and A-OKVQA, and on the full validation set (2,097 instances) for ScienceQA. All inference networks (priors and posteriors) and projection matrices are trained with AdamW (Loshchilov & Hutter, 2019) using learning rate 1×10^{-4}, weight decay 1×10^{-3}, linear warmup, and cosine decay. The effective batch size is 4, the maximum input length is $1{,}024$ tokens, and training runs for 3 epochs. The KL weight is linearly warmed up over the first 200 optimization steps. During training, the posterior receives features pooled from the full sample $(\mathbf{x}, \mathbf{c}, \mathbf{y})$, while the prior receives features from the input $\mathbf{x}$ only.

Latent architectures. The vision-language setting uses a bottleneck latent space, with the latent dimension d_z (and d_g) much smaller than the model hidden size d. The bottleneck limits the capacity of the latent representation. Each prior and posterior network is implemented as an MLP with two hidden layers of width 512, followed by two linear output heads that produce the mean and the standard deviation.

For HieraCoT-S, the single latent $\mathbf{z}$ has dimension $d_z = 128$ and is projected to the hidden space $\mathbb{R}^d$ by a learned linear map $\mathbf{W}_z \in \mathbb{R}^{d \times d_z}$ before injection following equation 1 of the main text. The KL weight is set to $\beta = 5.0$.

For HieraCoT-H, the global latent $\mathbf{g}$ has dimension $d_g = 128$ and the local latent $\mathbf{z}$ has dimension $d_z = 32$. The global prior is a diagonal Gaussian with learnable mean and standard-deviation parameters. The local prior, global posterior, and local posterior are each MLPs with two hidden layers of width 512, each followed by two linear output heads for the mean and the standard deviation. Two separate projections $\mathbf{W}_g \in \mathbb{R}^{d \times d_g}$ and $\mathbf{W}_z \in \mathbb{R}^{d \times d_z}$ map each latent to the hidden space, following equation 8 of the main text. The KL weights are set to $\beta_g = 0.5$ for the global term and $\beta_z = 5.0$ for the local term.

[1] `https://huggingface.co/Qwen/Qwen2.5-VL-7B-Instruct`

[2] `https://huggingface.co/OpenGVLab/InternVL2_5-8B`

The ℓ_2 norm of each projected vector is clamped before injection, which improves training stability. We use $\tau_g = 10.0$ for the projected global vector and $\tau_z = \tau_g/3 \approx 3.33$ for the projected local vector. For HIERACOT-S, the single projected vector is clamped at $\tau = 10.0$. The latent network architecture and all optimization hyperparameters are shared across the two models; only the projection output dimension changes to match each model's hidden size.

Baselines. We compare against four baselines, all of which use the frozen model without our learned latent. *Direct* produces an answer directly from the input without chain-of-thought prompting. *CoT* (Kojima et al., 2022) prepends a step-by-step prompt before answer generation. *Extracted* (Li et al., 2026b) computes a fixed CoT vector from the mean activation difference between teacher and student passes over the support set. *Learnable* (Li et al., 2026b) optimizes a single global vector with a teacher-student objective combining hidden-state alignment and answer prediction. The two deterministic CoT-vector baselines are injected additively with scaling factor 1.0 at the same layer used by our method.

Evaluation. We report exact-match accuracy. At inference time, latent samples are drawn from the prior alone, which depends only on the input $\mathbf{x}$. Generation uses greedy decoding with up to 128 new tokens. For multiple-choice tasks, predictions are normalized to a single option letter. For open-ended tasks, predicted and gold answers are compared after standard text normalization.

B.2 LANGUAGE REASONING

Models. We use two frozen language models. **Qwen2.5-Math-7B**[3] (Yang et al., 2024a) has hidden size $d = 3{,}584$ and 28 transformer layers. **Llama-3.1-8B-Instruct**[4] (Grattafiori et al., 2024) has hidden size $d = 4{,}096$ and 32 transformer layers. Both models are loaded in `bfloat16`, and all model parameters θ remain frozen throughout training and inference.

Datasets. We evaluate on GSM8K (Cobbe et al., 2021), MATH-E, MATH-H, and MMLU-Pro (Wang et al., 2024c). Following CoT Vectors (Li et al., 2026b), MATH is split into an easy subset (Levels 1–3) and a hard subset (Levels 4–5). All training samples include chain-of-thought annotations. Final answers are formatted uniformly within `\boxed{}`. For MMLU-Pro, each question is concatenated with its answer choices, and the target is the correct option letter.

Training setup. For each experiment, we sample up to $3{,}000$ training instances. All inference networks and projections are trained with AdamW using learning rate 1×10^{-4}, weight decay 1×10^{-3}, linear warmup, and cosine decay. The effective batch size is 4, the maximum input length is $1{,}024$ tokens, and training runs for 5 epochs. As in the vision-language setting, the posterior receives features pooled from the full sample $(\mathbf{x}, \mathbf{c}, \mathbf{y})$, while the prior receives features from the input $\mathbf{x}$ only.

Latent architectures. Unlike the vision-language setting, the language setting does not use a bottleneck. The latent dimension matches the model hidden size directly, so $d_z = d$ for HIERACOT-S and $d_g = d_z = d$ for HIERACOT-H. Each prior and posterior network is implemented as an MLP with two hidden layers of width 512, followed by two linear output heads for the mean and the standard deviation. The global prior is a diagonal Gaussian with learnable mean and standard-deviation parameters. Since the latent and hidden spaces have the same dimension, the projections $\mathbf{W}_z$ and $\mathbf{W}_g$ in equation 1 and equation 8 of the main text reduce to identity maps.

Baselines. The same four baselines are used as in the vision-language setting (Section B.1).

Evaluation. We report exact-match accuracy on $1,000$ randomly sampled test instances per dataset. Generation uses up to 512 new tokens, with beam search (beam size 3) for Qwen2.5-Math-7B and greedy decoding for Llama-3.1-8B-Instruct. Final answers are extracted from the generated text and compared with gold answers after normalization. For MMLU-Pro, predictions are further normalized to a single option letter.

[3] `https://huggingface.co/Qwen/Qwen2.5-Math-7B`

[4] `https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct`

Table 4: **Effect of the posterior input on reasoning accuracy.** All variants use HIERACOT-H. Full, No answer, and No CoT use $(\mathbf{x}, \mathbf{c}, \mathbf{y})$, $(\mathbf{x}, \mathbf{c})$, and $(\mathbf{x}, \mathbf{y})$, respectively, to compute the posterior features. Accuracy is reported in percent.

Model	Dataset	Full	No_answer	No_CoT
Qwen2.5-Math-7B	GSM8K	**91.2**	90.8	86.6
	MMLU-Pro	**32.9**	32.1	25.2
Qwen2.5-VL-7B	ScienceQA	**79.1**	78.8	76.7
	GQA	**58.4**	58.2	56.2

B.3 EVALUATION PROTOCOL AND BASELINE IMPLEMENTATION

Our language experiments use the backbone models and benchmark split definitions adopted by CoT Vectors (Li et al., 2026b). We independently implement Direct, CoT, Extracted, Learnable, and HIERACOT and evaluate them on the same test examples. For each backbone, all methods share the decoding settings, answer extraction procedure, and normalization rules. Each method uses the prompt specified by its definition in Section 4.

Evaluation settings. We sample 1,000 test examples per benchmark using seed 42 and keep them fixed across methods and training seeds. We use raw prompts for Qwen2.5-Math-7B and the official chat template for Llama-3.1-8B-Instruct. Decoding settings are provided in Appendix B.2. For CoT prompting on GSM8K and MATH, we use:

> You are a helpful and precise assistant for solving math problems. Please reason step by step, and put your final answer within \boxed{}.

For MMLU-Pro, we use the prompt template in Appendix A.2.2 of CoT Vectors (Li et al., 2026b), with answer choices appended to the question. Our answer parser extracts the last \boxed{} expression by matching opening and closing braces, including nested braces. MMLU-Pro predictions are normalized to a single option letter.

Relation to the reported baseline scores. All baseline scores in Table 2 are obtained with our evaluation pipeline, and the reported gains are based on comparisons under the shared conditions described above.

B.4 POSTERIOR INPUT ABLATION

We examine the contributions of the CoT trace $\mathbf{c}$ and the separate gold answer $\mathbf{y}$ to the posterior features in HIERACOT-H. We compare three inputs to the same feature extraction and pooling operation:

$$\mathbf{r}_{\text{Full}} = \mathcal{R}_\theta^{(l)}(\mathbf{x}, \mathbf{c}, \mathbf{y}),$$
$$\mathbf{r}_{\text{No_answer}} = \mathcal{R}_\theta^{(l)}(\mathbf{x}, \mathbf{c}),$$
$$\mathbf{r}_{\text{No_CoT}} = \mathcal{R}_\theta^{(l)}(\mathbf{x}, \mathbf{y}).$$

The selected feature vector is used by both posterior networks. All variants use the same architecture, training data, and optimization settings. The gold answer remains the prediction target in the likelihood term, including for No answer. At inference, all variants use only the priors, without access to the trace or gold answer.

Table 4 shows that removing the separate answer reduces accuracy by 0.2 to 0.8 percentage points, whereas removing the CoT trace reduces accuracy by 2.2 to 7.7 points. On MMLU-Pro, No CoT achieves 25.2%, close to the CoT prompting baseline of 25.0% in Table 2. No answer achieves 32.1%, compared with 32.9% for Full. The same ordering holds on the other three benchmarks.

These results indicate that the CoT trace provides useful information beyond the separate answer input, whose removal has a smaller effect on the evaluated tasks. They do not fully separate reasoning information from answer information, since the trace itself may contain the final answer or information closely related to it.

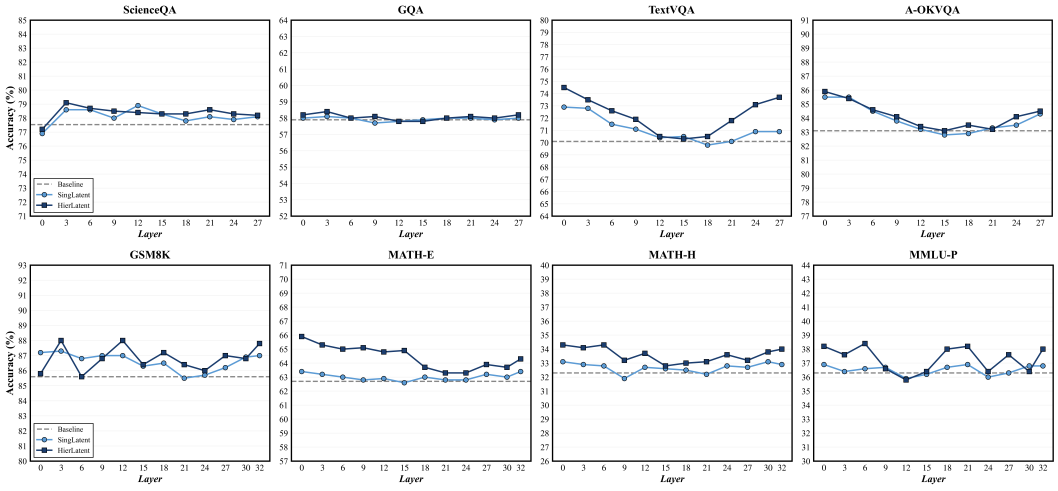


Figure 6: **Effect of the injection layer on reasoning accuracy.** Each panel shows accuracy as a function of the injection layer for both HIERACOT-S and HIERACOT-H. *Top*: vision-language reasoning on Qwen2.5-VL-7B (28 layers). *Bottom*: language reasoning on Llama3.1-8B-Instruct (32 layers). The dashed line indicates the baseline without injection.

C INJECTION LAYER ABLATION

We vary the injection layer in Qwen2.5-VL-7B and Llama-3.1-8B-Instruct and report accuracy on each benchmark for HIERACOT-S and HIERACOT-H. The dashed line in each panel indicates the baseline without injection.

Figure 6 shows that performance depends on both the benchmark and the injection layer. In the vision-language setting, early layers give strong results on TextVQA and A-OKVQA, but accuracy does not decrease steadily with depth. In the language setting, HIERACOT-H generally outperforms HIERACOT-S on MATH-E and MATH-H, while their relative performance varies across layers on GSM8K and MMLU-Pro. Some layer choices fall below the baseline without injection. We select the injection layer on a separate validation set for each backbone and fix it before test evaluation.

D LIMITATIONS

The number of latent levels is selected empirically, and the latent states are injected at a single layer chosen using validation data. Adapting the hierarchy depth and injection location remains a direction for future work. Our transfer results cover a limited set of distribution shifts and do not guarantee improvements on other distributions. The current method has no automatic mechanism to detect harmful injections or disable them. Finally, the posterior input ablation supports the contribution of CoT traces beyond the separate answer input, but does not fully separate reasoning information from answer information within those traces.