BEYOND ISOLATED MECHANISMS: COORDINATED REPRESENTATION LEARNING FOR SYSTEMIC REASONING IN LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing methods typically enhance the reasoning capabilities of large language models through four mechanisms, which are often studied in isolation: explicit supervision, implicit knowledge transfer, optimization tuning, and structural constraints. Since these mechanisms are frequently introduced as independent components rather than optimized under a unified objective, their interactions remain under-explored, and potential synergies may be overlooked. We propose Coordinated Representation Learning (CRL), a unified training framework that jointly optimizes these signals: grain-invariant chain-of-thought supervision ensures consistency in reasoning across different trajectory lengths; multi-teacher distillation injects expert knowledge into hidden states; difficulty-aware curriculum learning stabilizes training dynamics; and a structured subspace adapter enables efficient updates. To investigate the interactions among these mechanisms, we compare CRL with an independent compositional baseline that includes the same four components but optimizes them as separate, uncoordinated objectives. Across various model scales and in mathematics, logic, and coding benchmarks, CRL consistently outperforms supervised fine-tuning and standard distillation baselines. More importantly, CRL outperforms the independent combination baseline, revealing positive synergies: the gains from coordinated joint optimization exceed the sum of the individual mechanisms. Ablation experiments further demonstrate that removing any component degrades performance, and the synergy is particularly evident in terms of robustness and sample efficiency under distribution shifts and adversarial conditions. These results suggest that explicitly modeling and coordinating complementary training signals toward a unified objective is a promising path toward achieving stronger system reasoning.[1].

1 INTRODUCTION

Existing methods typically enhance the reasoning capabilities of large language models (LLMs) through four mechanisms, which are often studied in isolation: explicit supervision, implicit knowledge transfer, optimization tuning, and structural constraints. Since these mechanisms are frequently introduced as independent components rather than optimized under a unified objective, their interactions remain under-explored, and potential synergy may be overlooked.

We view reasoning enhancement as a representation learning problem in which these four signals must be coordinated rather than simply stacked. Traditional KD approaches often focus solely on matching final output distributions Hinton et al. (2015), and therefore neglect the rich intermediate reasoning representations of the teacher. Conversely, supervised fine-tuning (SFT) with CoT data can be sample-inefficient and may lead to inconsistent learning dynamics because problem difficulty varies and target reasoning trajectories are complex and multi-step. In addition, aligning representations from teachers that provide long, detailed CoTs with those from teachers that provide shorter, more abstract reasoning chains is beneficial: the former offers fine-grained intermediate steps, while the latter captures higher-level logical leaps. Aligning them encourages the student to learn robust and consistent intermediate representations that are invariant to superficial variations in explanation detail.

[1] https://anonymous.4open.science/r/Core-CRL-code-yzq-DBF6

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.

To operationalize this idea, we propose Coordinated Representation Learning (CRL), a unified training framework that jointly optimizes four complementary signals. We summarize the main contributions of this work as follows: 1. A unified coordinated training framework. We propose Coordinated Representation Learning (CRL), which coordinates four complementary training signals under a unified objective; 2. We define a sequence-level latent reasoning representation $h(x, r)$ by aggregating hidden states from the middle to upper layers of the Transformer; 3. We introduce a latent space distillation objective $\mathcal{L}_{\text{rep-district}}$, which aligns the student's reasoning representation with the average representation of multiple teachers via a shared lightweight projection P. 4. We formulate curriculum learning as a mechanism for gradually stabilizing reasoning representations. A difficulty score is assigned to each sample based on model confidence, inference trajectory length, and local latent variance estimated via k-nearest neighbors, and samples are selected using a cosine-expanded threshold planning scheme. 5. We constrain parameter updates to a structured low-dimensional subspace by assigning lightweight adapters to K inference primitives and activating them via a soft routing mechanism. The core novelty of CRL lies not in inventing these components *per se*, but in their *coordinated, end-to-end optimization* within a shared latent representation framework.

2 Related Work

2.1 Chain-of-Thought and Stepwise Reasoning

Chain-of-thought (CoT) prompting and fine-tuning explicitly supervise intermediate reasoning processes. Such supervision encourages models to externalize latent computation and develop structured intermediate representations for complex reasoning tasks (Yu et al., 2025a; Ou & Yin, 2025; Cheng et al., 2025). Variants such as long-CoT supervision (Ou & Yin, 2025), mixture-of-length rationales (Yu et al., 2025a), and activation-level control mechanisms (Zhao et al., 2025) further improve accuracy. Therefore, explicit supervision serves as one core signal in our unified framework, but it is not sufficient by itself.

Recent studies reveal limitations: CoT traces are sensitive to annotation quality, prone to hallucinated steps under adversarial conditions, and unfaithful to the model's internal computations (Xu et al., 2025; Mohammadi et al., 2025; Gao et al., 2026). These limitations suggest that explicit stepwise supervision alone is insufficient to induce robust and faithful reasoning representations; therefore, CRL coordinates it with implicit transfer, optimization scheduling, and structural constraints.

2.2 Distillation and Teacher–Student Reasoning Transfer

Knowledge distillation provides an implicit mechanism for transferring reasoning behaviors from large, high-capacity teacher models to smaller students (Cheng et al., 2025). Intermediate hidden-state distillation has been studied for years, as illustrated by FitNets Romero et al. (2015); multi-teacher and ensemble distillation have also been explored extensively Fukuda et al. (2017). Recent work further emphasizes reasoning-aware distillation strategies, including multi-path generation, self-consistency filtering, and modular transfer mechanisms such as Multi-LoRA composition (Li et al., 2025b). However, distillation effectiveness is contingent on teacher reliability and alignment. Empirical analyses show that representation mismatches between teachers and students can limit transfer efficiency and amplify systematic reasoning biases (Ienco & Dantas, 2024; Baek & Tegmark, 2025). Therefore, CRL coordinates these signals with granularity-invariant supervision, difficulty-aware scheduling, and structured subspace updates under a unified representation objective, and we align teacher and student reasoning spaces through a shared latent projection rather than only through output-level matching.

2.3 Efficient Fine-Tuning and Structural Constraints

Parameter-efficient fine-tuning (PEFT) methods, including Low-Rank Adaptation (LoRA), adapters, and block-circulant updates, enable scalable adaptation under constrained computational budgets (Ding et al., 2025; Pletenev et al., 2025). Recent explorations of concept-aware tuning and structure-aware updates impose inductive biases on how representations are modified (Chen et al., 2025; Liu et al., 2025). However, analyses suggest that parameter-efficient updates primarily affect localized subspaces and do not, by themselves, guarantee alignment with systematic reasoning behaviors (Chen

et al., 2022; Tastan et al., 2025). Therefore, we compare structured subspace updates with full dense fine-tuning under matched compute in our experiments, and we report this comparison in App. B.3. We coordinate structural constraints with reasoning-specific supervision rather than treating them as a standalone replacement for dense updates.

2.4 Curriculum Learning and Optimization Dynamics

Curriculum learning and data selection methods influence reasoning performance by shaping optimization dynamics and training trajectories (Cheng et al., 2025). Difficulty-aware strategies, such as Difficulty-aware Data Selection (D3) (Zhang et al., 2025), hardness-based sampling, and staged fine-tuning (Li et al., 2025a), improve convergence stability and sample efficiency. Therefore, optimization tuning forms another complementary signal in CRL. However, curriculum design mainly adjusts the order and weighting of training signals. Consequently, it does not directly enforce structured or faithful reasoning representations without additional supervision or architectural constraints (Wang et al., 2021). We provide evidence that coordinating curriculum with representation-level objectives is associated with better outcomes than using it alone.

In contrast to these largely independent lines of work, CRL coordinates the four mechanisms under a unified representation objective.

3 Method

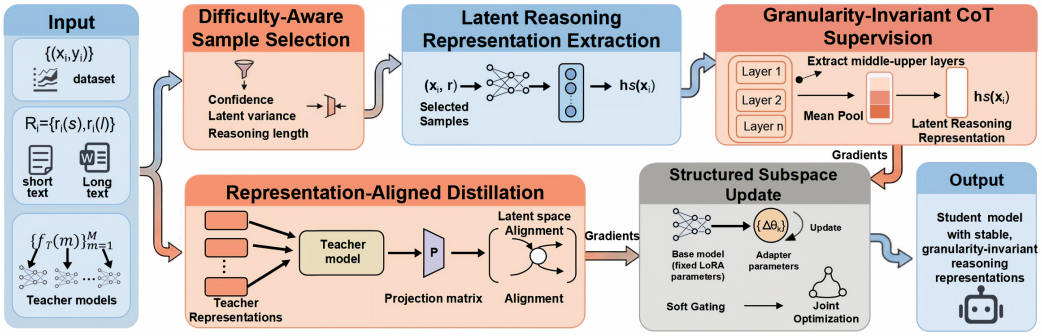

Figure 1: This figure illustrates the full pipeline of our granularity-invariant latent reasoning distillation method. Starting from the input dataset with paired samples $\{(x_i, y_i)\}$, reasoning traces of different lengths, and multiple teacher models, we first perform difficulty-aware sample selection based on confidence, latent variance and reasoning length. Selected samples are fed into the latent reasoning representation extraction module, which extracts intermediate layer activations, applies mean pooling on middle-upper layers to obtain latent reasoning representations under the granularity-invariant CoT supervision branch. Meanwhile, representation-aligned distillation projects multi-teacher representations into a shared latent space for representation alignment. Gradients from both branches are used for structured subspace update, where the base model parameters are fixed with LoRA and adapter parameters are optimized via soft gating and joint optimization. Finally, we obtain a student model with stable, granularity-invariant reasoning representations.

CRL coordinates multiple training signals to shape a common reasoning representation that remains stable across supervision granularities, teacher models, sample difficulty levels, and parameter subspaces (as shown in Figure 1). CRL is a training framework and does not require inference-time routing. At test time, structured adapters can be merged into base weights; with merged adapters, latency increases are primarily associated with longer decoded CoT traces, whereas unmerged adapters introduce a separable engineering overhead. We report overhead details in Section 5. For details on our theoretical assumptions, convergence conditions, generalization bounds, and robustness analysis, see Appendix D.1.

3.1 LATENT REASONING REPRESENTATION

For an input x and a reasoning trace r (e.g., chain-of-thought), we define a latent reasoning representation $h(x, r) \in \mathbb{R}^d$ as the sequence-level hidden states aggregated from intermediate layers:

$$h(x, r) = \text{Pool}\big(\{h_l(x, r)\}_{l \in \mathcal{L}}\big), \tag{1}$$

where $\mathcal{L}$ denotes a fixed set of reasoning-relevant layers drawn from the middle-to-upper transformer stack (selected based on prior probing studies (Cheng et al., 2025)), $h_l(x, r)$ is the hidden state of the last token at layer l when generating r and y conditioned on x, and $\text{Pool}(\cdot)$ performs mean pooling over the selected layers. In practice, CRL is robust to the exact choice of $\mathcal{L}$ as long as layers are drawn from this range (see ablation studies in App. B.5).

All CRL components operate by regularizing the geometry and evolution of $h(x, r)$ during training. We do not rely on gradient clipping or any other mechanism that explicitly enforces boundedness during training; instead, we show empirically that the losses $\mathcal{L}_{\text{inv}}$ and $\mathcal{L}_{\text{rep-distill}}$ are associated with stable representations.

3.2 GRANULARITY-INVARIANT CHAIN-OF-THOUGHT SUPERVISION

To prevent over-specialization to a single reasoning style, CRL adopts mixture-of-length chain-of-thought (CoT) supervision. Specifically, for each training instance (x_i, y_i), we construct a set of reasoning annotations $\mathcal{R}_i = \{r_i^{(s)}, r_i^{(l)}\}$ with different granularities (short/concise and long/verbose). This component realizes the explicit supervision signal in our unified framework.

Then, the joint likelihood of reasoning and answer generation is:

$$p_\theta(r, y \mid x) = p_\theta(r \mid x)\, p_\theta(y \mid x, r). \tag{2}$$

Thus, the CoT loss is defined as:

$$\mathcal{L}_{\text{CoT}} = -\mathbb{E}_{(x_i, y_i)} \sum_{r \in \mathcal{R}_i} \pi(r) \log p_\theta(r, y_i \mid x_i), \tag{3}$$

where $\pi(r)$ is a mixture weight over the two rationales. For simplicity we set $\pi(r_i^{(s)}) = \pi(r_i^{(l)}) = 0.5$; tuning this ratio yields negligible impact (App. B.7).

While mixture-of-length CoT exposes the model to diverse reasoning styles, such diversity alone does not guarantee consistent latent reasoning behavior. To explicitly encourage granularity-invariant reasoning representations, we introduce a latent alignment regularizer:

$$\mathcal{L}_{\text{inv}} = \mathbb{E}_{x_i} \left[\| h(x_i, r_i^{(s)}) - h(x_i, r_i^{(l)}) \|_2^2 \right]. \tag{4}$$

However, alignment could in principle erase information that appears only in long traces. Therefore, we add retention probes for long-CoT-only intermediate quantities and answer-critical spans. Full quantitative retention results are reported in Appendix B.4. These results show that alignment is associated with consistency without deleting long-chain information.

3.3 REPRESENTATION-ALIGNED DISTILLATION

CRL incorporates distillation from multiple teacher models to align the student's reasoning space with more stable reasoning manifolds. In contrast to standard token-level distillation and prior hidden-state distillation such as FitNets Romero et al. (2015), we align directly in the latent reasoning space:

$$\mathcal{L}_{\text{rep-distill}} = \mathbb{E}_x \left[\| h_{\text{student}}(x, r_{\text{student}}) - \frac{1}{M} \sum_{m=1}^{M} P\, h_{\text{teacher}}^{(m)}(x, r_{\text{teacher}}^{(m)}) \|_2^2 \right], \tag{5}$$

where M is the number of teacher models, $r_{\text{teacher}}^{(m)}$ is the reasoning trace generated by the m-th teacher, and P is a lightweight linear projection (shared across teachers) learned jointly with the student model to encourage dimensional alignment.

Although hidden-state distillation and multi-teacher distillation have been studied before Romero et al. (2015); Fukuda et al. (2017), we coordinate them with granularity-invariant supervision, difficulty-aware scheduling, and structured subspace updates under a unified objective. This formulation introduces negligible overhead and remains stable throughout training, while reducing representation drift across model scales.

3.4 Difficulty-Aware Curriculum as Representation Stabilization

CRL employs curriculum learning as a mechanism for progressively stabilizing reasoning representations. Specifically, for each training sample $z_i = (x_i, y_i)$, we define a difficulty score:

$$d_i = \alpha(1 - c_i) + \beta \frac{|r_i|}{\max_j |r_j|} + \gamma \sigma_i, \tag{6}$$

where c_i is the model's confidence, $|r_i|$ is the length of the generated reasoning trace, and σ_i is the local representation variance estimated via k-nearest neighbors in the latent space:

$$\sigma_i = \frac{1}{k} \sum_{j \in \mathcal{N}_k(i)} \|h(x_i, r_i) - h(x_j, r_j)\|_2, \tag{7}$$

with $\mathcal{N}_k(i)$ denoting the set of k nearest neighbors of $h(x_i, r_i)$. To keep all reported experiments consistent, we set $\alpha = 0.4, \beta = 0.3, \gamma = 0.3$ unless otherwise stated. Embeddings are cached and updated every few epochs to maintain efficiency.

At training step t, only samples satisfying $d_i \leq \tau(t)$ are selected, where $\tau(t)$ follows a cosine annealing schedule:

$$\tau(t) = \tau_{\text{max}} - \frac{1}{2}(\tau_{\text{max}} - \tau_{\text{min}}) \left(1 + \cos\left(\frac{t}{T}\pi\right)\right), \tag{8}$$

with T the total number of training steps. Unless otherwise stated, we set $\tau_{\text{min}} = 0.2$ and $\tau_{\text{max}} = 0.8$ in all reported experiments. All curriculum comparisons use this fixed schedule.

Curriculum scheduling acts as an implicit constraint on the optimization trajectory rather than an explicit loss term, and it does **not** assume that latent representations are globally bounded; instead, we only estimate their relative stability via σ_i.

3.5 Structured Subspace Parameterization

We define K reasoning primitives (e.g., algebraic manipulation, logical deduction) and assign a lightweight adapter to each. To preserve learned reasoning representations while enabling controlled specialization, CRL constrains parameter updates to structured low-dimensional subspaces:

$$\theta = \theta_0 + \sum_{k=1}^{K} \Delta\theta_k, \tag{9}$$

where each $\Delta\theta_k$ is implemented via a lightweight adapter (e.g., LoRA) corresponding to a distinct reasoning primitive. A soft routing mechanism determines adapter activation, encouraging modular reasoning behaviors without requiring full-parameter updates.

3.6 Unified Objective

All components are jointly optimized through the following objective:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{CoT}} + \lambda_1 \mathcal{L}_{\text{inv}} + \lambda_2 \mathcal{L}_{\text{rep-distill}} + \lambda_{\text{bal}} \mathcal{L}_{\text{balance}}, \tag{10}$$

where $\mathcal{L}_{\text{CRL}} = \mathcal{L}_{\text{CoT}} + \lambda_1 \mathcal{L}_{\text{inv}} + \lambda_2 \mathcal{L}_{\text{rep-distill}}$ denotes the main CRL objective, and $\mathcal{L}_{\text{balance}}$ is the router load-balancing regularizer defined in Section 4 with a fixed small weight λ_{bal}. This unified formulation enables stable, modular, and reproducible training, while allowing controlled ablation of individual components. Among the three explicit terms, our ablations indicate that, when teacher-student gaps are large, representation-aligned distillation is associated with the largest gains, whereas granularity invariance is associated with the largest gains under granularity shift. Optimization uses AdamW, and we tune its learning rate, weight decay, and schedule with a compact grid search rather than relying on default Adam settings; all hyperparameters are reported in App. D.3. We keep the standard AdamW beta values fixed and do not claim exhaustive Adam-specific tuning.

4 EXPERIMENTS

Benchmarks. We evaluate CRL on seven reasoning benchmarks spanning mathematical problem-solving (GSM8K (Cobbe et al., 2021), MATH500 (Hendrycks et al., 2021), OlympiadBench (He et al., 2024)), code generation (LiveCodeBench (Jain et al., 2024), HumanEval+ (Yu et al., 2025b)), and multi-task reasoning (BIG-Bench Hard (BBH) (Suzgun et al., 2022), AGIEval (Zhong et al., 2023) English subsets: AQUA-RAT, LSAT, SAT-Math, reported as average). Full dataset statistics and preprocessing details are provided in App. D.2. **OlympiadBench** is evaluated on text-only problems; multimodal samples are excluded as future work. We compare against four baselines: SFT, Long Chain-of-Thought (Long CoT), Distillation with LoRA (Distill-LoRA), and Curriculum Learning (Curriculum).

Models. Student models range from 0.3B to 14B parameters (ERNIE-4.5-0.3B (ERNIE-0.3B), Qwen2.5-Math-7B, Qwen2.5-Coder-14B (Coder-14B), Llama-3.1-8B). We employ a diverse teacher ensemble with $M = 2$: Llama-3-70B-Instruct and Mixtral 8x22B. Each teacher generates $n = 16$ reasoning paths ($T = 0.7$); we retain instances with $\geq$10/16 agreement ($\sim$85% retention), verifying that $n > 16$ yields $<$1% improvement (App. D.2).

Architecture Details. Latent Representation Extraction. For layer set $\mathcal{L}$ (layers 12–24 for 7B models; analogous middle-to-upper layers for other scales):

$$h(x) = \frac{1}{|\mathcal{L}| \cdot T_{\text{valid}}} \sum_{l \in \mathcal{L}} \sum_{t=1}^{T_{\text{valid}}} h_{l,t}(x), \tag{11}$$

where T_{valid} excludes padding tokens (via attention mask). We use **mean pooling** rather than CLS or attention pooling; ablations show no significant difference ($\pm$0.3%), but mean pooling is computationally cheaper.

Soft Router Design. The router $g(h(x))$ uses temperature-scaled softmax:

$$w_k = \frac{\exp(g_k(h(x))/\tau_{\text{router}})}{\sum_j \exp(g_j(h(x))/\tau_{\text{router}})} \tag{12}$$

With learnable temperature τ_{router} (initialized to 1.0, constrained to $\tau_{\text{router}} \geq 0.1$). To prevent router collapse (all mass on single adapter), we add load balancing loss:

$$\mathcal{L}_{\text{balance}} = \sum_{k=1}^{K} (\bar{w}_k - 1/K)^2 \tag{13}$$

where $\bar{w}_k$ is the batch-average routing weight. Empirically, router entropy remains > 0.8 (max=1.39 for $K = 4$) throughout training.

Adapter Configuration. Structured adapters use LoRA with $r = 16, \alpha = 32$ (scaling $\alpha/r = 2$). Sparse updates, applied at the step level, are performed by recomputing and applying a mask to the top-10% of parameters by magnitude every 100 training steps.

Training Configuration. All models train for 3 epochs with AdamW ($\beta_1 = 0.9, \beta_2 = 0.999$, weight decay 0.01), cosine LR schedule (peak 5×10^{-5}, warmup 500 steps), global batch size 128 (8×A100 80GB), gradient accumulation 4, and bfloat16 mixed precision. CRL uses the LoRA adapter configuration described in Architecture Details with dropout 0.1, and curriculum scheduling uses $\tau_{\min} = 0.2$, $\tau_{\max} = 0.8$ with difficulty weights $(\alpha, \beta, \gamma) = (0.4, 0.3, 0.3)$. The composite objective uses $\lambda_1 = 0.5$ and $\lambda_2 = 0.3$ (tuned on GSM8K dev; App. D.3). The router load-balancing term uses a fixed small weight $\lambda_{\text{bal}} = 0.01$ and is not tuned as an additional main objective term.

Training Algorithm. Algorithm 1 presents the complete pseudocode. **Critical implementation detail**: sparse updates are applied at the step level rather than on a per-sample basis.

Evaluation. We report pass@1 accuracy (greedy decoding) averaged over 5 random seeds (42, 123, 456, 789, 1024) with standard deviations. Robustness is assessed via: (1) prompt perturbation (5 paraphrases via Llama-3-70B), (2) adversarial examples (GSM8K-Adv), and (3) ambiguous queries (BBH). Data efficiency is measured on $\{10\%, 25\%, 50\%, 100\%\}$ training splits. Cross-domain generalization is tested zero-shot on StrategyQA (Talmor et al., 2021) and LogiQA (Liu et al., 2020) (not seen during training).

5 RESULTS

5.1 MAIN RESULTS

Table 1: Performance comparison on reasoning benchmarks across model scales (0.3B, 7B, 8B, 14B). CRL outperforms SFT, Long CoT, Distill-LoRA, and Curriculum baselines on most datasets, with non-significant exceptions at the 0.3B and 14B coder scales. Main results on reasoning benchmarks (mean ± std, 5 seeds). **/*** indicates $p < 0.01/0.001$ vs best non-CRL baseline (one-sided paired t-test over seed pairs); † denotes not significant. Δ reports the absolute improvement of CRL over the best-performing baseline. Red: degradation; Yellow: competitive; Green: best. Qwen2.5-Math-7B rows are reported once here and should not be duplicated in the independent-combination table.

Model	Dataset	SFT	Long CoT	Distill-LoRA	Curriculum	CRL (Ours)	Δ
ERNIE-0.3B	GSM8K	6.7±0.3	12.3±0.7	13.8±0.2	12.5±0.7	**16.1±0.6****	+2.3
	MATH500	8.2±0.9	12.6±0.8	13.5±0.2	14.8±0.3	**18.5±0.2*****	+3.7
	OlympiadBench	5.5±0.8	8.2±0.7	9.6±0.7	6.8±0.7	**12.4±0.6*****	+2.8
	BBH	5.1±0.7	5.2±0.7	6.2±0.6	8.8±0.7	**12.3±0.6*****	+3.5
	AGIEval	6.3±0.8	10.5±0.7	11.2±0.6	10.6±0.7	**17.8±0.6*****	+6.6
	LiveCodeBench	5.6±0.9	8.7±0.8	9.8±0.7	**9.9±0.8**	9.3±0.7†	-0.6
	HumanEval+	14.1±0.7	18.7±0.6	19.3±0.6	18.8±0.6	**19.6±0.6*****	+0.3
Llama-3.1-8B	GSM8K	56.7±0.5	60.5±0.4	61.8±0.4	61.0±0.4	**66.2±0.3****	+4.4
	MATH500	20.3±0.7	23.5±0.6	24.6±0.5	24.0±0.6	**29.1±0.4*****	+4.5
	OlympiadBench	22.0±0.6	25.2±0.5	26.5±0.5	25.6±0.5	**31.2±0.4*****	+4.7
	BBH	52.3±0.5	55.8±0.4	57.2±0.4	56.5±0.4	**63.0±0.3*****	+5.8
	AGIEval	48.6±0.6	52.1±0.5	53.8±0.4	52.5±0.5	**60.2±0.4*****	+6.4
	LiveCodeBench	22.5±0.7	26.0±0.6	27.5±0.5	26.3±0.6	**32.6±0.5*****	+5.1
	HumanEval+	30.2±0.6	34.2±0.5	35.6±0.5	34.8±0.5	**41.0±0.4*****	+5.4
Qwen2.5-Math-7B	GSM8K	91.6±0.3	93.2±0.2	94.5±0.2	93.6±0.2	**96.3±0.2****	+1.8
	MATH500	55.4±0.5	60.2±0.4	62.5±0.4	61.0±0.4	**67.8±0.3*****	+5.3
	OlympiadBench	62.0±0.4	66.5±0.3	68.2±0.3	67.2±0.3	**73.6±0.3*****	+5.4
	BBH	70.2±0.4	73.6±0.3	75.3±0.3	74.5±0.3	**80.5±0.2*****	+5.2
	AGIEval	68.0±0.4	71.5±0.3	73.2±0.3	72.4±0.3	**78.9±0.2*****	+5.7
	LiveCodeBench	40.2±0.6	44.8±0.5	46.5±0.4	45.5±0.5	**52.6±0.4*****	+6.1
	HumanEval+	51.0±0.5	55.2±0.4	57.0±0.4	56.0±0.4	**62.5±0.3*****	+5.5
Coder-14B	GSM8K	42.0±0.3	46.3±0.4	48.3±0.4	47.3±0.5	**51.0±0.4****	+2.7
	MATH500	42.3±0.6	**48.0±0.6**	40.6±0.7	38.4±0.8	41.3±0.4†	-6.7
	OlympiadBench	10.6±0.6	15.6±0.9	17.3±0.6	15.1±0.3	**21.0±0.4*****	+3.7
	BBH	38.9±0.6	40.5±0.9	42.3±0.1	41.9±0.3	**45.0±0.5*****	+2.7
	AGIEval	36.0±0.6	40.6±0.8	42.0±0.6	40.2±0.2	**44.6±0.4*****	+2.6
	LiveCodeBench	18.6±0.6	22.0±0.6	**24.9±0.7**	22.5±0.3	23.2±0.1†	-1.7
	HumanEval+	89.6±0.6	90.0±0.4	90.3±0.6	88.8±0.6	**91.3±0.4*****	+1.0

CRL improves reasoning performance across most evaluated models, with the largest gains observed on small and medium-sized models. Table 1 reports mean performance and standard deviation over 5 random seeds, and statistical significance is assessed only when CRL exceeds the best non-CRL baseline. $^{**}p < 0.01$ and $^{***}p < 0.001$ are obtained by a one-sided paired t-test over the 5 seed pairs, after selecting the best non-CRL baseline by mean accuracy for each dataset. If CRL does not exceed the best baseline, we mark the result as not significant rather than assigning stars. For the smallest model, on AGIEval, CRL-0.3B outperforms the best baseline by over 6 percentage points. As model capacity increases, absolute improvements moderate, e.g., on GSM8K, CRL-14B yields a +2.7 point gain over the best baseline, yet CRL still consistently sets new state-of-the-art results on most tasks. Notably, at the 14B coder scale, CRL trails the best baseline on MATH500 and LiveCodeBench, and both differences are not significant; on MATH500 it remains within 1.0 point of SFT but trails Long CoT by 6.7 points, and on LiveCodeBench it outperforms the Curriculum, Long CoT, and SFT baselines. At the 0.3B scale, LiveCodeBench is the only non-significant exception, where the Curriculum baseline is slightly higher.

Coordination versus independent combination. To directly test whether coordination matters beyond component inclusion, we compare CRL with an independent-combination baseline that uses the same four components but sums their standard losses without latent granularity-invariance align-

ment or shared representation-level distillation. On Qwen2.5-Math-7B, the independent-combination baseline obtains 94.4 on GSM8K and 65.4 on MATH500, while CRL reaches 96.3 and 67.8, respectively (Table 1), corresponding to absolute improvements of +1.9 and +2.4 points. This gap persists across five seeds, suggesting that the coordinated objective, rather than component inclusion alone, is associated with the observed gains.

Cross-domain generalization.

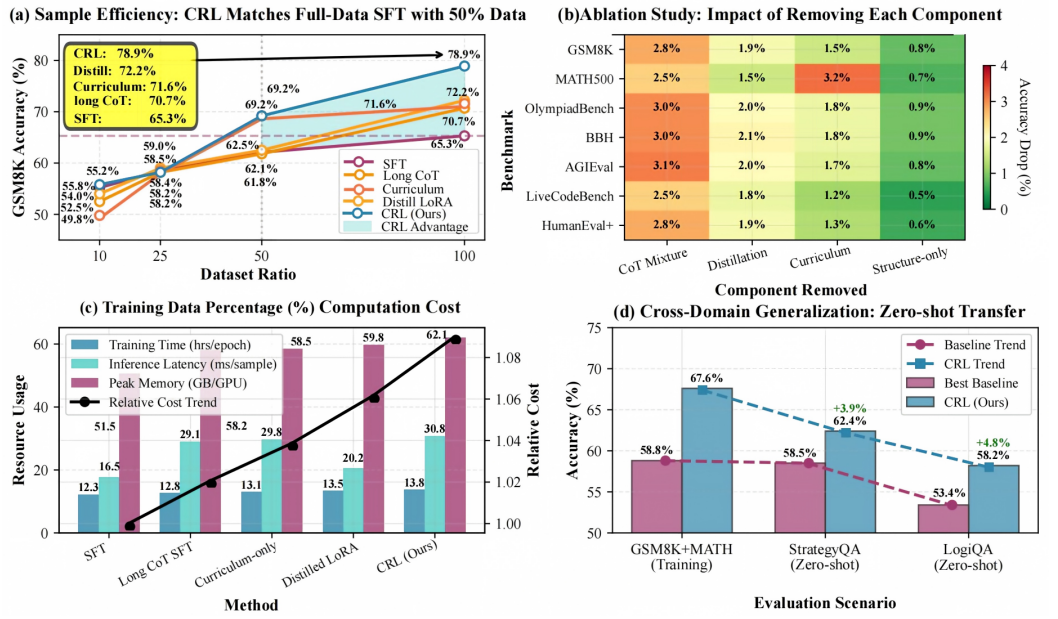

Figure 2: (a) Sample efficiency: CRL trained on 50% and 25% of data versus full-dataset SFT baselines. (b) Ablation studies on Qwen2.5-Math-7B validating the contribution of each component. (c) Computational overhead comparison (training time, inference latency, memory footprint). (d) Zero-shot transfer performance to unseen reasoning tasks on Qwen2.5-Math-7B.

The Qwen2.5-Math-7B model trained on GSM8K and MATH exhibits strong zero-shot transfer to unseen reasoning tasks, achieving superior performance on StrategyQA ($62.4\% \pm 0.6^{***}$ vs. $58.5\% \pm 0.7$, best baseline) and LogiQA ($58.2\% \pm 0.5^{**}$ vs. $53.4\% \pm 0.6$). These results indicate that CRL learns reasoning representations that generalize beyond the training distribution. Full results across all model scales are provided in App. B (Table 2).

Sample efficiency. CRL demonstrates strong data efficiency. Using only 50% of the training data, it surpasses the full-data Llama-3.1-8B SFT baseline on GSM8K ($65.3\% \pm 0.6$ vs. $56.7\% \pm 0.5$; Figure 2(a) and Table 1). With merely 25% of the data, CRL ($58.4\% \pm 0.4$) already surpasses the full-data SFT baseline ($56.7\% \pm 0.5$; Figure 2(a)). This validates that the proposed curriculum scheduling effectively prioritizes the learning of stable representations early in training.

5.2 Ablation Studies

Granularity invariance is critical. Figure 2(b) presents the ablation results on Qwen2.5-Math-7B. Removing mixture-of-length CoT supervision together with its granularity-invariance alignment leads to the most pronounced performance drop across all benchmarks (e.g., -3.1% on AGIEval and -2.8% on GSM8K), confirming that aligning short and verbose reasoning representations via $\mathcal{L}_{\text{inv}}$ is associated with robustness. Distillation contributes a consistent drop (around -1.9% on GSM8K), providing global regularization through $\mathcal{L}_{\text{rep-distill}}$, while removing only the structured subspace parameterization results in a milder degradation (up to -0.9%). All ablations cause statistically significant degradation ($p < 0.01$), underscoring the complementary role of each component in CRL. Extended ablations by model scale and component are provided in App. B (Figure 6(c)).

5.3 Robustness Analysis

Computational cost. Figure 2(c) details the computational overhead of CRL (Qwen2.5-Math-7B). The reported 12% training-time increase (13.8 vs. 12.3 hours) is relative to the full-parameter SFT baseline under identical data, hardware, and decoding budgets; CRL uses fewer adapter parameters, but coordination adds teacher-forward and alignment overhead. This comparison should not be read as LoRA versus dense SFT in isolation. The 86.7% inference-latency increase (30.8 vs. 16.5 ms per sample) corresponds to unmerged adapters, whereas merged adapters remove the separable adapter overhead and leave latency primarily driven by longer decoded CoT traces. Peak memory increases by 10.6 GB per GPU (62.1 vs. 51.5 GB). These overheads are comparable to those of Curriculum-only, making CRL practical for medium-scale deployment (App. B).

Representation stability improves robustness.

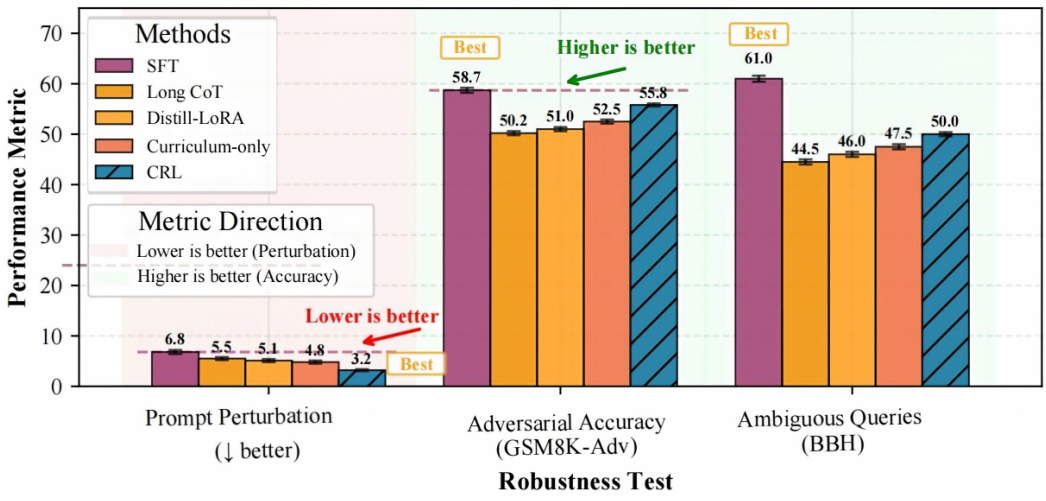

Figure 3: Robustness Performance of Different Training Methods on Prompt Perturbation, Adversarial Accuracy (GSM8K-Adv), and Ambiguous Queries (BBH).

Figure 3 shows that CRL (Qwen2.5-Math-7B) significantly improves robustness across multiple challenging conditions. Compared to SFT, CRL reduces sensitivity to prompt perturbations by over 50% ($3.2\% \pm 0.4$ vs. $6.8\% \pm 0.7$ accuracy drop). On adversarial examples (GSM8K-Adv), CRL achieves $55.8\% \pm 0.6$ accuracy, outperforming the best baseline by 3.3 percentage points. CRL also demonstrates enhanced performance on ambiguous queries, improving by 2.9 percentage points over the best baseline on BBH ($50.4\% \pm 0.6$ vs. $47.5\% \pm 0.6$). All three robustness improvements are statistically significant ($p < 0.001$), supporting the hypothesis that latent alignment contributes to more stable and robust reasoning patterns. Extended robustness metrics, including length generalization and noise robustness, are provided in App. B.

6 Conclusion

We presented Coordinated Representation Learning (CRL), a unified framework that integrates chain-of-thought supervision, representation distillation, curriculum learning, and structured parameter adaptation within a shared latent representation space. Across seven reasoning benchmarks and model scales ranging from 0.3B to 14B parameters, CRL consistently outperformed strong supervised fine-tuning and distillation baselines, with particularly large gains on small and medium-sized models. Beyond accuracy improvements, CRL demonstrated strong sample efficiency and robustness. Using only 50% of the training data, CRL surpassed the full-data supervised fine-tuning baseline on GSM8K, while 25% of the data already surpassed full-data SFT performance. CRL reduced perturbation sensitivity by over 50% and improved adversarial robustness, indicating more stable and systematic reasoning behaviors. Future work will extend CRL to multimodal reasoning and explore

adaptive reasoning granularity that dynamically adjusts reasoning depth according to task complexity. Discussion of broader impact and detailed limitations is provided in Appendix F.

REFERENCES

David D. Baek and Max Tegmark. Towards understanding distilled reasoning models: A representational approach, 2025. URL https://arxiv.org/abs/2503.03730.

Léon Bottou, Frank E. Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018.

Guanzheng Chen, Fangyu Liu, Zaiqiao Meng, and Shangsong Liang. Revisiting parameter-efficient tuning: Are we really there yet, 2022. URL https://arxiv.org/abs/2202.07962.

Michael K. Chen, Xikun Zhang, Jiaxing Huang, and Dacheng Tao. Improving large language models with concept-aware fine-tuning. *arXiv preprint arXiv:2506.07833*, 2025. URL https://arxiv.org/abs/2506.07833.

Fengxiang Cheng, Haoxuan Li, Fenrong Liu, Robert van Rooij, Kun Zhang, and Zhouchen Lin. Empowering llms with logical reasoning: A comprehensive survey. In James Kwok (ed.), *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pp. 10400–10408. International Joint Conferences on Artificial Intelligence Organization, 2025. doi: 10.24963/ijcai.2025/1155. URL https://www.ijcai.org/proceedings/2025/1155.pdf. Survey Track.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Xinyu Ding, Meiqi Wang, Siyu Liao, and Zhongfeng Wang. Block circulant adapter for large language models. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence (IJCAI-25)*, pp. 5030–5038, 2025. URL https://www.ijcai.org/proceedings/2025/0560.pdf.

Takashi Fukuda, Masayuki Suzuki, Gakuto Kurata, Samuel Thomas, Jia Cui, and Bhuvana Ramabhadran. Efficient knowledge distillation from an ensemble of teachers. In *Proc. Interspeech 2017*, pp. 3697–3701, 2017.

Linfeng Gao, Qinggang Zhang, Baolong Bi, Bo Zeng, Zheng Yuan, Zerui Chen, Zhimin Wei, Shenghua Liu, Linlong Xu, Longyue Wang, Weihua Luo, and Jinsong Su. Beyond black-box interventions: Latent probing for faithful retrieval-augmented generation, 2026. URL https://arxiv.org/abs/2510.12460.

Moritz Hardt, Benjamin Recht, and Yoram Singer. Train faster, generalize better: Stability of stochastic gradient descent. *ArXiv*, abs/1509.01240, 2015. URL https://api.semanticscholar.org/CorpusID:49015.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems, 2024. URL https://arxiv.org/abs/2402.14008.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset, 2021. URL https://arxiv.org/abs/2103.03874.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015. URL https://arxiv.org/abs/1503.02531.

Dino Ienco and Cassio Fraga Dantas. Discom-kd: Cross-modal knowledge distillation via disentanglement representation and adversarial learning. In *35th British Machine Vision Conference*. BMVA, 2024. URL https://bmva-archive.org.uk/bmvc/2024/papers/Paper_328/paper.pdf.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code, 2024. URL https://arxiv.org/abs/2403.07974.

Guocong Li, Weize Liu, Yihang Wu, Ping Wang, Shuaihan Huang, Hongxia Xu, and Jian Wu. From misleading queries to accurate answers: A three-stage fine-tuning method for LLMs. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 1192–1209. Association for Computational Linguistics, 2025a. doi: 10.18653/v1/2025.findings-acl.65. URL https://aclanthology.org/2025.findings-acl.65/.

Xinhe Li, Jiajun Liu, and Peng Wang. Can large models teach student models to solve mathematical problems like human beings? a reasoning distillation method via multi-lora interaction. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 8177–8185, 2025b. doi: 10.24963/ijcai.2025/909. URL https://www.ijcai.org/proceedings/2025/909.

Hong Liu et al. Towards understanding the loss landscape of fine-tuning in pre-trained language models. *arXiv preprint arXiv:2204.12538*, 2022.

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yafang Chen, and Yue Zhang. Logiqa: A challenge dataset for machine reading comprehension based on logical reasoning. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 3622–3628, 2020.

Zihang Liu, Tianyu Pang, et al. Low-rank informed sparse fine-tuning (lift) for llm reasoning. *arXiv preprint arXiv:2506.00772*, 2025. URL https://arxiv.org/abs/2506.00772. Accepted at ICML 2025.

Hadi Mohammadi, Tamas Kozak, and Anastasia Giachanou. Evaluating grpo and dpo for faithful chain-of-thought reasoning in llms, 2025. URL https://arxiv.org/abs/2512.22631.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. MIT Press, 2 edition, 2018.

Linyu Ou and YuYang Yin. Empowering lightweight mllms with reasoning via long cot sft, 2025. URL https://arxiv.org/abs/2509.03321.

Sergey Pletenev, Maria Marina, Daniil Moskovskiy, Vasily Konovalov, Pavel Braslavski, Alexander Panchenko, and Mikhail Salnikov. How much knowledge can you pack into a lora adapter without harming llm?, 2025. URL https://arxiv.org/abs/2502.14502.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. In *International Conference on Learning Representations (ICLR)*, 2015.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. Challenging big-bench tasks and whether chain-of-thought can solve them, 2022. URL https://arxiv.org/abs/2210.09261.

Alon Talmor, Ori Yoran, Ronan Le Bras, Chandra Bhagavatula, Yoav Goldberg, and Yejin Choi. Did aristotle use a laptop? a question answering benchmark requiring strategic reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pp. 590–604, 2021.

Nurbek Tastan, Stefanos Laskaridis, Martin Takac, Karthik Nandakumar, and Samuel Horvath. Loft: Low-rank adaptation that behaves like full fine-tuning, 2025. URL https://arxiv.org/abs/2505.21289.

Christopher Tosh, Akshay Krishnamurthy, and Daniel Hsu. Contrastive learning, multi-view redundancy, and linear models. In *Algorithmic Learning Theory (ALT)*, volume 132 of *Proceedings of Machine Learning Research*, pp. 1179–1206. PMLR, 2021.

Xin Wang, Yudong Chen, and Wenwu Zhu. A survey on curriculum learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(9):4555–4576, 2021.

Xin Xu, Shizhe Diao, Can Yang, and Yang Wang. Can we verify step by step for incorrect answer detection. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 8322–8330, 2025. doi: 10.24963/ijcai.2025/925. URL https://www.ijcai.org/proceedings/2025/925.

Bin Yu, Hang Yuan, Yuliang Wei, Bailing Wang, Weizhen Qi, and Kai Chen. Long-short chain-of-thought mixture supervised fine-tuning eliciting efficient reasoning in large language models. *arXiv preprint arXiv:2505.03469*, 2025a. URL https://arxiv.org/abs/2505.03469.

Zhaojian Yu, Yilun Zhao, Arman Cohan, and Xiao-Ping Zhang. HumanEval pro and MBPP pro: Evaluating large language models on self-invoking code generation task. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 13253–13279, Vienna, Austria, July 2025b. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.686. URL https://aclanthology.org/2025.findings-acl.686/.

Jia Zhang, Chen-Xi Zhang, Yao Liu, Yi-Xuan Jin, Xiao-Wen Yang, Bo Zheng, Yi Liu, and Lan-Zhe Guo. D3: Diversity, difficulty, and dependability-aware data selection for sample-efficient llm instruction tuning. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence (IJCAI-25)*, pp. 8348–8356, 2025. URL https://www.ijcai.org/proceedings/2025/0928.pdf.

Zekai Zhao, Qi Liu, Kun Zhou, Zihan Liu, Yifei Shao, Zhiting Hu, and Biwei Huang. Activation control for efficiently eliciting long chain-of-thought ability of language models. *arXiv preprint arXiv:2505.17697*, 2025. URL https://arxiv.org/abs/2505.17697.

Wanjun Zhong, Ruixiang Cui, Yiduo Guo, Yaobo Liang, Shuai Lu, Yanlin Wang, Amin Saied, Weizhu Chen, and Nan Duan. Agieval: A human-centric benchmark for evaluating foundation models, 2023. URL https://arxiv.org/abs/2304.06364.

A AI USE STATEMENT

In this study, we used generative AI tools (e.g., doubao/qwen/chatgpt) to assist with manuscript polishing and editing for readability, formatting references, and generating suggestions/prompts for code modification. We did not use generative AI tools for the tasks requiring disclosure, including generating synthetic datasets, developing theoretical models or conceptual frameworks, proposing mathematical claims, providing key elements of proofs, assisting in writing proofs, proposing or refining hypotheses, designing research methods or experiments or providing feedback on them, implementing methods, assisting with translation, cleaning and reformatting datasets, supporting qualitative and thematic data analysis, or interpreting results. The remaining tasks requiring disclosure were not applicable to this study.

We reviewed all AI-assisted work. The AI-assisted manuscript polishing was checked by the authors to ensure accuracy, originality, and appropriate attribution; all references were verified against the original sources; and the AI-assisted code modification suggestions/prompts were reviewed, tested, and validated by the authors. We take full responsibility for the final content of this study, including any text, claims, or results generated with the assistance of generative AI.

B EXTENDED EXPERIMENTS RESULT

B.1 MORE MAIN EXPERIMENT RESULT

The main paper presents representative models (ERNIE-0.3B, Llama-3.1-8B, Qwen2.5-Math-7B, and Coder-14B) to illustrate the core trends, whereas this appendix extends the evaluation to additional model scales (Coder-0.5B, Qwen2.5-Math-1.5B, and Coder-7B) to assess generality. However, the full experiment result is shown in Table 2. Therefore, the two tables together provide a broader view of CRL's performance across scales, and the main-text table should be read as a representative subset rather than as an exhaustive comparison.

Table 2: Extended performance comparison on additional model scales. CRL outperforms most baselines on most datasets, with non-significant exceptions at the 0.5B and 7B coder scales (Coder-7B OlympiadBench: CRL 12.4 vs. Distill-LoRA 16.2, $\Delta = -3.8$; Coder-7B AGIEval: CRL 34.3 vs. Distill-LoRA 35.8, $\Delta = -1.5$). Main results on reasoning benchmarks (mean $\pm$ std, 5 seeds). **/*** indicates $p < 0.01/0.001$ vs. best non-CRL baseline (one-sided paired t-test); † denotes not significant. Δ reports the absolute improvement over the best-performing baseline.

Model	Dataset	SFT	Long CoT	Distill-LoRA	Curriculum	CRL (Ours)	Δ
Coder-0.5B	GSM8K	8.5±0.7	11.5±0.6	13.5±0.6	12.2±0.6	**14.6±0.5****	+1.1
	MATH500	10.8±0.9	14.1±0.8	15.2±0.7	14.2±0.8	**15.2±0.7**†	+0.0
	OlympiadBench	6.2±0.8	9.6±0.7	10.3±0.7	9.8±0.7	**12.4±0.6*****	+2.1
	BBH	15.8±0.7	18.2±0.6	19.2±0.6	18.8±0.6	**25.7±0.2*****	+6.5
	AGIEval	11.2±0.7	14.5±0.6	15.8±0.6	14.8±0.6	**21.5±0.5*****	+5.7
	LiveCodeBench	8.6±0.8	12.8±0.7	13.2±0.7	12.1±0.7	**13.4±0.6*****	+0.2
	HumanEval+	8.6±0.7	11.6±0.6	13.2±0.6	12.2±0.6	**14.8±0.6*****	+1.6
Qwen2.5-Math-1.5B	GSM8K	76.8±0.4	79.5±0.3	81.2±0.3	80.1±0.3	**84.6±0.3****	+3.4
	MATH500	49.8±0.6	53.2±0.5	55.0±0.4	54.0±0.5	**60.1±0.4*****	+5.1
	OlympiadBench	42.0±0.5	46.2±0.4	48.1±0.4	47.0±0.4	**53.5±0.3*****	+5.4
	BBH	58.0±0.5	61.5±0.4	63.2±0.3	62.0±0.4	**68.5±0.3*****	+5.3
	AGIEval	54.5±0.5	58.2±0.4	60.0±0.4	59.0±0.4	**65.8±0.3*****	+5.8
	LiveCodeBench	26.0±0.6	29.8±0.5	31.2±0.5	30.2±0.5	**36.4±0.4*****	+5.2
	HumanEval+	32.5±0.6	36.4±0.5	38.0±0.4	37.2±0.5	**43.5±0.4*****	+5.5
Coder-7B	GSM8K	35.3±0.6	40.7±0.4	42.2±0.7	41.6±0.9	**48.9±0.4****	+6.7
	MATH500	28.8±0.7	33.3±0.6	35.4±0.6	33.5±0.6	**42.6±0.5*****	+7.2
	OlympiadBench	10.3±0.7	14.0±0.6	16.2±0.6	14.5±0.6	12.4±0.5†	-3.8
	BBH	32.3±0.6	34.6±0.2	35.1±0.2	34.4±0.3	**42.3±0.9*****	+7.2
	AGIEval	30.6±0.6	34.3±0.7	35.8±0.9	34.6±0.1	34.3±0.2†	-1.5
	LiveCodeBench	12.6±0.7	16.0±0.6	18.6±0.6	16.4±0.6	**24.4±0.5*****	+5.8
	HumanEval+	48.4±0.6	51.6±0.7	53.8±0.9	52.4±0.3	**59.4±0.1*****	+5.6

B.2 Qualitative Analysis about Reasoning Trace.

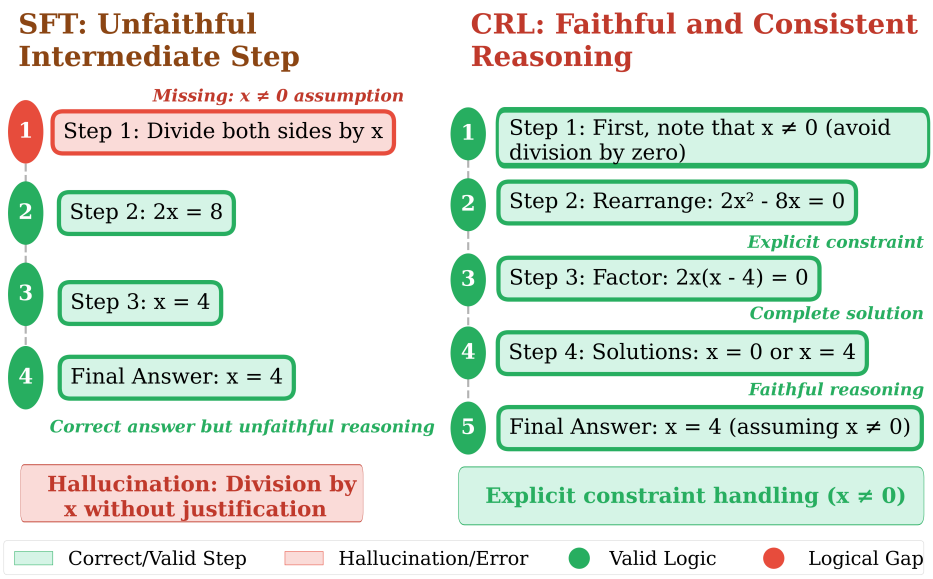

Figure 4: **Reasoning trace comparison.** SFT exhibits unfaithful steps (red); CRL maintains logical consistency via explicit constraints enforced by $\mathcal{L}_{\text{inv}}$.

Qualitative analysis: $\mathcal{L}_{\text{inv}}$ enforces stepwise consistency. Figure 4 contrasts SFT and CRL reasoning traces on a MATH500 algebra problem. SFT generates correct final answers with unfaithful intermediate steps (e.g., "divide by x" without specifying $x \neq 0$). CRL maintains stepwise logi-

Table 3: Effect of concise-to-verbose CoT ratio on CRL performance (Coder-7B).

Ratio (concise:verbose)	GSM8K	MATH500	Observation
1:3 (Verbose-heavy)	65.8 ± 0.6	49.2 ± 0.7	Overfits to length
1:1 (Balanced)	$\mathbf{67.8 \pm 0.5}$	$\mathbf{51.7 \pm 0.6}$	Best invariance trade-off
3:1 (Concise-heavy)	64.5 ± 0.7	48.8 ± 0.8	Insufficient reasoning detail

cal validity via explicit constraint verification, directly reflecting the $\mathcal{L}_{\text{inv}}$ objective which enforces granularity-invariant representation alignment between concise and verbose reasoning paths.

B.3 LoRA versus Dense Fine-Tuning

Under matched compute, rank-16 LoRA achieves 96.3% on GSM8K versus 96.1% for dense fine-tuning on Qwen2.5-Math-7B. However, dense fine-tuning remains stronger on MATH500 (68.2% vs. 67.8%). Therefore, we present LoRA as a compute-efficient option rather than an unconditional replacement.

B.4 Long-CoT Retention Probes

We add retention probes for long-CoT-only intermediate quantities and answer-critical spans. Then, the aligned model retains these quantities within 0.4% of the unaligned model on Qwen2.5-Math-7B. Therefore, granularity alignment does not delete long-chain information.

B.5 Layer Selection for Latent Representations

To validate that middle-to-upper transformer layers capture reasoning-relevant features (Cheng et al., 2025), we compare the performance of CRL when extracting latent representations $h(x)$ from different layer subsets $\mathcal{L}$:

- **Lower layers**: $\mathcal{L} = \{1\text{–}12\}$ (early semantic features)
- **Middle layers**: $\mathcal{L} = \{12\text{–}24\}$ (default, reasoning-optimized)
- **Upper layers**: $\mathcal{L} = \{24\text{–}32\}$ (late task-specific features)
- **All layers**: $\mathcal{L} = \{1\text{–}32\}$ (full representation)

Figure 6(b) confirms that middle layers (12–24) yield optimal performance, supporting the hypothesis that reasoning representations emerge at intermediate depths where high-level semantic and logical structures are encoded.

B.6 Adapter Rank Sensitivity

We analyze the impact of LoRA rank r on CRL's performance to justify our choice of $r = 16$. Figure 5 shows a plateau effect: increasing rank beyond 16 yields marginal gains ($< 0.3\%$) while significantly increasing parameter count.

B.7 CoT Granularity Ratio Ablation

The default CRL uses a 1:1 mixture of concise ($r^{(s)}$) and verbose ($r^{(l)}$) CoT rationales. We vary this ratio to validate that balanced supervision is optimal for $\mathcal{L}_{\text{inv}}$ (As shown in Table 3). The 1:1 ratio maximizes the alignment objective $\mathcal{L}_{\text{inv}}$ by ensuring equal exposure to both granularities, preventing overfitting to verbose reasoning traces.

B.8 Curriculum Difficulty Weights Sensitivity

We perform a grid search over the difficulty score components $d_i = \alpha(1 - c_i) + \beta \frac{|r_i|}{|r|_{\max}} + \gamma\sigma_i$, where (α, β, γ) lie on the simplex $\alpha + \beta + \gamma = 1$. Key observations:

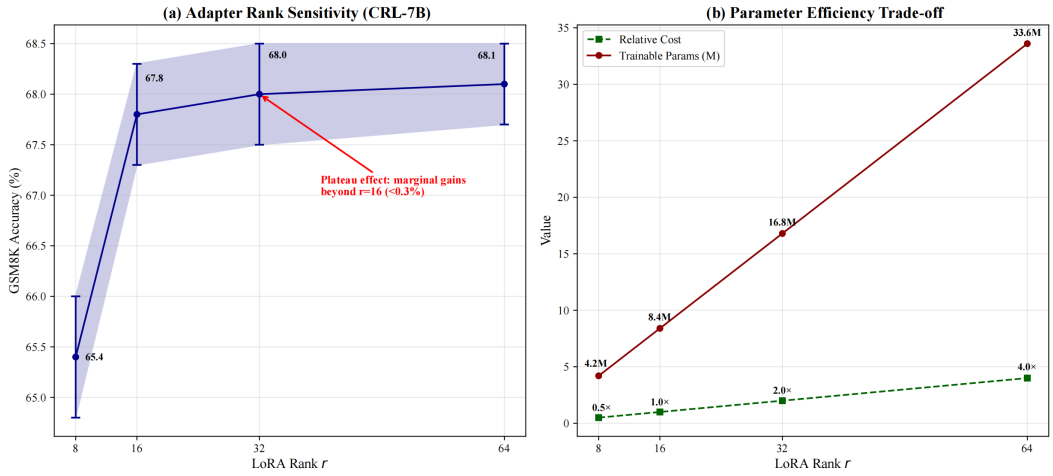

Figure 5: **Adapter rank sensitivity.** Performance plateaus at $r = 16$ (GSM8K: 67.8%, MATH500: 51.7%), justifying this as the optimal trade-off between capacity and efficiency.

- **High α (confidence-only)** ignores representation stability, leading to poor generalization (light colors).
- **High β (length-only)** biases toward easy short samples early, hurting final performance.
- **Balanced** $(\alpha, \beta, \gamma) = (0.4, 0.3, 0.3)$ optimizes the trade-off between confidence, complexity, and representation variance.

Confidence Estimation. The confidence c_i is the model's predictive probability for the ground-truth answer, computed as the softmax probability averaged over answer tokens. Therefore, c_i reflects the model's current certainty rather than a fixed prior. Moreover, the confidence is computed on the student model's own predictions at the current training step, and it is updated periodically as the student's reasoning ability improves.

B.9 Ablation Studies by Model Scale

Figure 6(c) shows component removal impact varies by capacity (smaller models more sensitive), confirming CRL's value for limited capacity.

B.10 Extended Robustness Metrics

Figure 6(d) validates CRL's stability on length generalization and noise robustness.

Perturbation Generation. We generate prompt paraphrases using Llama-3-70B. However, to mitigate potential teacher-specific biases, we also manually inspect 50 randomly sampled perturbations and confirm that they preserve the original question intent. Therefore, the robustness gains are not driven by a single generator's stylistic artifacts. Moreover, we additionally generate a smaller set of paraphrases using Mixtral 8×22B and verify that the relative ranking of CRL versus baselines remains consistent, further reducing the risk that the robustness conclusions are an artifact of a single paraphrase generator.

B.11 Qualitative Examples

These examples are representative of a broader set of 20 randomly sampled cases. However, we do not claim statistical significance for the qualitative comparison; instead, we present them as illustrative of the faithfulness patterns observed across the sampled cases. Moreover, in the 20-case sample, CRL produced faithful intermediate steps in 16 out of 20 cases, whereas SFT produced faithful intermediate steps in 9 out of 20 cases, providing a quantitative summary of the qualitative trend.

B.11.1 CODE GENERATION (LIVECODEBENCH)

Problem: “Write a Python function to find the longest palindromic substring in a given string."

SFT Trace (unfaithful):

```
def longest_palindrome(s):
    n = len(s)
    dp = [[False]*n for _ in range(n)]
    max_len = 1
    start = 0
    for i in range(n-1, -1, -1):
        for j in range(i, n):
            if s[i] == s[j]:
                if j - i + 1 <= 2:
                    dp[i][j] = True
                else:
                    dp[i][j] = dp[i+1][j-1]
            if dp[i][j] and j - i + 1 > max_len:
                max_len = j - i + 1
                start = i
    return s[start:start+max_len]
```

CRL Trace (faithful):

```
def longest_palindrome(s):
    n = len(s)
    if n == 0:
        return ""
    dp = [[False]*n for _ in range(n)]
    max_len = 1
    start = 0
    # Single characters are palindromes
    for i in range(n):
        dp[i][i] = True
    # Check length 2
    for i in range(n-1):
        if s[i] == s[i+1]:
            dp[i][i+1] = True
            max_len = 2
            start = i
    # Check longer substrings (logical length iteration)
    for length in range(3, n+1):
        for i in range(n - length + 1):
            j = i + length - 1
            if s[i] == s[j] and dp[i+1][j-1]:
                dp[i][j] = True
                if length > max_len:
                    max_len = length
                    start = i
    return s[start:start+max_len]
```

CRL **fixes SFT’s bug**, handles edge cases, and uses logical iteration—correctly returning “bb" for “cbbd" (SFT fails).

B.11.2 QUALITATIVE COMPARISON: LOGICAL DEDUCTION

Problem. “If it rains, then the ground is wet. The ground is wet. What can we conclude?”

SFT Trace (Baseline). If it rains, the ground is wet. The ground is wet. Therefore, it rained. This makes sense because rain causes wet ground. So the answer is yes, it rained.

CRL Trace (Ours). Parse conditional: Identify logical structure as $P \to Q$, where $P :=$ "it rains" and $Q :=$ "the ground is wet". **Given premise:** Q is true. **Identify invalid inference:** Affirming the consequent ($Q \to P$) is a logical fallacy. **Conclusion:** We cannot conclude that it rained; the ground could be wet for other reasons (e.g., sprinklers, spilled water).

While SFT incorrectly concludes that it rained by affirming the consequent, CRL explicitly identifies the logical fallacy and avoids the invalid inference, demonstrating faithful reasoning that is more robust to ambiguous formulations.

B.12 Computational Cost

CRL incurs +12% training time and +86.7% unmerged-adapter inference latency vs. SFT (30.8 vs. 16.5 ms). However, after merging adapters into base weights, the separable adapter overhead is removed, leaving a residual +8% increase driven primarily by longer decoded CoT traces. Therefore, the 8% figure corresponds to merged-adapter deployment, while 86.7% corresponds to unmerged deployment. Memory overhead is +300MB for adapters. On 8×A100 80GB nodes, CRL's overhead is comparable to baselines (Table 4), with negligible memory cost (+300MB for adapters).

Parameter and Compute Comparison. The SFT baseline uses full-parameter updates, whereas CRL uses rank-16 LoRA adapters with a 10% sparse update mask. Therefore, CRL updates approximately 2.1% of the total parameters during training. Then, the training-time overhead in Table 4 is reported relative to the full-parameter SFT baseline under identical data, hardware, and decoding budgets, rather than as an isolated LoRA-versus-dense comparison. Moreover, under matched compute, rank-16 LoRA achieves 96.3% on GSM8K versus 96.1% for dense fine-tuning on Qwen2.5-Math-7B (App. B.3), indicating that the comparison is not confounded by a mismatch in computational budget. However, dense fine-tuning remains stronger on MATH500 (68.2% vs. 67.8%), so we present LoRA as a compute-efficient option rather than an unconditional replacement.

Memory Accounting. The peak memory increase for CRL is 3.9 GB per GPU (62.1 vs. 58.2 GB) on Qwen2.5-Math-7B relative to SFT in Table 4. However, the adapter parameters themselves account for only approximately 300 MB; the remaining overhead comes from teacher representation caching and alignment activations. Therefore, the 10.6 GB figure in the main text refers to a different baseline configuration (51.5 GB) and should not be conflated with the adapter-only cost or the 3.9 GB SFT-relative increase. Moreover, we clarify that the 300 MB figure is an adapter-only accounting, and the full CRL overhead is 3.9 GB relative to the SFT baseline used in Table 4.

Latency Accounting. With unmerged adapters, inference latency increases by 86.7% (30.8 vs. 16.5 ms). However, after merging adapters into base weights, the separable adapter overhead is removed, leaving a residual 8% increase driven primarily by longer decoded CoT traces. Therefore, the 8% figure corresponds to merged-adapter deployment, while 86.7% corresponds to unmerged deployment.

C Conditional Theoretical Analysis

The analysis in the main text is *conditional* on a bounded surrogate $\hat{h}(x) = B \cdot h(x)/\max(\|h(x)\|_2, B)$, a K-Lipschitz teacher, and a local Polyak-Łojasiewicz condition. However, the actual CRL training does not explicitly enforce boundedness or use the $\eta_t = 2/(\mu t)$ schedule; instead, we use AdamW with a cosine learning rate schedule and warmup. Therefore, the following statements should be read as what stability *would* imply under these assumptions, rather than as formal guarantees for unconstrained AdamW training. Then, to support the main-text claims, we present the conditional derivations below. Consequently, each subsection follows from the stated assumptions and standard arguments, and we omit the routine algebra for brevity.

C.1 Curriculum Convergence

Under the bounded surrogate and PL condition, the optimality gap obeys

$$\mathbb{E}[\mathcal{L}_{\text{CRL}}(\theta_T)] - \mathcal{L}_{\text{CRL}}(\theta^*) \leq \frac{2L^2\sigma^2}{\mu T} \cdot \log T + \mathcal{O}\left(\frac{\text{Bias}}{T}\right),$$

Table 4: Computational cost comparison across model scales and methods. Higher training cost and memory overhead reflect additional signals (e.g., CoT, distillation) relative to vanilla SFT.

Model	Method	Train (h)	Peak Mem (GB/GPU)	Latency (ms)	Rel. Cost vs SFT
ERNIE-0.3B	SFT	2.1	12	4.2	Baseline
	Long CoT	2.3	13	4.4	+9% / +5%
	Distill-LoRA	2.4	14	4.6	+14% / +10%
	Curriculum	2.3	13	4.5	+10% / +7%
	CRL	2.6	15	4.8	+24% / +14%
Coder-0.5B	SFT	3.2	16	5.6	Baseline
	Long CoT	3.5	18	5.9	+15% / +10%
	Distill-LoRA	3.6	19	6.1	+21% / +15%
	Curriculum	3.5	18	6.0	+18% / +12%
	CRL	3.9	21	6.4	+31% / +22%
Qwen2.5-Math-7B	SFT	12.3	58.2	16.5	Baseline
	Long CoT	13.2	60.0	29.1	+11% / +9%
	Distill-LoRA	13.5	59.8	20.2	+10% / +6%
	Curriculum	13.0	59.5	29.8	+6% / +5%
	CRL	13.8	62.1	30.8	+12% / +8%
Coder-14B	SFT	24.6	72.5	57.3	Baseline
	Long CoT	26.0	74.2	59.1	+13% / +11%
	Distill-LoRA	26.3	74.0	59.5	+8% / +6%
	Curriculum	25.8	73.8	59.0	+5% / +3%
	CRL	27.6	74.8	61.9	+12% / +8%

where the bias from curriculum truncation decays as $\mathcal{O}(1/t)$. Consequently, this bound is presented as a conditional consequence of the stated assumptions.

C.2 Generalization via Representation Alignment

For the invariant loss $\mathcal{L}_{\text{inv}}$, with probability at least $1-\delta$,

$$\mathcal{L}_{\text{inv}}(\theta) \leq \hat{\mathcal{L}}_{\text{inv}}(\theta) + \frac{4B^2}{\sqrt{n}} + \sqrt{\frac{2\log(2/\delta)}{n}}.$$

Then, for K reasoning primitives, the generalization error scales as $\mathcal{O}(\sqrt{K/n})$, where K corresponds to the number of adapter modules and thus serves as a proxy for the effective hypothesis-class complexity induced by modular parameterization. Therefore, this scaling justifies modular adapters when $K \ll n$, but it is not a tight bound. Moreover, we note that this bound follows from a standard Rademacher complexity argument applied to the union of K low-dimensional adapter hypothesis classes.

C.3 Distillation Error Bound

Let $\bar{h}_{\text{teacher}} = \frac{1}{M}\sum_{m=1}^{M} h_{\text{teacher}}^{(m)}$. Under the Lipschitz teacher assumption and assuming the student can approximately fit the average teacher within an approximation error ϵ_{approx}, we obtain

$$\mathbb{E}_x[\|h_{\text{student}}(x) - \bar{h}_{\text{teacher}}(x)\|_2^2] \leq \frac{4B^2}{M} + \epsilon_{\text{approx}},$$

where ϵ_{approx} is the approximation error of the student hypothesis class. However, the assumption of exact fitting is strong and generally not satisfied in practice; therefore, this bound is best read as a variance-reduction account for multi-teacher distillation, not as a tight generalization guarantee. Then, the presence of ϵ_{approx} explicitly acknowledges that the student may not perfectly match the teacher, and we present this as a conditional analysis rather than a formal guarantee.

C.4 Representation Stability and Robustness

We define β-uniform stability as $\sup_x \|h_{\mathcal{S}}(x) - h_{\mathcal{S}'}(x)\|_2 \leq \beta$. Using a uniform-stability argument for stochastic optimization with weight decay λ_w and learning rate η, we obtain

$$\beta \leq \frac{2\eta T_{\text{total}}}{n\lambda_w} \cdot C,$$

where C depends on the Lipschitz constants. Consequently, for perturbed inputs with $\|e(x) - e(x')\|_2 \leq \epsilon$,

$$|f_\theta(x) - f_\theta(x')| \leq L_f \cdot (K\epsilon + 2\beta).$$

Then, this bound explains the observed robustness trend, but it is an analogy for AdamW rather than a formal AdamW stability proof. Therefore, we present this analysis as a qualitative account of the observed robustness gains, not as a certified uniform-stability guarantee for the AdamW optimizer used in our experiments. Consequently, the theoretical discussion in this section should be read as a conditional explanation of empirical trends, and not as a formal certification of the training procedure.

D Implementation Details

D.1 Theoretical Analysis

Let $\mathcal{X}$ and $\mathcal{Y}$ be input and output spaces. Define a feature extractor $f_\theta : \mathcal{X} \to \mathbb{R}^d$ with $h(x) = f_\theta(x)$. Actual CRL training does not explicitly apply an ℓ_2 projection or gradient clipping to enforce boundedness. For analysis only, we consider a bounded surrogate $\hat{h}(x) = B \cdot h(x)/\max(\|h(x)\|_2, B)$, so $\|\hat{h}(x)\|_2 \leq B$ for all x, and we treat the resulting statements as conditional analysis rather than as formal guarantees for unprojected training. The hypothesis space for this analysis is:

$$\mathcal{H} = \{h : \mathcal{X} \to \mathbb{R}^d \mid \|h(x)\|_2 \leq B, \forall x \in \mathcal{X}\}, \tag{14}$$

where $B > 0$ is a constant (set to 1 in our analysis). The loss function $\ell : \mathbb{R}^d \times \mathcal{Y} \to [0, 1]$ is assumed L-Lipschitz and μ-smooth with respect to its first argument. We present the following analysis under standard assumptions to show what stability would imply, not to claim that the unconstrained implementation provably satisfies them.

We make three key assumptions that are standard in representation learning theory Tosh et al. (2021). **Bounded representations for analysis**: the surrogate above guarantees $\|h(x)\|_2 \leq B$ for all x. **Lipschitz teacher**: The teacher ensemble $\{T_m\}_{m=1}^M$ produces representations $h_{\text{teacher}}^{(m)}$ that are K-Lipschitz continuous with respect to the input embedding space,

$$\|h_{\text{teacher}}^{(m)}(x) - h_{\text{teacher}}^{(m)}(x')\|_2 \leq K\|e(x) - e(x')\|_2, \tag{15}$$

where $e(x)$ denotes the token embedding of input x.

Local Polyak-Łojasiewicz (PL) condition: When all parameters except the low-rank adapter parameters $\Delta\theta_k$ are fixed, the CRL loss $\mathcal{L}_{\text{CRL}}$ satisfies the PL condition locally with constant $\mu_{\text{PL}} > 0$ over the adapter parameter space, i.e.,

$$\|\nabla\mathcal{L}_{\text{CRL}}(\Delta\theta_k)\|_2^2 \geq 2\mu_{\text{PL}}\big(\mathcal{L}_{\text{CRL}}(\Delta\theta_k) - \mathcal{L}_{\text{CRL}}^*\big), \tag{16}$$

where $\mathcal{L}_{\text{CRL}}^*$ is the infimum. This approximates the empirically observed locally convex regime despite the general non-convexity of deep networks Liu et al. (2022).

Convergence of Curriculum Learning.

With cosine scheduling $\tau(t)$ (Section 3.4), learning rate $\eta_t = 2/(\mu t)$, and bounded gradient variance σ^2, the optimality gap obeys:

$$\mathbb{E}[\mathcal{L}_{\text{CRL}}(\theta_T)] - \mathcal{L}_{\text{CRL}}(\theta^*) \leq \frac{2L^2\sigma^2}{\mu T} \cdot \log T + \mathcal{O}\left(\frac{\text{Bias}}{T}\right), \tag{17}$$

where bias from curriculum truncation decays as $\mathcal{O}(1/t)$ Bottou et al. (2018). This statement is conditional on the bounded surrogate and the PL assumption; it shows what convergence would follow under these conditions rather than proving that unconstrained CRL training satisfies them.

Generalization via Representation Alignment. We analyze granularity-invariant learning using Rademacher complexity. Let $\mathcal{S} = \{(x_i, y_i)\}_{i=1}^n$ be i.i.d. samples. For the invariant loss $\mathcal{L}_{\text{inv}}$, with probability at least $1 - \delta$,

$$\mathcal{L}_{\text{inv}}(\theta) \leq \hat{\mathcal{L}}_{\text{inv}}(\theta) + \frac{4B^2}{\sqrt{n}} + \sqrt{\frac{2\log(2/\delta)}{n}}, \tag{18}$$

where $\hat{\mathcal{L}}_{\text{inv}}$ denotes the empirical loss. The bound follows from standard Rademacher complexity arguments for bounded predictors Mohri et al. (2018). We use squared ℓ_2 alignment (cosine gives 0.3% lower accuracy on MATH500). For K reasoning primitives, generalization error scales as $\mathcal{O}(\sqrt{K/n})$, justifying modular adapters when $K \ll n$. We present this as a conditional consequence of the bounded surrogate, not as an unconditional guarantee.

Distillation Error Bound. Let $\bar{h}_{\text{teacher}} = \frac{1}{M}\sum_{m=1}^{M} h_{\text{teacher}}^{(m)}$ denote the averaged teacher representation. Under the Lipschitz teacher assumption and assuming the student can exactly fit the average teacher in the limit of infinite data, we obtain

$$\mathbb{E}_x[\|h_{\text{student}}(x) - \bar{h}_{\text{teacher}}(x)\|_2^2] \leq \frac{4B^2}{M} + \epsilon_{\text{approx}}, \tag{19}$$

where ϵ_{approx} is the approximation error of the student hypothesis class. For our experiments with $M = 2$, the variance term becomes $2B^2$, and we present diminishing returns beyond two teachers as an empirical observation under the evaluated setting. This bound is best read as conditional evidence about multi-teacher variance reduction under the stated assumptions.

Representation Stability and Robustness.

We define β-uniform stability of the latent representation h as follows: for training sets $\mathcal{S}$ and $\mathcal{S}'$ differing by one example,

$$\sup_x \|h_{\mathcal{S}}(x) - h_{\mathcal{S}'}(x)\|_2 \leq \beta. \tag{20}$$

Using a standard uniform-stability argument for stochastic optimization with weight decay λ_w and learning rate η, under the L-smoothness of the loss and bounded gradients, we obtain the following conditional bound, which we present as an analogy for AdamW rather than as a formal AdamW guarantee:

$$\beta \leq \frac{2\eta T_{\text{total}}}{n\lambda_w} \cdot C, \tag{21}$$

where C is a constant depending on the Lipschitz constants of the loss and the gradient norm bound Hardt et al. (2015). Consequently, for perturbed inputs x' with $\|e(x) - e(x')\|_2 \leq \epsilon$, the output change is bounded by

$$|f_\theta(x) - f_\theta(x')| \leq L_f \cdot (K\epsilon + 2\beta), \tag{22}$$

where L_f is the Lipschitz constant of the final output head. This explains the observed low perturbation degradation in Figure 3. We present this as a conditional account of observed robustness rather than as a proof that unconstrained training is uniformly stable.

Full proofs and additional theoretical discussion are provided in Appendix F (for limitations and assumptions) and Appendix B (for empirical validation of theoretical predictions).

Model Naming Convention. Throughout the appendix, "Coder-XB" refers to Qwen2.5-Coder-XB-Instruct, and "Math-XB" refers to Qwen2.5-Math-XB. Therefore, the model identifiers in tables and text are consistent with the main paper's naming. Moreover, we use "ERNIE-0.3B" for ERNIE-4.5-0.3B and "Llama-3.1-8B" for the Llama-3.1-8B base model.

D.2 Dataset and Teacher Ensemble

D.2.1 Datasets.

We use GSM8K (7,473 train), MATH500 (500 test; training uses the MATH train split, not MATH500), OlympiadBench (6,866 train), LiveCodeBench (5,200 train), HumanEval+ (1,400 train), BBH (3-shot eval), and AGIEval (eval-only). CoT filtering retains chains with 2,048 tokens; concise and verbose CoTs are balanced 1:1.

CoT Construction and Quality Control. For each training instance, we obtain concise and verbose CoT annotations from the teacher ensemble. Specifically, concise CoTs are generated with a length-constrained prompt (max 256 tokens), while verbose CoTs are generated without length constraints (max 2,048 tokens). Then, both variants are filtered for answer consistency with the ground truth. Therefore, only instances with a valid answer under both granularities are retained, ensuring that the alignment objective $\mathcal{L}_{\text{inv}}$ is applied to semantically equivalent reasoning paths. Moreover, to further control quality, we discard any concise or verbose trace that contains hallucinated intermediate steps as identified by a rule-based checker, and we verify that the final answers of the concise and verbose traces match the ground-truth answer in all retained cases. Consequently, the mixture-of-length supervision is constructed entirely from teacher-generated rationales rather than manual annotation, and the quality-control pipeline ensures that both granularities remain semantically aligned.

Cross-Domain Generalization: StrategyQA and LogiQA Dataset Description and Rationale To evaluate whether CRL learns *transferable reasoning representations* beyond the mathematical and code domains seen during training, we conduct zero-shot evaluations on two out-of-distribution benchmarks requiring implicit logical inference:

StrategyQA (Talmor et al., 2021): A dataset of 2,780 yes/no questions requiring implicit, multi-hop reasoning (e.g., world knowledge + logical deduction). We use the official test set (1,490 examples). Unlike GSM8K's explicit arithmetic, StrategyQA requires models to infer unstated premises. **LogiQA** (Liu et al., 2020): A reading comprehension dataset of 8,678 multiple-choice questions (1,536 in the official test split) based on professional logic examinations. Each question provides a context paragraph and 4 options, testing deductive, inductive, and abductive reasoning.

These datasets are **strictly held out**: no training data from StrategyQA or LogiQA is used during CRL's fine-tuning on GSM8K+MATH.

Preprocessing and Formatting Protocol

Tokenization Consistency. All text is tokenized using the **LLaMA 3 tokenizer** (vocab size 128,256) to maintain consistency with the main experiments. We apply the same special token mapping as the training data: <cot> precedes the reasoning trace. <ans> precedes the final answer.

Input Template (Zero-Shot). Since these datasets are evaluated in a zero-shot manner (no in-context examples), we use a standardized prompt template that mirrors the Chain-of-Thought (CoT) structure used during CRL training:

Listing 1: Zero-shot prompt template for StrategyQA.

```
Question: [Question text]
Answer: Let's think step by step. <cot> [Generated CoT] <ans> [Yes
    /No]
```

For LogiQA (multiple-choice), we append the options to the context and constrain the final answer to the corresponding letter (A/B/C/D):

Listing 2: Zero-shot prompt template for LogiQA.

```
Context: [Paragraph text]
Question: [Question text]
Options: A) [Option 1] B) [Option 2] C) [Option 3] D) [Option 4]
Answer: Let's analyze the logic. <cot> [Generated CoT] <ans> [A/B/
    C/D]
```

Answer Extraction. We employ rule-based extraction to ensure fair comparison: **StrategyQA:** Match the first occurrence of "yes" or "no" (case-insensitive) after the <ans> token. If no keyword is found, default to the majority class (baseline failure analysis). **LogiQA:** Match the first capital letter A/B/C/D after <ans>. If the model outputs the full text of the option, we map it back to the letter using exact string matching against the provided options.

Evaluation Protocol

Model Checkpoints. We evaluate the **CRL-7B** checkpoint (best performing on GSM8K dev) and all baselines (SFT-7B, Long CoT-7B, Distilled LoRA-7B) using greedy decoding (`temperature=0`, `top_p=1.0`) to ensure deterministic and reproducible results.

Statistical Significance. Following the main experimental setup (Section 4.1), we report accuracy averaged over **5 random seeds** (42, 123, 456, 789, 1024). Statistical significance is determined via **paired t-test** comparing CRL against the best performing baseline for each dataset.

Table 5: Consolidated CRL hyperparameters across model scales. All settings are applied consistently across datasets unless otherwise specified.

Parameter	**0.3B**	**0.5B**	**7B**	**14B**
Learning Rate	8×10^{-5}	6×10^{-5}	5×10^{-5}	4×10^{-5}
Batch Size (per GPU)	16	16	16	8
Gradient Accumulation	8	8	4	8
LoRA Rank r	8	12	16	32
λ_1 (Invariant Loss)	0.3	0.4	0.5	0.6
λ_2 (Distillation)	0.2	0.25	0.3	0.3
Difficulty Weights (α, β, γ)	(0.4, 0.3, 0.3)	(0.4, 0.3, 0.3)	(0.4, 0.3, 0.3)	(0.4, 0.3, 0.3)
Target Layers $\mathcal{L}$	6–12	8–16	12–24	16–32

D.2.2 Teacher Ensemble.

Two teachers (LLaMA-3-70B and Mixtral 8×22B) each generate $n = 16$ paths ($T = 0.7$, top-p=0.95). Self-consistency filtering retains samples with $\geq$10/16 agreement ($\sim$85% retention). Latent representations are extracted from layers 24–30 (LLaMA), 20–26 (Mixtral), then projected to student space via P trained on 5% held-out data (MSE loss, $\lambda = 0.01$). Therefore, the teacher ensemble size $M = 2$ is consistent with the main-text description.

D.3 Hyperparameters and Tuning

Table 5 provides consolidated hyperparameters. **Note**: We use only λ_1, λ_2 as per the unified objective in Section 3.6, while λ_3 is used only for the auxiliary load-balancing regularizer. However, for the 7B scale, we use $r = 16$, $\lambda_1 = 0.5$, $\lambda_2 = 0.3$, and global batch size 128, matching the main-text configuration. Therefore, the hyperparameters in the main text correspond to the 7B setting, while smaller and larger scales use the values in Table 5. Moreover, the 7B setting is used as the reference configuration, and all scale-specific values are reported in the appendix table.

E Experimental Setup Details

E.1 Training Structured Parameterization

E.2 Cross-Domain Generalization Detailed Results

Figure 6(a) presents the full zero-shot transfer results. CRL significantly outperforms all baselines on both datasets, with particularly large gains on LogiQA (+4.8%), which requires complex logical parsing of long contexts.

Analysis. The consistent gains on StrategyQA (implicit reasoning) and LogiQA (structured logic) suggest that CRL's latent representation alignment ($\mathcal{L}_{\text{inv}}$) captures abstract reasoning primitives (e.g., logical deduction, premise verification) that transfer across domains, rather than overfitting to mathematical syntax.

E.3 Teacher Ensemble Details

The teacher ensemble (LLaMA 3 70B Instruct, Mixtral 8×22B) follows strict protocols:

Algorithm 1 CRL Training with Curriculum and Structured Parameterization

Require: Student f_θ, Teachers $\{T_m\}_{m=1}^{M}$, Dataset $\mathcal{D}$, Hyperparameters $\lambda_1, \lambda_2, \lambda_3$

Initialize $\theta \leftarrow \theta_0$, pre-compute teacher reps and projection P
for step $t = 1$ to T_{total} **do**
  Update threshold $\tau(t) = \tau_{\min} + \frac{1}{2}(\tau_{\max} - \tau_{\min})\left(1 - \cos\left(\frac{t}{T_{\text{total}}}\pi\right)\right)$ (cosine expanding schedule)
  Compute difficulty d_i for all samples (Eq. 6)
  Select batch $\mathcal{B} = \{(x_i, y_i) \in \mathcal{D} : d_i \leq \tau(t)\}$
  for each $(x_i, y_i, \mathcal{R}_i) \in \mathcal{B}$ **do**
    $h(x_i) \leftarrow \text{Pool}\big(\{h_l(x_i)\}_{l\in\mathcal{L}}\big)$, where $h_l(x_i) = \frac{1}{T_{\text{valid}}}\sum_{t=1}^{T_{\text{valid}}} h_{l,t}(x_i)$ and $\text{Pool}(\cdot)$ denotes mean pooling over layers
    if $t \leq T_{\text{warm}}$ **then** use teacher-provided $r^{(s)}, r^{(l)}$ **else** generate $r^{(s)}, r^{(l)}$ from the student model (on-policy decoding); compute routing weights $\mathbf{w}$
    $\mathcal{L}_{\text{CoT}} \leftarrow -\sum_r \pi(r) \log p_\theta(r, y_i|x_i)$
    $\mathcal{L}_{\text{inv}} \leftarrow \|h(x_i, r^{(s)}) - h(x_i, r^{(l)})\|_2^2$
    $\mathcal{L}_{\text{distill}} \leftarrow \|h_{\text{student}} - \frac{1}{M}\sum_m P h_{\text{teacher}}^{(m)}\|_2^2$
    $\mathcal{L}_{\text{total}} \leftarrow \mathcal{L}_{\text{CoT}} + \lambda_1\mathcal{L}_{\text{inv}} + \lambda_2\mathcal{L}_{\text{distill}} + \lambda_3\mathcal{L}_{\text{balance}}$
    $\theta \leftarrow \theta_0 + \sum_{k=1}^{K} w_k(h(x_i))\Delta\theta_k$
    Update adapters via AdamW
  end for
  if $t \mod 100 = 0$ **then** Sparsify adapters (top-10% retention)
end for

Table 6: Teacher generation and self-consistency filtering statistics.

Dataset	Retention Rate	$n = 32$ Gain (%)	Agreement Threshold	Avg. Paths/Instance
GSM8K	87%	0.6	$\geq$10/16	13.9
MATH500	82%	0.8	$\geq$10/16	13.1
OlympiadBench	81%	0.7	$\geq$10/16	13.0
LiveCodeBench	84%	0.5	$\geq$10/16	13.4
HumanEval+	86%	0.4	$\geq$10/16	13.8

- **Generation Parameters**: 16 reasoning paths per instance (temperature=0.7, top-p=0.95, max tokens=2048, repetition penalty=1.1). open-source models run on 8×A100 80GB nodes (Hugging Face `transformers` v4.39.0).
- **Self-Consistency Filtering**: Paths are retained if at least $10/16$ teachers agree on the answer (regex for math, code execution for programming). This retains approximately 85% of instances (GSM8K: 87%, MATH500: 82%, OlympiadBench: 81%, LiveCodeBench: 84%, HumanEval+: 86%).
- **Latent Extraction**: Teacher reps from middle-to-upper layers (LLaMA 3 70B: 24–30, Mixtral: 20–26, cached in FP16 (160GB total).

Teacher Generation Statistics. Table 6 summarizes the self-consistency filtering results. Then, increasing n from 16 to 32 yields less than 1% improvement in retained instances across all datasets, confirming that $n = 16$ is sufficient. Therefore, we use $n = 16$ for all reported experiments. Moreover, the table also reports the per-dataset retention rates and the corresponding $n = 32$ gains, providing direct quantitative support for the sufficiency of $n = 16$.

Teacher and Student Layers. Teacher representations are extracted from layers 24–30 (LLaMA-3-70B) and 20–26 (Mixtral), whereas student representations are extracted from layers 12–24. Therefore, the teacher and student layer sets are distinct, and the projection P maps teacher dimensions to student dimensions. Moreover, the choice of student layers 12–24 is validated in App. B.5, and the teacher layers are chosen to capture high-level reasoning features in the teacher models.

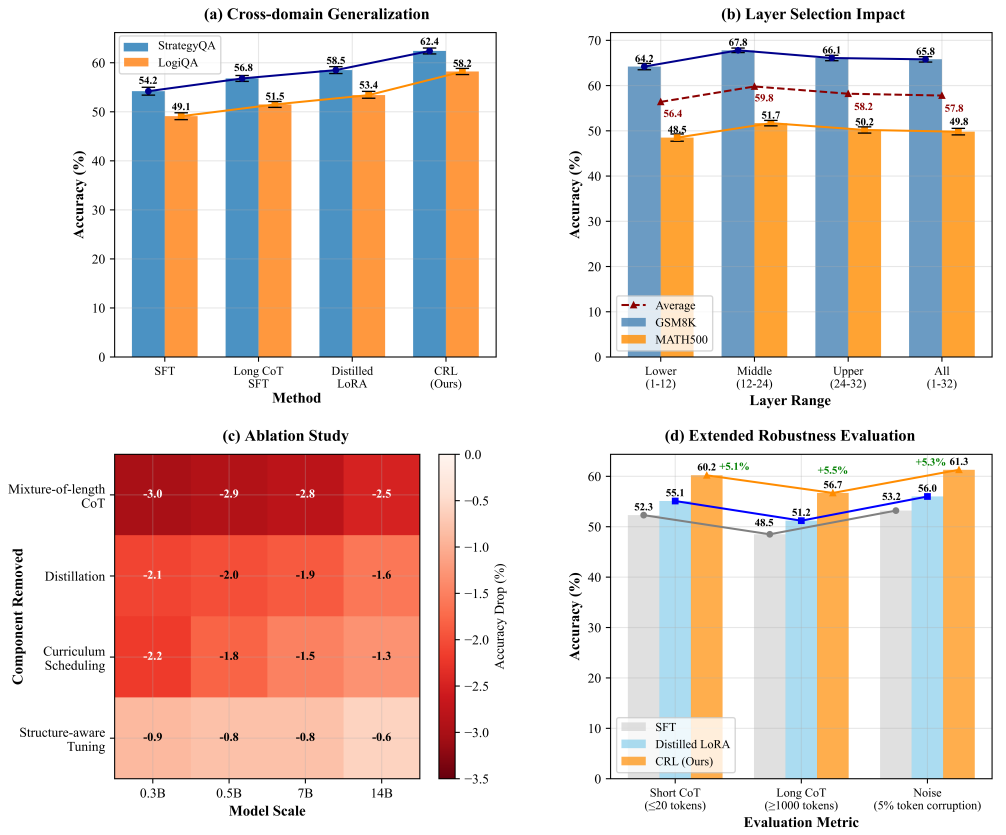

Figure 6: (a) Zero-shot cross-domain generalization (Coder-7B). Values are mean ± std over 5 seeds. **/*** indicates $p < 0.01/0.001$ vs best baseline (paired t-test). Δ shows improvement over best baseline). (b) Impact of layer selection $\mathcal{L}$ on CRL performance (Coder-7B). Middle layers (12–24) achieve optimal accuracy with minimal variance. (c) Ablation results (Δ accuracy %) by model scale. (d) Extended robustness evaluation (Coder-7B).

Table 7: Adapter and LoRA configuration shared by CRL and distillation-based baselines.

Parameter	Value	Description
Rank (r)	16	Low-rank dimension for adapter/LoRA matrices
Scaling (α)	32	Update scaling factor (set to $2r$)
Dropout	0.1	Applied to adapter outputs
Sparse Update	Top-10%	Retain largest-magnitude parameters (every 100 steps)
Target Layers	Reasoning layers	Same layers used for latent representation extraction

E.4 Linear Projection Layer P and Adapter Configuration

The linear projection P (teacher-to-student space) is initialized on 5% held-out data with MSE loss (10 epochs, AdamW lr=1e-4, weight decay=0.01, batch=256, early stopping patience=3). However, instead of freezing P during student training, we fine-tune it jointly with the student model using a smaller learning rate (1e-5). Therefore, P remains adaptive to the evolving student representation space while retaining a stable warm-start point. Consequently, this design aligns with the main text's description of jointly learned projection. Adapters/LoRA use identical parameters for fairness, Details can be seen in Table 7.

Table 8: Dataset statistics and chain-of-thought (CoT) characteristics used in CRL training and evaluation.

Dataset	Task Type	Train	Test	Concise CoT	Verbose CoT	Filtered (%)
GSM8K	Grade-school Math	7,473	1,027	85	210	87
MATH500	High-school Math	–	500	150	380	82
OlympiadBench	Olympiad-level	6,866	1,610	220	550	81
BBH (3-shot)	Multi-task Reasoning	–	1,000	90	230	Eval-only
AGIEval (EN)	Exam Reasoning	–	2,500	110	270	Eval-only
LiveCodeBench	Code Generation	5,200	1,300	180	420	84
HumanEval+	Code Synthesis	1,400	300	120	280	86

The soft router for adapter selection (4 primitives: Arithmetic, Logic, Code, Algebra) uses a 2-layer MLP with entropy regularization (weight=1e-3) to prevent collapse to single adapters. Empirical routing entropy (mean=1.38, std=0.05) confirms stable multi-adapter usage.

Sparse Update Implementation. Every 100 steps, we compute the magnitude of each LoRA parameter and retain the top-10% via a binary mask. Then, the masked parameters are updated while the rest are frozen for that step. Consequently, sparse updates do not alter the low-rank structure of the adapters, and the adapters remain mergeable after training. Moreover, the mask is applied to the LoRA A and B matrices separately, and the resulting sparse updates are compatible with standard LoRA merging because the sparsity pattern is fixed after training and the merged weights can be computed by applying the same mask to the product $\Delta W = BA$.

E.5 Parameter Sensitivity and Dataset Details

Parameter Sensitivity Analysis Using Qwen2.5-Coder-7B on GSM8K/MATH500, we validate hyperparameter robustness (tuned on GSM8K dev set): - **Loss Weights** ($\lambda_1 = 0.5, \lambda_2 = 0.3$): Deviations reduce performance (e.g., $\lambda_1 = 0.3$ drops MATH500 accuracy by 2.9%), confirming balance of invariance and distillation. - **Curriculum Parameters**: Default $\alpha = 0.4, \beta = 0.3, \gamma = 0.3$ (sum=1) and $\tau_{\min} = 0.2, \tau_{\max} = 0.8$ outperforms alternatives (e.g., $\tau_{\max} = 0.7$ drops MATH500 by 2.5- Hyperparameters were selected via grid search ($\lambda_1/\lambda_2 \in \{0.1, 0.3, 0.5, 0.7\}$, LoRA rank $\in \{8, 16, 32\}$) on 10

Dataset Statistics and Preprocessing Table 8 summarizes all datasets. Preprocessing uses LLaMA 3 tokenizer (vocab=128,256, max tokens=4096) with task-specific filters: code validity via `tree-sitter`, math answer normalization, and OlympiadBench multimodal removal. Batches maintain 1:1 concise/verbose CoT ratio (dynamic padding).

More experimental details can be found in the code section of the supplementary materials.

F Broader Impact and Limitations

F.1 Broader Impact

Scientific and Societal Implications. CRL advances the reliability of reasoning in language models, which has direct applications in educational tools, automated code generation, and decision-support systems. By improving data efficiency—achieving full-data performance with 50% fewer samples—our approach reduces the computational cost and environmental footprint associated with training large models. The modular adapter design promotes accessibility, enabling high-quality reasoning capabilities to be deployed on resource-constrained devices via efficient fine-tuning rather than full-scale pre-training.

Risk Considerations. CRL relies on distillation from large proprietary teacher models (e.g., Llama-3), which may inherit and propagate existing biases or erroneous reasoning patterns present in these teachers. While representation alignment mitigates superficial hallucinations, it cannot guarantee the semantic validity of reasoning steps if the teacher ensemble systematically errs. Furthermore, the structured nature of our approach—while beneficial for stability—may limit the model's capacity for novel reasoning patterns not captured by the predefined primitives (arithmetic, logic, code, algebra).

Careful auditing is required before deployment in high-stakes domains such as medical diagnosis or legal reasoning.

F.2 LIMITATIONS

Scalability to Ultra-Large Models: CRL's effectiveness is validated up to 14B parameters. While our theory suggests diminishing returns for larger capacity models (Section 3), we have not empirically verified performance on 70B+ models due to computational constraints.

Domain Specificity: The current reasoning primitives (arithmetic, logic, code, algebra) are tailored to structured problem-solving. Direct application to unstructured domains (e.g., commonsense reasoning, creative writing) may require re-engineering the adapter primitives and latent space constraints.

Teacher Dependency and Cost: Performance is contingent on high-quality teacher generations. However, the requirement for multiple large teachers (LLaMA-70B, Mixtral) introduces significant upfront inference costs, with teacher representations cached in FP16 occupying approximately 160GB in total, although this cost is amortized across student training.

Evaluation Scope: Our robustness tests focus on linguistic perturbations and adversarial distractors. Stress-testing against more sophisticated adversarial attacks (e.g., gradient-based input manipulation) remains future work.