Reasoning as Compression: Unifying Budget Forcing via the Conditional Information Bottleneck

Fabio Valerio Massoli[*]
Qualcomm AI Research[†]

Andrey Kuzmin
Qualcomm AI Research[†]

Arash Behboodi
Qualcomm AI Research[†]

Abstract

Chain-of-Thought (CoT) prompting improves LLM accuracy on complex tasks but often increases token usage and inference cost. Existing "Budget Forcing" methods reduce cost via fine-tuning with heuristic length penalties, suppressing both essential reasoning and redundant filler. We recast efficient reasoning as a lossy compression problem under the Information Bottleneck (IB) principle, and identify a key theoretical gap when applying naive IB to transformers: attention violates the Markov property between prompt, reasoning trace, and response. To resolve this issue, we model CoT generation under the Conditional Information Bottleneck (CIB) principle, where the reasoning trace Z acts as a computational bridge that contains only the information about the response Y that is not directly accessible from the prompt X. This yields a general Reinforcement Learning objective: maximize task reward while compressing completions under a prior over reasoning traces, subsuming common heuristics (e.g., length penalties) as special cases (e.g., uniform priors). In contrast to naive token-counting approaches, we introduce a semantic prior that measures token cost by surprisal under a language model. Crucially, the prior is queried only for token-level log-probabilities, adding negligible overhead to the training loop. Empirically, our CIB objective prunes reasoning redundancy while preserving fluency and logic, improving accuracy at moderate compression and enabling aggressive compression with minimal accuracy drop. These gains generalize across model families and task domains, confirming CIB as a domain-agnostic CoT compression framework.

1 Introduction

Chain-of-Thought (CoT) prompting [45] is the primary mechanism for unlocking reasoning in Large Language Models (LLMs), allowing models to allocate test-time computation for complex tasks. However, this gain incurs significant costs: reasoning chains are often excessively verbose, increasing latency and compute usage. Consequently, "Budget Forcing"—constraining models to yield correct answers within a restricted token budget—has emerged as a critical frontier in efficient inference. Current approaches relying on naive length penalties or strict training-time length constraints are suboptimal. Whether penalizing output length or enforcing a hard token limit, these methods impose a uniform cost on every token, implicitly assuming all tokens contribute equally to the solution. This uniform token penalty—effectively a "flat tax" on computation—ignores the distinction between essential reasoning steps and redundant fillers. Optimizing under such a metric is brittle: models are incentivized to delete tokens regardless of semantic relevance, discarding crucial intermediate logic to satisfy the budget. This makes the accuracy–compute trade-off difficult to tune, as a single weight (or limit) may over-penalize hard prompts while under-penalizing redundancy in easy ones.
In this work, we reframe efficient reasoning not as token minimization, but as *lossy compression*.

[*]Corresponding author: fmassoli@qti.qualcomm.com
[†]Qualcomm AI Research is an initiative of Qualcomm Technologies, Inc.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

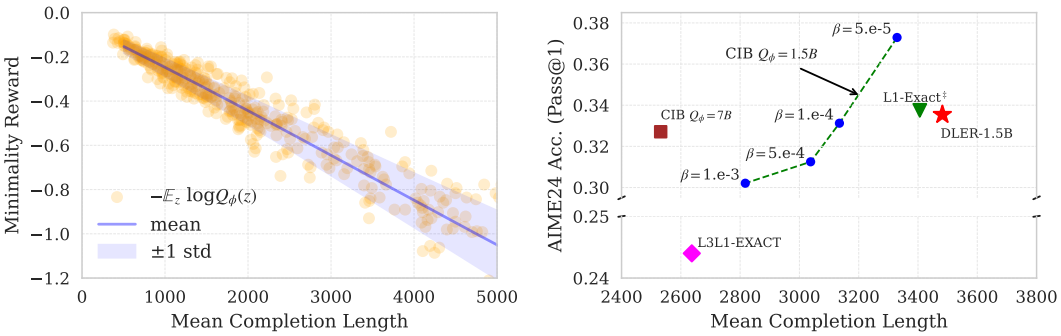

Figure 1: **Left: Minimality Reward vs. Completion Length.** We observe a consistent negative correlation between the completion length and the minimality reward. The shaded blue region shows the $\pm 1\sigma$ band representing the spread of the information cost for the token chosen within CoTs with similar length. **Right: Accuracy vs. Completion Length.** The β parameter confers fine-grained control over the accuracy-compression trade-off. 7B prior (■) yields stronger compression than a 1.5B prior (●). Baselines: DLER [22] (★), L3L1-EXACT [1] (◆), and our L1-Exact‡ model (▼).

We propose a unified framework based on the Information Bottleneck (IB) principle [38], positing that an ideal reasoning chain is the minimal sufficient statistic of the prompt required to predict the answer. We identify that standard IB [38] cannot be naively applied to transformers due to a theoretical inconsistency we term the "*Attention–Markov Incompatibility*": the attention mechanism grants the decoder direct access to the prompt, violating the Markov chain assumption ($Y \leftrightarrow X \leftrightarrow Z$) required by standard IB. We resolve the problem by modeling CoT generation under the Conditional Information Bottleneck (CIB) as *Source Coding with Side Information*. As a result, a novel Reinforcement Learning (RL) objective naturally arises from the CIB framework. Instead of a uniform length penalty, we assign a *semantic cost* to each token based on its information content relative to a frozen base model. This formulation aligns cost with information flow: the model is encouraged to "pay" for informative tokens that increase answer probability while suppressing redundancy. Empirically, this allows for precise navigation of the Pareto frontier, achieving a superior accuracy–compression trade-off compared to length-based baselines (Figure 1, right).
Our contributions are as follows:

- We identify the limitations of length-based budget forcing, showing that uniform penalties and hard limits conflate essential reasoning with redundancy.
- We propose a theoretical framework resolving the "Attention–Markov Incompatibility" via the CIB, yielding a semantic token cost based on relevance rather than length.
- We demonstrate that this formulation compresses reasoning traces while achieving Pareto optimal accuracy-compression trade-off.

The remainder of the paper is organized as follows. Section 2 reviews related work. Section 3 formalizes the "Attention–Markov Incompatibility", derives the CIB objective, and presents our reward model. Section 4 proves that standard length-based penalties arise as a special case of CIB under non-informative priors. Section 5 reports experimental results. Section 6 we report our conclusions and briefly discuss limitations.

2 Related Work

Budget Forcing and Efficient Reasoning. Recent studies suggest that optimal reasoning compute should scale with problem complexity [51], yet unconstrained models often exhibit excessive verbosity even on simple tasks [31]. This has motivated "Budget Forcing" strategies spanning training and inference, including reward shaping with length costs [1], token-level advantages [10], and hard truncation [22]. More granular approaches include difficulty-aware allocation [5] and reference-guided budgeting [46, 20, 23], monotonic uncertainty reduction [44], sometimes tracking history [12] or decomposing costs per-token [15]. Inference-only methods steer generation via auxiliary predictors [19, 8] or employ early-exit decoding [26, 42]. Alternative paradigms replace verbose

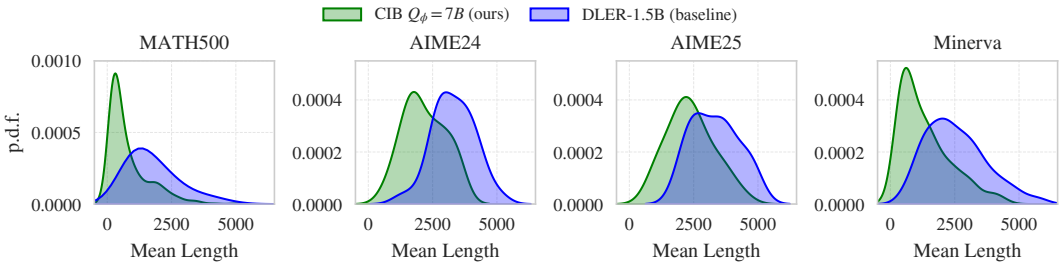

Figure 2: **Lengths Distribution.** Compared to the baseline length distribution (blue curve), the minimality term shifts the length distribution towards shorter completions (green curve). The plotted distributions correspond to models with similar accuracy (see Table 1).

CoT with concise drafting [49, 33], selective reasoning policies [41], or trace compression via token pruning and skipping [48, 6, 7, 4]. [40] further propose budget-aware evaluation metrics. Several more recent works consider budget forcing in the context of multi-turn reasoning tasks [14, 35, 25]. While effective, these methods largely rely on naive token counts as a cost proxy. In contrast, we ground budget forcing in information theory, penalizing tokens based on semantic surprisal rather than raw length.

Information Theory in Large Language Models. The IB principle [38] was proposed as a framework for analyzing deep learning [37], followed by various discussions [34], applications in reasoning and robustness [16], and hallucination detection [39]. However, these works differ from ours in two key respects. First, their objectives typically target generalization or explainability of deep learning rather than strict computational efficiency of reasoning models. Second, they apply the standard IB formulation, which assumes a Markov chain where the latent representation Z mediates all information. Instead, we explicitly take into account the structure of transformer architectures, where the attention mechanism grants the decoder direct access to the prompt, X, creating a collider structure $(X, Z) \to Y$ which breaks the aforementioned Markov property. To the best of our knowledge, this work is the first to unify "Budget Forcing" and Information Theory under a CIB framework.

3 Methodology

We now formalize efficient reasoning as an optimization problem within the CIB framework. We report here the most relevant equations and we refer the reader to Appendix A for the full derivation. In what follows, we refer to X, Z, and Y, as the prompt, CoT, and ground truth answer, respectively.

The Attention–Markov Incompatibility. The standard Information Bottleneck (IB) principle [38] seeks a representation Z that maximally compresses the input X while preserving information about the target Y. Formally, it minimizes the Lagrangian:

$$\mathcal{L}_{\text{IB}} = I(X; Z) - \mu I(Y; Z) \tag{1}$$

over $P(Z|X)$ where μ controls the trade-off between compression (minimizing mutual information $I(X; Z)$) and prediction (maximizing $I(Y; Z)$). Crucially, the standard IB assumes the Markov chain $Y \leftrightarrow X \leftrightarrow Z$, implying that Z is the sole channel through which information flows from X to Y. However, this assumption is fundamentally violated in transformer-based LLMs. Due to the causal attention mechanism, the decoder predicting Y attends to *both* the prompt X and the generated chain Z. This forms a collider structure: $(X, Z) \to Y$. We term this inconsistency the *Attention–Markov Incompatibility*. Under the standard IB objective, maximizing $I(Y; Z)$ can be inefficient as it ignores that the model has access to the query X during the answer generation. This can lead to keeping redundant information about the query X. It is important to note that the conditional probability $P(Y|X)$ of the answer given the query is unknown, and exactly what we want to *simulate* using the intermediate reasoning trace Z.

Conditional Information Bottleneck for Reasoning. To resolve the Attention–Markov Incompatibility, we propose grounding "Budget Forcing" in the Conditional Information Bottleneck (CIB). We

view the prompt X as *side information* that is always available to the answer generator. We require Z to encode only the *additional* information necessary to predict Y given X. The objective becomes:

$$\mathcal{L}_{\text{CIB}} = I(X;Z) - \mu I(Y;Z|X) \tag{2}$$

Minimizing $I(X;Z)$ (or a related upper bound on the rate) while maximizing the conditional predictive power $I(Y;Z|X)$ ensures that the chain Z is penalized for redundancy with X but rewarded for explaining Y. We use an LLM $\pi_\theta(\cdot|\cdot)$ to re-parameterize the optimization problem.

3.1 Problem Formulation

We consider a reasoning task defined by a dataset distribution $P_{\mathcal{D}}(X, Y)$, where X is the prompt and Y the ground truth answer. We aim to learn a stochastic policy $\pi_\theta(Z \mid X)$ that generates a CoT Z to bridge the gap between X and Y, while $\pi_\theta(Y \mid X, Z)$ generates the correct answer.
Our goal is to optimize the policy π_θ to maximize the **Sufficiency** of Z for predicting Y, while minimizing the **Minimality** (information cost) of Z relative to the side information X. This is formalized by the CIB objective: $\min_\theta \mathcal{L}_{\text{CIB}}(\theta) = \min_\theta I(X;Z) - \mu I(Z;Y \mid X)$, where $\mu \geq 0$ controls the rate-distortion trade-off. To derive our final reward function, we rewrite the previous objective as a maximization problem, rather than a minimization one. Therefore, our objective becomes:

$$\max_\theta \mathcal{L}_{\text{CIB}}(\theta) = \max_\theta \underbrace{I(Z;Y \mid X)}_{\text{Sufficiency}} - \beta \underbrace{I(X;Z)}_{\text{Minimality}} \tag{3}$$

where β gives direct control on the trade-off between accuracy and compression level (Figure 1, right). We refer the reader to Appendix A for the detailed discussion on the derivation.

Deriving the Sufficiency Term (Accuracy Reward). We aim to maximize the conditional Mutual Information (MI) $I(Z;Y \mid X)$. We can write it as a function of the policy $\pi_\theta(y|x,z), \pi_\theta(z|x)$ as:

$$\begin{aligned} I(Y;Z|X) &= \sum_{x,y,z} P(x,y)P(z|x,y) \log \frac{\pi_\theta(y|x,z)}{P(y|x)} \\ &= \sum_{x,y,z} P(x,y)\pi_\theta(z|x) \frac{\pi_\theta(y|x,z)}{P(y|x)} \log \frac{\pi_\theta(y|x,z)}{P(y|x)} \geq \sum_{x,y,z} P(x,y)\pi_\theta(z|x) \log \frac{\pi_\theta(y|x,z)}{P(y|x)}, \end{aligned}$$

where we used the inequality $x \log x \geq \log x$ in the last step. Note that the mutual information $I(Z;Y \mid X)$ can be decomposed as $H(Y \mid X) - H(Y \mid X, Z)$. The first term $H(Y \mid X)$ represents the inherent difficulty of the dataset and is constant with respect to θ. Thus, maximizing sufficiency is equivalent to minimizing the conditional entropy $H(Y \mid X, Z)$. We can maximize the lower bound on it and approximate it further using the query-answer samples (x_i, y_i). The first term of the optimization problem can then be approximated as: $\sum_{i=1}^m \mathbb{E}_{Z \sim \pi_\theta(\cdot|x_i)}[\log \pi_\theta(y_i|x_i, Z)]$, where m is the number of samples. In many cases, like RLVR, a verifier $Q_\rho(y_i|x_i, z)$ is used to score the answer. Therefore, we can also optimize the following variational lower bound: $\sum_{i=1}^m \mathbb{E}_{Z \sim \pi_\theta(\cdot|x_i)}[\log Q_\rho(y_i|x_i, Z)]$. See Appendix A for the details of our derivation. In our experiments, we choose $Q_\rho(Y|X, Z)$ such that it gives a reward of 1 (0) for correct (wrong) answers.

From Log-Verifier to Binary Accuracy Reward. Our variational surrogate for sufficiency uses a verifier score $\log Q_\rho(y_i \mid x_i, z_i)$. In our setting the verifier is deterministic, returning $Q_\rho(y \mid x, z) \in \{0, 1\}$ (1 if the extracted answer is correct, else 0), so the log-score is ill-defined for incorrect answers. We therefore use the ε-smoothed verifier $\widetilde{Q}_\rho(y \mid x, z) := \varepsilon + (1-\varepsilon)\, \mathbb{1}(\widehat{y}(x,z) = y)$, where $\varepsilon \in (0,1)$ and $\widehat{y}$ is the predicted answer. Then $\log \widetilde{Q}_\rho(y \mid x, z) = \log \varepsilon - \log \varepsilon\, \mathbb{1}(\widehat{y}(x,z) = y)$. Since $\log \varepsilon$ is a constant w.r.t. (θ, z) and $-\log \varepsilon > 0$, maximizing $\mathbb{E}[\log \widetilde{Q}_\rho(y \mid x, z)]$ is *equivalent* (up to an affine transformation) to maximizing $\mathbb{E}[\mathbb{1}(\widehat{y}(x,z) = y)]$. Accordingly, we define the accuracy reward as: $r_{\text{acc}}(x, y, z) := \mathbb{1}(\widehat{y}(x,z) = y)$, which is a finite, stable surrogate for the log-verifier objective.

Deriving the Minimality Term (Information Cost). We aim to minimize the MI $I(X;Z)$ to penalize redundancy in the CoT: $I(X;Z) = \mathbb{E}_{X,Z}\left[\log \frac{\pi_\theta(Z|X)}{P(Z)}\right]$.

However, computing $P(Z)$ is not tractable. Therefore, we introduce an unconditional variational prior $Q_\phi(Z)$ (a distribution over Z that does not observe X) to find a variational bound similar to [3].

$$\begin{aligned} I(X;Z) = \mathbb{E}_{X,Z}\left[\log \frac{\pi_\theta(Z \mid X)}{P(Z)}\right] &= \mathbb{E}_{X,Z}\left[\log \frac{\pi_\theta(Z \mid X) Q_\phi(Z)}{P(Z) Q_\phi(Z)}\right] \\ &= \mathbb{E}_{X,Z}\left[\log \frac{\pi_\theta(Z \mid X)}{Q_\phi(Z)}\right] - \underbrace{D_{KL}(P(Z) \parallel Q_\phi(Z))}_{\geq 0} \end{aligned} \tag{4}$$

Dropping the non-negative KL term gives the upper bound: $I(X;Z) \leq \mathbb{E}_{X,Z}\left[-\log Q_\phi(Z)\right] - H(Z \mid X)$, where $Z \sim \pi_\theta(\cdot|X)$. We instantiate $Q_\phi(Z)$ using a frozen, pre-trained base model (not an instruction-finetuned model), ensuring it captures the statistics of general language without task-specific conditioning. More details on the derivation can be found in Appendix A.
The first term, $\mathbb{E}_{X,Z}[-\log Q_\phi(Z)]$, represents the cross-entropy rate (or description length) of the chain under the prior. It corresponds to the expected value of the reasoning trace information cost: $C(Z) := \sum_{t=1}^{|Z|} -\log Q_\phi(z_t \mid z_{<t})$. The second term, $-H(Z \mid X)$, corresponds to the negative entropy of the policy. In RL algorithms like PPO, this term is naturally handled via an entropy regularization bonus to encourage exploration.

Reward Modeling. Combining the bounds, we aim to maximize the following objective:

$$\mathcal{L}_{\text{CIB}} = \mathbb{E}_{(X,Y)\sim P_{\mathcal{D}}, Z\sim\pi_\theta}\Big[\log \widetilde{Q}_\rho(Y|X,Z) + \beta \sum_{t=1}^{T} \log Q_\phi(z_t \mid z_{<t})\Big], \tag{5}$$

where the first term represents the accuracy score from the verifier, $\widetilde{Q}_\rho$, and Q_ϕ is chosen as prior distribution. We define our reward model as: $R(X,Y,Z) := r_{\text{acc}}(X,Y,Z) + \beta r_{\text{min}}(X,Z)$, where $r_{\text{acc}}(X,Y,Z) := \mathbb{1}\Big(\widehat{Y}(X,Z) = Y\Big)$ is the accuracy reward, taking a value of 1 if the predicted answer matches the ground truth Y, and 0 otherwise, and $r_{\text{min}}(X,Z) := \sum_{t=1}^{T} \log Q_\phi(z_t \mid z_{<t})$,is the cumulative surprisal (information cost) of the reasoning chain relative to the prior. Accuracy remains the primary objective, while r_{min} acts as a semantic regularizer controlled by the coefficient β. This effectively assigns a semantic cost to every token: the cost $-\log Q_\phi$ penalizes low-probability (high-surprisal) tokens unless they contribute significantly to solving the task (r_{acc}). Tokens that are redundant or verbose increase the cumulative cost without improving accuracy, and are thus suppressed by the policy.

4 Theoretical Analysis: A Unified Framework

A central motivation for this work is demonstrating that the CIB serves as a general framework from which, e.g., length-based penalties naturally arise as a special case. As an example, we prove that length-constrained methods correspond to the CIB rate term with non-informative priors.

Proposition 4.1. A standard length-based penalty (e.g., $g(Z) = \alpha f(|Z|)$) is equivalent to the CIB objective under the assumption of a maximum entropy (uniform) prior, Q, over the vocabulary.

Proof. Let $|V|$ be the vocabulary size and consider the minimality term $\sum -\log Q(z_t)$. A Maximum Entropy prior implies a uniform distribution over the vocabulary V (i.e., $Q(z_t) = \frac{1}{|V|}$ for all z_t). Thus, the surprisal of every token becomes constant: $c = \log|V|$. Then, the total information cost for a CoT, Z, of length T becomes: $-\log Q(Z) = -\sum_{t=1}^{T} \log(\frac{1}{|V|}) = T \cdot \log|V|$.
Substituting this into the CIB objective, the penalty term becomes $\beta T \log|V|$. By setting $\alpha = \beta \log|V|$, we recover a linear length penalty. This proves that linear penalties implicitly assume that all tokens carry equal information content ($\log|V|$), ignoring the underlying semantics of the CoT. We provide further experimental analysis of uniform prior and its relation to the semantic prior in Appendix D. □

Proposition 4.2. Target-length penalties, such as LCPO-Exact [1], correspond to the CIB objective with a Laplace prior.

Proof. Any penalty function $g(Z)$ applied to the reward can be interpreted as an implicit prior $Q(Z) \propto \exp(-g(Z))$. LCPO-Exact penalizes deviation from a target length n_{gold} via the term

$g(Z) = |n_{gold} - n_y|$, where n_y is the length of the generated CoT. The corresponding implicit prior is: $Q_{\text{LCPO}}(Z) \propto e^{-|n_{gold}-n_y|}$. This is a Laplace-like distribution over the sequence length T, centered at n_{gold}. Interpreting LCPO through this lens reveals a strong inductive bias: it posits that there exists a golden length for reasoning length, and any deviation (shorter or longer) is exponentially improbable. □

Crucially, in both Propositions 4.1 and 4.2, the implicit prior $Q(Z)$ depends solely on the sequence length, whereas our proposed CIB method uses a language model prior defining a per-token cost.

5 Experimental Results

Training and Evaluation Details We evaluate CIB on three families of small reasoning models: DLER-{1.5B, 7B} [22], OpenReasoning-Nemotron-1.5B [2, 30], and DeepScaleR-1.5B [24]. We focus on the 1.5B–7B range, as it is the primary deployment target for efficient reasoning under latency and memory constraints. Training uses GRPO [36] with a group size of 16 and a Qwen2.5-Base-{1.5B, 7B} prior; the prior is used at training time only and adds no inference cost. Our main evaluation covers five mathematical reasoning benchmarks—MATH500 [21], AIME24 [28], AIME25 [29], Minerva [17], and OlympiadBench [9]—which provide unambiguous, deterministic targets and therefore a controlled evaluation setting. To assess whether CIB generalizes beyond math, we also evaluate on GPQA [32], MMLU-Pro [43], and LiveCodeBench [13]. We follow the same protocol as in Liu et al. [22], using vLLM as the inference engine (temperature 0.6, $\text{top}_p = 0.95$, max tokens 32K, 16 generations per prompt), and report pass@1 accuracy. The standard error of the mean (SEM) is computed over 10 seeds for every benchmark except MATH500 and LiveCodeBench. Further training and evaluation details are in Appendix E, and the list of assets used in the paper in Appendix G.

5.1 Main Results

We first verify that the minimality reward (Section 3) provides a usable learning signal. Figure 1 (left) shows a negative correlation with completion length, with dispersion suggesting sensitivity to the token sequence beyond length alone.

Table 1 reports accuracy and average CoT length on the mathematical benchmarks. For each model family we evaluate CIB at two compression strengths ($\beta^- = 5.0 \times 10^{-5}$, $\beta^+ = 1.5 \times 10^{-4}$), selected on the training distribution to target mild and more aggressive compression, and with two prior sizes (1.5B and 7B). For DeepScaleR-1.5B, we include external baselines where available: L3L1-EXACT [1], evaluated from the authors' public checkpoint, and ThinkPrune-I2k [11], HAPO [12], and LAPO-I [46] (marked §), with results quoted directly from Wu et al. [46]; some benchmark entries are therefore unavailable. Further comparisons are reported in Appendix F.

CIB Improves Accuracy while Compressing Reasoning. Across all four model groups, CIB$_{\beta^-}$ improves average accuracy over the uncompressed baseline while simultaneously shortening the CoT (Table 1). DeepScaleR-1.5B offers the broadest comparison against length-based penalties: CIB$_{\beta^-}$ gains 1.5 points of average accuracy and cuts average CoT length by $\sim 24\%$, with improvements consistent across both easy and hard benchmarks. Length-based penalties, in contrast, often trade accuracy for shorter outputs: L3L1-EXACT loses 13.8 points on AIME24 despite strong compression. CIB$_{7B}$ is both more accurate and more compressive than ThinkPrune-I2k on every shared benchmark, while CIB$_{\beta^-,1.5B}$ is more accurate still, at the cost of slightly longer traces. LAPO-I sits between the two on the accuracy–compression plane: CIB$_{\beta^-,1.5B}$ is more accurate on MATH500 and AIME24, CIB$_{7B}$ compresses more aggressively, and LAPO-I's apparent accuracy advantage over CIB$_{7B}$ is within uncertainty.

The two levers behave predictably. Raising β from β^- to β^+ trades up to ~ 1.6 points of accuracy for 5%–21% fewer tokens (Table 1, Avg. column). The 7B prior enables deeper compression, yielding the shortest CoT in every group; the only shorter model is L3L1-EXACT on DeepScaleR-1.5B, whose compression comes with substantial accuracy degradation (Table 1, red entries). These trends hold across families. On Nemotron-1.5B, CIB$_{\beta^-,1.5B}$ achieves the largest accuracy gain ($+2.3$ points) with $\sim 7\%$ fewer tokens, while CIB$_{7B}$ removes $\sim 20\%$ and still improves on the baseline. CIB also scales to larger models: on DLER-7B, CIB$_{\beta^-,7B}$ matches the baseline's strong accuracy with a

Table 1: **Math benchmarks: accuracy (↑) and average CoT length (↓).** Bold indicates best performance within each model group. **Red** marks significant degradation, green highlights improvement over the baseline. CIB subscripts denote the compression setting ($\beta^- = 5 \times 10^{-5}$, $\beta^+ = 1.5 \times 10^{-4}$) and prior size (1.5B or 7B). No SEM available for baselines reported from literature.
*: Pass@1 with $n = 16$ for consistency, the official HuggingFace page reports pass@1(Avg@64);
†: Our models; ‡: Models we trained using the L1-EXACT length penalty [1]; §: Results from [46].

Model	MATH500 Acc. / Tok.	AIME24 Acc. / Tok.	AIME25 Acc. / Tok.	Minerva Acc. / Tok.	Olympiad Acc. / Tok.	Avg. Acc. / Tok.
DeepScaleR-1.5B						
Baseline	$85.8_{\pm 1.6}$ / $3190_{\pm 132}$	$38.2_{\pm 1.5}$ / $9331_{\pm 116}$	$25.4_{\pm 1.5}$ / $8907_{\pm 144}$	$19.9_{\pm 0.8}$ / $6386_{\pm 95}$	$54.8_{\pm 0.4}$ / $5922_{\pm 29}$	44.8 / 6747
L3L1-EXACT	$85.2_{\pm 1.6}$ / $1932_{\pm 60}$	**$24.4_{\pm 1.3}$ / $2637_{\pm 15}$**	**$20.2_{\pm 1.3}$ / $2608_{\pm 17}$**	$19.8_{\pm 0.8}$ / $2124_{\pm 29}$	**$48.8_{\pm 0.3}$ / $2369_{\pm 6}$**	**39.7 / 2334**
ThinkPrune-I2k§	$85.5_{\pm}$ / $1707_{\pm}$	$34.9_{\pm}$ / $5095_{\pm}$	– / –	– / –	$54.7_{\pm}$ / $3498_{\pm}$	– / –
HAPO§	$84.4_{\pm}$ / $2370_{\pm}$	$31.4_{\pm}$ / $7702_{\pm}$	– / –	– / –	$51.4_{\pm}$ / $4571_{\pm}$	– / –
LAPO-I§	$86.3_{\pm}$ / $2168_{\pm}$	$38.1_{\pm}$ / $5371_{\pm}$	– / –	– / –	**$56.3_{\pm}$** / $4024_{\pm}$	– / –
CIB$^{\dagger}_{\beta^-,1.5B}$	**$87.6_{\pm 1.5}$** / $2359_{\pm 121}$	**$40.2_{\pm 1.9}$** / $7298_{\pm 141}$	**$27.8_{\pm 2.1}$** / $6866_{\pm 170}$	**$20.4_{\pm 0.8}$** / $4510_{\pm 77}$	$55.5_{\pm 0.7}$ / $4472_{\pm 47}$	**46.3** / 5101
CIB$^{\dagger}_{\beta^+,1.5B}$	$87.2_{\pm 1.5}$ / $2017_{\pm 107}$	$38.2_{\pm 2.0}$ / $6937_{\pm 141}$	$27.6_{\pm 2.1}$ / $6757_{\pm 184}$	$20.1_{\pm 0.8}$ / $4147_{\pm 73}$	$55.3_{\pm 0.7}$ / $4211_{\pm 45}$	45.7 / 4814
CIB$^{\dagger}_{7B}$	$86.1_{\pm 1.6}$ / **$1337_{\pm 93}$**	$37.5_{\pm 1.9}$ / $5003_{\pm 143}$	$25.0_{\pm 1.9}$ / $5389_{\pm 164}$	$19.7_{\pm 0.8}$ / **$2058_{\pm 68}$**	$55.0_{\pm 0.7}$ / $3065_{\pm 41}$	44.7 / **3370**
Nemotron-1.5B						
Baseline*	$85.8_{\pm 1.6}$ / $8509_{\pm 382}$	$43.1_{\pm 1.2}$ / $22931_{\pm 326}$	**$35.1_{\pm 1.3}$** / $24086_{\pm 333}$	$11.0_{\pm 0.5}$ / $13011_{\pm 191}$	$43.8_{\pm 0.3}$ / $16270_{\pm 89}$	43.8 / 16961
L1-Exact‡	$85.8_{\pm 1.5}$ / $8807_{\pm 393}$	$40.2_{\pm 1.4}$ / $22079_{\pm 383}$	$30.8_{\pm 1.4}$ / $23243_{\pm 400}$	$11.4_{\pm 0.4}$ / $12925_{\pm 147}$	$44.0_{\pm 0.4}$ / $15316_{\pm 127}$	42.4 / 16474
CIB$^{\dagger}_{\beta^-,1.5B}$	**$86.8_{\pm 1.6}$** / $7536_{\pm 412}$	**$45.3_{\pm 1.9}$** / $21962_{\pm 483}$	$34.1_{\pm 1.9}$ / $22980_{\pm 547}$	**$11.7_{\pm 0.6}$** / $11118_{\pm 189}$	**$52.7_{\pm 0.6}$** / $15089_{\pm 176}$	**46.1** / 15737
CIB$^{\dagger}_{\beta^+,1.5B}$	$85.4_{\pm 1.6}$ / $5982_{\pm 354}$	$41.5_{\pm 1.9}$ / $20954_{\pm 454}$	$34.0_{\pm 1.8}$ / $21407_{\pm 542}$	$10.9_{\pm 0.6}$ / $10051_{\pm 168}$	$50.8_{\pm 0.6}$ / $13894_{\pm 173}$	44.5 / 14458
CIB$^{\dagger}_{7B}$	$86.7_{\pm 1.5}$ / **$5789_{\pm 366}$**	$42.7_{\pm 1.8}$ / **$20547_{\pm 470}$**	$32.3_{\pm 1.8}$ / **$20678_{\pm 532}$**	$10.6_{\pm 0.5}$ / **$8752_{\pm 177}$**	$51.1_{\pm 0.6}$ / **$12242_{\pm 170}$**	44.7 / **13602**
DLER-1.5B						
Baseline	$86.4_{\pm 1.5}$ / $1873_{\pm 91}$	$33.4_{\pm 1.4}$ / $3478_{\pm 25}$	$23.9_{\pm 1.5}$ / $3290_{\pm 29}$	$20.9_{\pm 0.8}$ / $2587_{\pm 29}$	$54.1_{\pm 0.4}$ / $2766_{\pm 10}$	43.7 / 2799
L1-Exact‡	$86.6_{\pm 1.5}$ / $1751_{\pm 111}$	$35.0_{\pm 2.1}$ / $3274_{\pm 34}$	$22.7_{\pm 2.0}$ / $3037_{\pm 41}$	$21.2_{\pm 0.8}$ / $2277_{\pm 26}$	$54.0_{\pm 0.7}$ / $2577_{\pm 20}$	43.9 / 2583
CIB$^{\dagger}_{\beta^-,1.5B}$	**$88.4_{\pm 1.5}$** / $1687_{\pm 102}$	**$35.4_{\pm 2.0}$** / $3198_{\pm 40}$	**$25.3_{\pm 2.1}$** / $3260_{\pm 56}$	**$21.6_{\pm 0.8}$** / $2346_{\pm 27}$	**$54.3_{\pm 0.7}$** / $2616_{\pm 18}$	**45.0** / 2621
CIB$^{\dagger}_{\beta^+,1.5B}$	$87.6_{\pm 1.4}$ / $1421_{\pm 64}$	$34.1_{\pm 2.0}$ / $3124_{\pm 43}$	$24.2_{\pm 2.1}$ / $3027_{\pm 47}$	$21.0_{\pm 0.8}$ / $2305_{\pm 27}$	$54.1_{\pm 0.7}$ / $2544_{\pm 17}$	44.2 / 2484
CIB$^{\dagger}_{7B}$	$85.0_{\pm 1.6}$ / **$805_{\pm 35}$**	$31.6_{\pm 2.0}$ / **$2504_{\pm 37}$**	$22.4_{\pm 1.9}$ / **$2257_{\pm 38}$**	$20.4_{\pm 0.9}$ / **$1327_{\pm 23}$**	$51.6_{\pm 0.7}$ / **$1631_{\pm 16}$**	42.2 / **1705**
DLER-7B						
Baseline	$93.6_{\pm 1.1}$ / $1159_{\pm 41}$	$49.6_{\pm 1.7}$ / $3221_{\pm 47}$	$36.6_{\pm 1.6}$ / $3271_{\pm 54}$	$27.3_{\pm 1.0}$ / $1849_{\pm 30}$	**$65.0_{\pm 0.4}$** / $2840_{\pm 27}$	54.4 / 2468
L1-Exact‡	$92.8_{\pm 1.2}$ / $675_{\pm 33}$	$46.2_{\pm 2.5}$ / $2982_{\pm 62}$	$31.2_{\pm 2.2}$ / $2564_{\pm 61}$	$26.1_{\pm 1.0}$ / $1144_{\pm 30}$	$61.9_{\pm 0.7}$ / $1872_{\pm 26}$	51.6 / 1847
CIB$^{\dagger}_{\beta^-,7B}$	**$94.0_{\pm 1.1}$** / $1033_{\pm 39}$	**$49.7_{\pm 2.3}$** / $2642_{\pm 69}$	**$37.1_{\pm 2.2}$** / $2962_{\pm 77}$	**$27.6_{\pm 1.0}$** / $1381_{\pm 26}$	$64.1_{\pm 0.7}$ / $2361_{\pm 39}$	**54.5** / 2076
CIB$^{\dagger}_{\beta^+,7B}$	$92.2_{\pm 1.2}$ / **$678_{\pm 34}$**	$48.2_{\pm 2.2}$ / **$2419_{\pm 75}$**	$35.4_{\pm 2.3}$ / **$2550_{\pm 72}$**	$25.8_{\pm 1.0}$ / **$851_{\pm 19}$**	$62.7_{\pm 0.7}$ / **$1704_{\pm 28}$**	52.9 / **1640**

$\sim$16% shorter CoT, while CIB$_{\beta^+,7B}$ reduces length by roughly one-third at a cost of 1.5 accuracy points.

Generalization Beyond Math. Although trained exclusively on math data, CIB benefits transfer to science, knowledge, and code benchmarks (Table 2). On DeepScaleR-1.5B, CIB$_{\beta^-}$ improves GPQA Diamond by 1.9 points ($36.5_{\pm 0.7} \rightarrow 38.4_{\pm 0.6}$) and MMLU-Pro by 2.2 points ($39.4_{\pm 0.2} \rightarrow 41.6_{\pm 0.2}$) while matching LiveCodeBench ($18.7_{\pm 2.6} \rightarrow 18.8_{\pm 1.2}$), with token reductions of up to 38.6%. On DLER-7B, it performs on par with the baseline across all three benchmarks while using 10.9%–17.4% fewer tokens. LiveCodeBench stays within 1 point of the baseline for β^- and β^+ in every family; only the most aggressive 7B-prior configurations exceed that margin (DeepScaleR -1.3; DLER-1.5B -1.1). Two family-specific patterns are stand out. On DLER-1.5B, the baseline holds the best GPQA Diamond accuracy in its group ($40.9_{\pm 0.7}$) and every compression method lowers it, but CIB$_{\beta^-}$ loses only 1.5 points ($39.4_{\pm 0.6}$) against L1-Exact's 4.1 ($36.8_{\pm 0.6}$), indicating that CIB preserves more domain-specific knowledge even where some regression is unavoidable. On Nemotron, the 1.5B-prior variants cost up to 2.0 points on MMLU-Pro ($41.4 \rightarrow 39.4$), which the 7B prior fully recovers (41.6, $+0.2$ over the baseline) while still compressing by 19.0% ($\sim$10.5K $\rightarrow$ $\sim$8.5K), suggesting that a stronger prior compensates for the capacity gap on knowledge-intensive tasks.

Statistical Reliability. On competition-level benchmarks (AIME24, AIME25), individual CIB$_{\beta^-}$ gains are comparable in magnitude to the SEM (DeepScaleR AIME24: $38.2_{\pm 1.5} \rightarrow 40.2_{\pm 1.9}$; DLER-1.5B AIME24: $33.4_{\pm 1.4} \rightarrow 35.4_{\pm 2.0}$), as expected for 30-problem test sets. The key signal is *consistency*: CIB$_{\beta^-}$ raises accuracy on 25 of 32 model$\times$benchmark evaluations across Tables 1 and 2. The length-penalty degradations, by contrast, are individually unambiguous—L3L1-EXACT loses 13.8 points on AIME24 ($38.2_{\pm 1.5} \rightarrow 24.4_{\pm 1.3}$) and DLER-7B L1-Exact 3.1 points on OlympiadBench ($65.0_{\pm 0.4} \rightarrow 61.9_{\pm 0.7}$)—well beyond overlapping confidence intervals.

Controlling the Accuracy–Efficiency Trade–Off. The weight β in the CIB objective (Equation 3) gives fine-grained control over the accuracy–efficiency trade–off. Sweeping β traces a smooth Pareto frontier (Figure 1, right): low values prioritize accuracy, high values compress aggressively. On

Table 2: **Science, Knowledge & Coding benchmarks: Accuracy (↑) and average CoT length (↓).** Bold indicates best performance within each model group. **Red** marks significant degradation, **green** highlights improvement over the baseline. CIB subscripts denote the compression setting ($\beta^- = 5 \times 10^{-5}$, $\beta^+ = 1.5 \times 10^{-4}$) and prior size (1.5B or 7B). No SEM available for baselines reported from literature.
*: Pass@1 with $n = 16$ for consistency, the official HuggingFace page reports pass@1(Avg@64);
†: Our models; ‡: Models we trained using the L1-EXACT length penalty [1].

Model	GPQA Diamond Acc. / Tok.	MMLU-Pro Acc. / Tok.	LCB Acc. / Tok.	Avg. Acc. / Tok.
DeepScaleR-1.5B				
Baseline	$36.5_{\pm 0.7}$ / $7445_{\pm 47}$	$39.4_{\pm 0.2}$ / $5285_{\pm 16}$	$18.7_{\pm 2.6}$ / $11403_{\pm 301}$	31.5 / 8044
L3L1-EXACT	$37.9_{\pm 0.7}$ / $3125_{\pm 16}$	$40.4_{\pm 0.2}$ / $2240_{\pm 5}$	$17.5_{\pm 2.5}$ / $\mathbf{4612}_{\pm 89}$	31.9 / **3326**
CIB$^{\dagger}_{\beta^-,\,1.5B}$	$\mathbf{38.4}_{\pm 0.6}$ / $4571_{\pm 98}$	$\mathbf{41.6}_{\pm 0.2}$ / $3286_{\pm 14}$	$\mathbf{18.8}_{\pm 1.2}$ / $10410_{\pm 151}$	**32.9** / 6089
CIB$^{\dagger}_{\beta^+,\,1.5B}$	$35.5_{\pm 0.6}$ / $4429_{\pm 87}$	$40.7_{\pm 0.2}$ / $3108_{\pm 14}$	$18.6_{\pm 1.1}$ / $10274_{\pm 134}$	31.6 / 5937
CIB$^{\dagger}_{7B}$	$39.5_{\pm 0.6}$ / $\mathbf{2850}_{\pm 87}$	$40.8_{\pm 0.2}$ / $\mathbf{2118}_{\pm 13}$	$17.4_{\pm 1.1}$ / $9481_{\pm 126}$	32.6 / 4816
Nemotron-1.5B				
Baseline*	$31.4_{\pm 0.7}$ / $21238_{\pm 154}$	$41.4_{\pm 0.2}$ / $10490_{\pm 39}$	$26.6_{\pm 2.5}$ / $24082_{\pm 620}$	33.1 / 18603
L1-Exact‡	$33.9_{\pm 1.1}$ / $21132_{\pm 219}$	$39.6_{\pm 0.1}$ / $10416_{\pm 30}$	$\mathbf{26.8}_{\pm 1.3}$ / $23995_{\pm 310}$	33.4 / 18514
CIB$^{\dagger}_{\beta^-,\,1.5B}$	$\mathbf{34.9}_{\pm 0.6}$ / $22558_{\pm 299}$	$39.4_{\pm 0.2}$ / $9851_{\pm 41}$	$26.1_{\pm 1.2}$ / $23557_{\pm 295}$	33.5 / 18655
CIB$^{\dagger}_{\beta^+,\,1.5B}$	$34.8_{\pm 0.6}$ / $19694_{\pm 307}$	$39.6_{\pm 0.2}$ / $8908_{\pm 38}$	$26.1_{\pm 1.2}$ / $22227_{\pm 308}$	33.5 / 16943
CIB$^{\dagger}_{7B}$	$33.6_{\pm 0.7}$ / $\mathbf{19533}_{\pm 319}$	$\mathbf{41.6}_{\pm 0.2}$ / $\mathbf{8496}_{\pm 39}$	$25.9_{\pm 1.2}$ / $\mathbf{21817}_{\pm 304}$	**33.7** / **16615**
DLER-1.5B				
Baseline	$\mathbf{40.9}_{\pm 0.7}$ / $3280_{\pm 16}$	$41.7_{\pm 0.2}$ / $2575_{\pm 12}$	$17.9_{\pm 2.5}$ / $4857_{\pm 89}$	**33.5** / 3571
L1-Exact‡	$36.8_{\pm 0.6}$ / $3012_{\pm 30}$	$40.8_{\pm 0.2}$ / $2326_{\pm 11}$	$17.8_{\pm 1.1}$ / $4585_{\pm 39}$	31.8 / 3308
CIB$^{\dagger}_{\beta^-,\,1.5B}$	$39.4_{\pm 0.6}$ / $3183_{\pm 47}$	$\mathbf{41.8}_{\pm 0.2}$ / $2469_{\pm 13}$	$\mathbf{18.2}_{\pm 1.3}$ / $4727_{\pm 46}$	33.1 / 3460
CIB$^{\dagger}_{\beta^+,\,1.5B}$	$38.8_{\pm 0.5}$ / $3044_{\pm 36}$	$41.6_{\pm 0.2}$ / $2353_{\pm 12}$	$17.9_{\pm 1.1}$ / $4629_{\pm 42}$	32.8 / 3342
CIB$^{\dagger}_{7B}$	$36.7_{\pm 0.6}$ / $\mathbf{2423}_{\pm 43}$	$40.4_{\pm 0.2}$ / $\mathbf{1973}_{\pm 15}$	$16.8_{\pm 1.1}$ / $\mathbf{3977}_{\pm 40}$	31.3 / **2791**
DLER-7B				
Baseline	$50.1_{\pm 0.8}$ / $3255_{\pm 19}$	$\mathbf{58.7}_{\pm 0.2}$ / $2012_{\pm 5}$	$\mathbf{35.9}_{\pm 3.2}$ / $4847_{\pm 130}$	48.2 / 3371
L1-Exact‡	$48.6_{\pm 0.7}$ / $2551_{\pm 40}$	$56.9_{\pm 0.2}$ / $1446_{\pm 6}$	$34.4_{\pm 1.4}$ / $4120_{\pm 52}$	46.6 / 2706
CIB$^{\dagger}_{\beta^-,7B}$	$\mathbf{50.5}_{\pm 0.8}$ / $2899_{\pm 33}$	$\mathbf{58.7}_{\pm 0.2}$ / $1718_{\pm 5}$	$35.6_{\pm 1.4}$ / $4003_{\pm 55}$	**48.3** / 2873
CIB$^{\dagger}_{\beta^+,7B}$	$49.7_{\pm 0.7}$ / $\mathbf{2476}_{\pm 32}$	$58.2_{\pm 0.2}$ / $\mathbf{1430}_{\pm 6}$	$34.5_{\pm 1.4}$ / $\mathbf{3796}_{\pm 61}$	47.5 / **2567**

AIME24, intermediate settings retain most of the baseline accuracy at a substantially lower token count, while extreme values reveal a clear knee beyond which compression degrades reasoning quality. This continuous knob lets practitioners pick the operating point matching their latency or cost constraints.

Qualitative CoT Comparison. Inspecting reasoning traces on arithmetic and symbolic tasks shows that CIB systematically removes conversational scaffolding, redundant verification loops, and tautological checks. Unlike naive truncation, the semantic prior alters the reasoning topology, preserving the "computational bridge" while filtering transitions with low marginal information about Y. Case studies are in Appendix C (Figures 5–7).

Prior Overhead. A natural concern is whether the prior becomes a training bottleneck. Since Q_ϕ is frozen, it needs no gradient accumulation, backward passes, or optimizer states: it adds only a static VRAM footprint and a single forward pass over the generated trajectory, scoring all tokens in parallel. On our distributed setup (4×H100, 8 generations per sample, vLLM backend), Table 3 compares wall-clock time across four prior sizes against a length-based penalty reward for DLER-1.5B. Scaling the prior to 7B reduces rollout token throughput by less than 6% and the effective data collection rate by less than 6.2% ($1.47 \rightarrow 1.38$ samples/sec), so training is largely insensitive to prior size in wall-clock terms. Prior *capacity*, however, does matter for the quality of the semantic signal. A prior that is too weak has a predictive distribution flattening toward uniform, and the semantic cost degenerates into a naive length penalty that cannot separate redundant tokens from critical reasoning steps.

A cross-family prior introduces tokenization-dependent confounds instead: vocabulary and segmentation mismatches can penalize the policy's native formatting rather than its actual verbosity. Coarser-grained costs that mitigate this effect are an interesting direction for future works.

Table 3: **Semantic prior training overhead.** Wall-clock comparison for DLER-1.5B. †: No prior model.

Prior	Step Time (s)	Tokens/sec	Samples/sec
Length-based†	$21.1_{\pm 0.3}$	$3691_{\pm 92}$	$1.47_{\pm 0.02}$
$Q_\phi = 0.5$B	$21.8_{\pm 0.3}$	$3677_{\pm 97}$	$1.46_{\pm 0.03}$
$Q_\phi = 1.5$B	$22.5_{\pm 0.3}$	$3482_{\pm 96}$	$1.42_{\pm 0.03}$
$Q_\phi = 3$B	$22.4_{\pm 0.4}$	$3489_{\pm 99}$	$1.43_{\pm 0.03}$
$Q_\phi = 7$B	$23.2_{\pm 0.4}$	$3449_{\pm 98}$	$1.38_{\pm 0.03}$

5.2 Scaling Properties

Figure 3 shows the inference-time compute (ITC) behaviour of our CIB models against the baselines on AIME24 and AIME25, measured as pass@k for k up to 128. At the standard 8K budget (first and last columns), CIB-compressed models scale on par with or better than their base models, despite producing substantially shorter reasoning traces. On DeepScaleR-1.5B, both priors improve over the base model at every k and the advantage of CIB grows as the budget tightens. When the maximum generation length is bounded to 3K (second column) or 2K (third column), CIB with the large prior ($Q_\phi = 7B$) achieves the best scaling performance in almost every setting. At 2K tokens it nearly doubles the pass@1 of DLER on AIME24 (0.21 vs. 0.11) and keeps a clear margin up to pass@128. Notably, under the 2K budget the small prior ($Q_\phi = 1.5B$) falls slightly below the base model at large k, which indicates that a stronger prior is key to retaining useful reasoning when the budget is severely constrained.

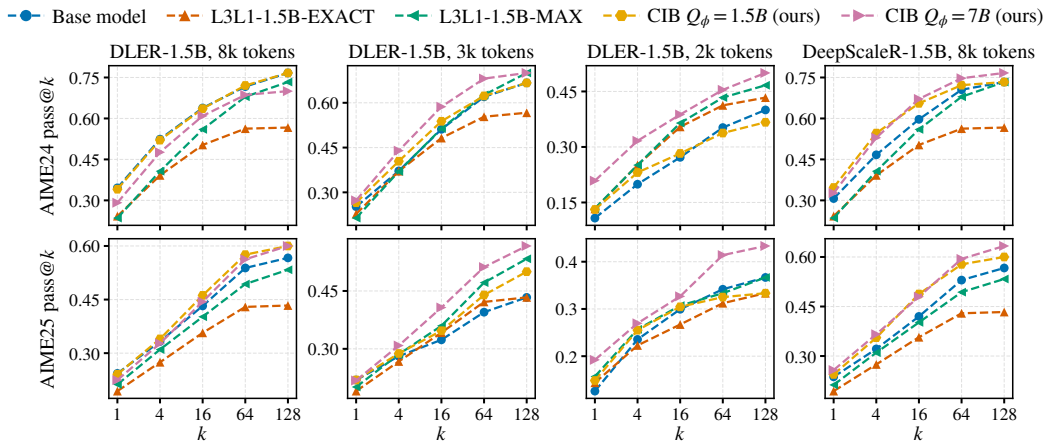

Figure 3: **Inference-time compute under different generation budgets.** Pass@k on AIME24 (top) and AIME25 (bottom). Columns 1–3: DLER-1.5B base model with a maximum generation length of 8K, 3K, and 2K tokens, respectively; column 4: DeepScaleR-1.5B base models with 8K tokens.

5.3 Comparison with a Uniform Prior

Matched–Length Comparison. A uniform per-token penalty also shortens trajectories, so penalizing redundant tokens alone does not explain the benefit of a *semantic* prior. To isolate its benefit, we train DeepScaler-1.5B and DLER-1.5B using a uniform-prior penalty, sweeping the penalty coefficient β_u over an order of magnitude and tuning hyperparameters at each value. These models provide a strong baseline tracing a clear length–accuracy trade-off. For each CIB model, we select the uniform-prior checkpoint closest in average length, breaking ties in favour of accuracy. With average lengths matched to within $\sim 1\%$, the semantic prior improves accuracy by 3.9 (β^-) and 2.8 (β^+) points on DeepScaler-1.5B and by 1.6 points on DLER-1.5B (Table 4, left). These gains hold across benchmarks in every comparison and are largest on GPQA, where CIB$_{\beta^-,1.5B}$ is both 6.6 points more accurate and 24% shorter (Table 6 in Appendix D). No uniform-prior checkpoint reaches the compression level of the 7B semantic-prior model, precluding a matched-length comparison in that regime.

Table 4: **Uniform versus semantic prior.** *Left:* accuracy and average completion length over the math benchmarks, comparing CIB models with the matched-length uniform-prior checkpoint ($\beta_u = 5 \times 10^{-4}$). *Right:* redundancy markers over 1432 prompt-paired completions, relative to the most accurate uniform-prior checkpoint ($\beta_u = 1 \times 10^{-4}$). Each arrow denotes uniform $\rightarrow$ CIB for the CIB setting specified in the same row. Density is available only for the β^- models.

	Matched-length comparison				Marker comparison	
	Uniform	CIB		Δ acc. ($\uparrow$)	Uniform $\rightarrow$ CIB	
Model	Acc. / len.	Setting	Acc. / len.		Per completion ($\downarrow$)	Per 1k tokens ($\downarrow$)
DeepScaler-1.5B	46.0 / 5224	β^-	49.9 / 5113	$+\mathbf{3.9}$	64.8 $\rightarrow$ 34.2	12.2 $\rightarrow$ 7.5
	46.0 / 5224	β^+	48.8 / 4870	$+\mathbf{2.8}$	64.8 $\rightarrow$ 42.2	–
DLER-1.5B	47.0 / 2813	β^-	48.6 / 2789	$+\mathbf{1.6}$	24.4 $\rightarrow$ 9.5	10.6 $\rightarrow$ 5.3

The Two Priors Compress Differently. The distinction lies in what the penalties can express. A uniform penalty depends only on trajectory length: removing any k tokens reduces the penalty by the same amount, regardless of their content. A content-dependent penalty instead allows the policy to optimize not only how much reasoning to retain, but also which reasoning to retain. To examine this distinction empirically, we measure both redundancy-marker counts and their per-token density. A proportional reduction in markers and tokens would leave marker density unchanged; a decrease in density indicates that markers become less prevalent within the retained reasoning. The right-hand columns of Table 4 show such a decrease. Here, the reference is the most accurate uniform-prior checkpoint, rather than the matched-length checkpoint. Relative to the uniform prior, the CIB prior reduces marker counts by 47.2% on DeepScaler-1.5B and 61.1% on DLER-1.5B across 1432 prompt-paired completions, while reducing marker density per token by 38.5% and 50.0%, respectively. Thus, markers are not merely fewer overall: they are also rarer per token. Relative to the uniform prior, the semantic prior changes the composition of the retained reasoning, rather than simply reducing its length. Appendix D provides the complete β_u sweeps, the matched-length selection procedure, marker definitions and per-category counts, and the full paired analysis for both model families.

6 Conclusions

In this work, we address efficient reasoning in LLMs by reframing "Budget Forcing" from an information-theoretic perspective. We identified the "Attention–Markov Incompatibility"—the structural inconsistency in applying standard Information Bottleneck principles to transformer architectures—and proposed a Conditional Information Bottleneck framework to resolve it. This yields a principled RL objective that subsumes existing length-based penalties as special cases (Propositions 4.1 and 4.2) while introducing a semantic token cost based on surprisal under a language model prior. Our experiments on reasoning benchmarks show that penalizing tokens by their semantic information content yields a more favorable accuracy–length trade-off than naive length penalties. By tuning β, one can traverse the Pareto frontier, achieving up to $\sim 48\%$ token reduction with $\lesssim 1.5\%$ accuracy degradation. We further find that stronger priors provide better redundancy estimates, enabling more aggressive compression with minimal performance loss—while introducing negligible training overhead ($\sim 6\%$ throughput reduction). These findings suggest that efficient inference requires moving beyond a "flat tax" on token count toward metrics that value computation based on its utility. More broadly, our framework's central contribution is a general recipe for optimizing reasoning efficiency: by varying the implementations of the verifier and the prior, practitioners can explore a broad design space tailored to specific downstream tasks and deployment constraints.

Limitations. While CIB compresses reasoning traces effectively, two limitations remain. First, compression quality hinges on the reference prior: a weak Q_ϕ flattens toward uniform and the semantic cost degenerates into a naive length penalty (Section 5), and cross-family priors (e.g. a Llama prior for a Qwen policy) add tokenization confounds we do not address—coarser-grained cost functions robust to vocabulary mismatch are a promising direction. Second, β and the prior capacity jointly set the operating point on the Pareto frontier; we tuned β for the 1.5B prior and did not re-tune at 7B, which may explain the slight accuracy gap, so a systematic study of the (β, Q_ϕ) interaction could yield further gains.

References

[1] P. Aggarwal and S. Welleck. L1: Controlling How Long A Reasoning Model Thinks With Reinforcement Learning. *arXiv preprint arXiv:2503.04697*, 2025. URL https://arxiv.org/pdf/2503.04697.

[2] W. U. Ahmad, S. Narenthiran, S. Majumdar, A. Ficek, S. Jain, J. Huang, V. Noroozi, and B. Ginsburg. Opencodereasoning: Advancing data distillation for competitive coding. *arXiv preprint arXiv:2504.01943*, 2025.

[3] A. A. Alemi, I. Fischer, J. V. Dillon, and K. Murphy. Deep Variational Information Bottleneck. In *International Conference on Learning Representations*, 2017. URL https://arxiv.org/pdf/1612.00410.

[4] J. Cheng and B. Van Durme. Compressed Chain of Thought: Efficient Reasoning Through Dense Representations. *arXiv preprint arXiv:2412.13171*, 2024. URL https://arxiv.org/pdf/2412.13171.

[5] Z. Cheng, D. Chen, M. Fu, and T. Zhou. Optimizing Length Compression in Large Reasoning Models. *arXiv preprint arXiv:2506.14755*, 2025. URL https://arxiv.org/pdf/2506.14755.

[6] S. Choi, Y. Kwon, and H. Lee. CAC-CoT: Connector-Aware Compact Chain-of-Thought for Efficient Reasoning Data Synthesis Across Dual-System Cognitive Tasks. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, 2025. URL https://aclanthology.org/2025.findings-emnlp.1062.

[7] Y. Cui, P. He, J. Zeng, H. Liu, X. Tang, Z. Dai, Y. Han, C. Luo, J. Huang, Z. Li, S. Wang, Y. Xing, J. Tang, and Q. He. Stepwise Perplexity-Guided Refinement for Efficient Chain-of-Thought Reasoning in Large Language Models. In *Findings of the Association for Computational Linguistics: ACL 2025*, 2025. URL https://aclanthology.org/2025.findings-acl.956.

[8] T. Han, Z. Wang, C. Fang, S. Zhao, S. Ma, and Z. Chen. Token-Budget-Aware LLM Reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, 2025. URL https://aclanthology.org/2025.findings-acl.1274.

[9] C. He, R. Luo, Y. Bai, S. Hu, Z. L. Thai, J. Shen, J. Hu, X. Han, Y. Huang, Y. Zhang, J. Liu, L. Qi, Z. Liu, and M. Sun. OlympiadBench: A Challenging Benchmark for Promoting AGI with Olympiad-Level Bilingual Multimodal Scientific Problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024. URL https://arxiv.org/pdf/2402.14008.

[10] Y. He, Y. Zhu, M. Shi, W. Zheng, L. Su, X. Wang, Q. Guo, and J. Li. IAPO: Information-Aware Policy Optimization for Token-Efficient Reasoning. *arXiv preprint arXiv:2602.19049*, 2026. URL https://arxiv.org/pdf/2602.19049.

[11] B. Hou, Y. Zhang, J. Ji, Y. Liu, K. Qian, J. Andreas, and S. Chang. Thinkprune: Pruning long chain-of-thought of llms via reinforcement learning. *arXiv preprint arXiv:2504.01296*, 2025.

[12] C. Huang, Z. Zhang, and C. Cardie. HAPO: Training Language Models to Reason Concisely via History-Aware Policy Optimization. *arXiv preprint arXiv:2505.11225*, 2025. URL https://arxiv.org/pdf/2505.11225.

[13] N. Jain, K. Han, A. Gu, W.-D. Li, F. Yan, T. Zhang, S. Wang, A. Solar-Lezama, K. Sen, and I. Stoica. LiveCodeBench: Holistic and Contamination Free Evaluation of Large Language Models for Code. *arXiv preprint arXiv:2403.07974*, 2024. URL https://arxiv.org/pdf/2403.07974.

[14] N. Jali, A. Nayak, and G. Joshi. Not All Turns Are Equally Hard: Adaptive Thinking Budgets For Efficient Multi-Turn Reasoning. *arXiv preprint arXiv:2604.05164*, 2026. URL https://arxiv.org/pdf/2604.05164.

[15] S. Jiang, Y. Liao, Y. Zhang, Y. Wang, and Y. Wang. Overthinking Reduction with Decoupled Rewards and Curriculum Data Scheduling. *arXiv preprint arXiv:2509.25827*, 2025. URL https://arxiv.org/pdf/2509.25827.

[16] S. Lei, Z. Cheng, K. Jia, and D. Tao. Revisiting LLM Reasoning via Information Bottleneck. *arXiv preprint arXiv:2507.18391*, 2025. URL https://arxiv.org/pdf/2507.18391.

[17] A. Lewkowycz, A. Andreassen, D. Dohan, E. Dyer, H. Michalewski, V. Ramasesh, A. Slone, C. Anil, I. Schlag, T. Gutman-Solo, Y. Wu, B. Neyshabur, G. Gur-Ari, and V. Misra. Solving Quantitative Reasoning Problems with Language Models. In *Advances in Neural Information Processing Systems*, volume 35, pages 3843–3857, 2022. URL https://arxiv.org/pdf/2206.14858.

[18] C. T. Li. Channel simulation: Theory and applications to lossy compression and differential privacy. *Foundations and Trends® in Communications and Information Theory*, 21(6):847–1106, 2024.

[19] J. Li, W. Zhao, Y. Zhang, and C. Gan. Steering LLM Thinking with Budget Guidance. *arXiv preprint arXiv:2506.13752*, 2025. URL https://arxiv.org/pdf/2506.13752.

[20] Z. Li, Q. Dong, J. Ma, D. Zhang, K. Jia, and Z. Sui. SelfBudgeter: Adaptive Token Allocation for Efficient LLM Reasoning. *arXiv preprint arXiv:2505.11274*, 2025. URL https://arxiv.org/pdf/2505.11274.

[21] H. Lightman, V. Kosaraju, Y. Burda, H. Edwards, B. Baker, T. Lee, J. Leike, J. Schulman, I. Sutskever, and K. Cobbe. Let's Verify Step by Step. *arXiv preprint arXiv:2305.20050*, 2023. doi: 10.48550/arXiv.2305.20050. URL https://arxiv.org/pdf/2305.20050.

[22] S.-Y. Liu, X. Dong, X. Lu, S. Diao, M. Liu, M.-H. Chen, H. Yin, Y.-C. F. Wang, K.-T. Cheng, Y. Choi, J. Kautz, and P. Molchanov. DLER: Doing Length pEnalty Right — Incentivizing More Intelligence per Token via Reinforcement Learning. *arXiv preprint arXiv:2510.15110*, 2025. URL https://arxiv.org/pdf/2510.15110.

[23] H. Luo, L. Shen, H. He, Y. Wang, S. Liu, W. Li, N. Tan, X. Cao, and D. Tao. O1-Pruner: Length-Harmonizing Fine-Tuning for O1-Like Reasoning Pruning. *arXiv preprint arXiv:2501.12570*, 2025. URL https://arxiv.org/pdf/2501.12570.

[24] M. Luo, S. Tan, J. Wong, X. Shi, W. Y. Tang, M. Roongta, C. Cai, J. Luo, L. E. Li, R. A. Popa, and I. Stoica. DeepScaleR: Surpassing O1-Preview with a 1.5b Model by Scaling RL. Notion Blog Post, 2025.

[25] S. Ma, B. Gao, X. Jia, S. Qin, T. Li, K. Ma, X. Jia, W. Ren, and Y. Liu. ODAR: Principled Adaptive Routing for LLM Reasoning via Active Inference. *arXiv preprint arXiv:2602.23681*, 2026. URL https://arxiv.org/pdf/2602.23681.

[26] M. Mao, B. Yin, Y. Zhu, and X. Fang. Early Stopping Chain-of-Thoughts in Large Language Models. *arXiv preprint arXiv:2509.14004*, 2025. URL https://arxiv.org/pdf/2509.14004.

[27] Mathematical Association of America. American Mathematics Competitions (AMC) 10/12 2023, 2023. URL https://maa.org/.

[28] Mathematical Association of America. American Invitational Mathematics Examination (AIME) 2024, 2024. URL https://maa.org/. Problems I and II.

[29] Mathematical Association of America. American Invitational Mathematics Examination (AIME) 2025, 2025. URL https://maa.org/. Problems I and II.

[30] I. Moshkov, D. Hanley, I. Sorokin, S. Toshniwal, C. Henkel, B. Schifferer, W. Du, and I. Gitman. Aimo-2 winning solution: Building state-of-the-art mathematical reasoning models with openmathreasoning dataset. *arXiv preprint arXiv:2504.16891*, 2025.

[31] N. Muennighoff, Z. Yang, W. Shi, X. L. Li, L. Fei-Fei, H. Hajishirzi, L. Zettlemoyer, P. Liang, E. Candès, and T. Hashimoto. s1: Simple Test-Time Scaling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 20286–20332, Suzhou, China, November 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.1025. URL https://aclanthology.org/2025.emnlp-main.1025/.

[32] D. Rein, B. L. Hou, A. C. Stickland, J. Petty, R. Y. Pang, J. Dirani, J. Michael, and S. R. Bowman. GPQA: A Graduate-Level Google-Proof Q&A Benchmark. *arXiv preprint arXiv:2311.12022*, 2023. URL https://arxiv.org/pdf/2311.12022.

[33] M. Renze and E. Guven. The Benefits of a Concise Chain of Thought on Problem-Solving in Large Language Models. *arXiv preprint arXiv:2401.05618*, 2024. URL https://arxiv.org/pdf/2401.05618.

[34] A. M. Saxe, Y. Bansal, J. Dapello, M. Advani, A. Kolchinsky, B. D. Tracey, and D. D. Cox. On the Information Bottleneck Theory of Deep Learning. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=ry_WPG-A-.

[35] K. Sethi. Don't Overthink It: Inter-Rollout Action Agreement as a Free Adaptive-Compute Signal for LLM Agents. *arXiv preprint arXiv:2604.08369*, 2026. URL https://arxiv.org/pdf/2604.08369.

[36] Z. Shao, P. Wang, Q. Zhu, R. Xu, J. Song, X. Bi, H. Zhang, M. Zhang, Y. Li, Y. Wu, and D. Guo. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. *arXiv preprint arXiv:2402.03300*, 2024. doi: 10.48550/arXiv.2402.03300. URL https://arxiv.org/pdf/2402.03300.

[37] R. Shwartz-Ziv and N. Tishby. Opening the Black Box of Deep Neural Networks via Information. *arXiv:1703.00810 [cs]*, Mar. 2017. URL https://arxiv.org/pdf/1703.00810.

[38] N. Tishby, F. C. Pereira, and W. Bialek. The Information Bottleneck Method. In *Allerton Conference*, 1999. URL https://arxiv.org/pdf/physics/0004057.

[39] J.-F. Ton, M. F. Taufiq, and Y. Liu. Understanding Chain-of-Thought in LLMs through Information Theory. *arXiv preprint arXiv:2411.11984*, 2024. URL https://arxiv.org/pdf/2411.11984.

[40] J. Wang, S. Jain, D. Zhang, B. Ray, V. Kumar, and B. Athiwaratkun. Reasoning in Token Economies: Budget-Aware Evaluation of LLM Reasoning Strategies. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 19916–19939, 2024.

[41] J. Wang, K. Q. Lin, J. Cheng, and M. Z. Shou. Think or Not? Selective Reasoning via Reinforcement Learning for Vision-Language Models. *arXiv preprint arXiv:2505.16854*, 2025. URL https://arxiv.org/pdf/2505.16854.

[42] X. Wang, J. McInerney, L. Wang, and N. Kallus. Entropy After </think> for Reasoning Model Early Exiting. *arXiv preprint arXiv:2509.26522*, 2025. URL https://arxiv.org/pdf/2509.26522.

[43] Y. Wang, X. Ma, G. Zhang, Y. Ni, A. Chandra, S. Guo, W. Ren, A. Arulraj, X. He, Z. Jiang, et al. MMLU-Pro: A More Robust and Challenging Multi-Task Language Understanding Benchmark. *Advances in Neural Information Processing Systems*, 37:95266–95290, 2024. URL https://arxiv.org/pdf/2406.01574.

[44] C. Wei, J.-j. Kim, L. Zhang, S. Chen, and N. F. Chen. InfoDensity: Rewarding Information-Dense Traces for Efficient Reasoning. *arXiv preprint arXiv:2603.17310*, 2026. URL https://arxiv.org/pdf/2603.17310.

[45] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, and D. Zhou. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022. URL https://arxiv.org/pdf/2201.11903.

[46] X. Wu, Y. Yan, S. Lyu, L. Wu, Y. Qiu, Y. Shen, W. Lu, J. Shao, J. Xiao, and Y. Zhuang. LAPO: Internalizing Reasoning Efficiency via Length-Adaptive Policy Optimization. *arXiv preprint arXiv:2507.15758*, 2025. URL https://arxiv.org/pdf/2507.15758.

[47] A. D. Wyner and J. Ziv. The Rate-Distortion Function for Source Coding with Side Information at the Decoder. *IEEE Transactions on Information Theory*, 22(1):1–10, 1976. doi: 10.1109/TIT.1976.1055508.

[48] H. Xia, C. T. Leong, W. Wang, Y. Li, and W. Li. TokenSkip: Controllable Chain-of-Thought Compression in LLMs. *arXiv preprint arXiv:2502.12067*, 2025. URL https://arxiv.org/pdf/2502.12067.

[49] S. Xu, W. Xie, L. Zhao, and P. He. Chain of Draft: Thinking Faster by Writing Less. *arXiv preprint arXiv:2502.18600*, 2025. URL https://arxiv.org/pdf/2502.18600.

[50] A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei, et al. Qwen2.5 Technical Report. *arXiv preprint arXiv:2412.15115*, 2024. URL https://arxiv.org/pdf/2412.15115.

[51] J. Zhang, Y. Sun, T. Leng, J. Shen, Z. Liu, P. P. Liang, and H. Zhang. When Reasoning Meets Its Laws. In *NeurIPS 2025 Workshop on Efficient Reasoning*, 2025. URL https://openreview.net/forum?id=lWjcbodr4M.

A Conditional Information Bottleneck

Preliminaries. We denote the query, answer, and reasoning traces respectively by the random variables X, Y and Z. In this work, we assume that $X, Y, Z \in \mathcal{X}^*$, where $\mathcal{X}^*$ is the set of all finite sequences of tokens in the token space $\mathcal{X}$. The underlying probability space of these random variables is given by $(\mathcal{X}^*, \Sigma^*)$, where Σ^* is the co-product σ-algebra using the σ-algebras on each $\mathcal{X}^n$. The space $\mathcal{X}$ is assumed to be discrete. The entropy of the random variable X, the conditional entropy of Y given X, and the mutual information between X and Y are defined as:

$$\begin{aligned} H(X) &:= \mathbb{E}_{X\sim P_X}(-\log P(X)), \\ H(Y|X) &:= \mathbb{E}_{(X,Y)\sim P_{XY}}(-\log P(Y|X)), \\ I(X;Y) &:= H(X) - H(X|Y). \end{aligned}$$

Formulation. Consider the query-answer pair (X, Y). The function of reasoning is to generate Z such that the LLM probability $\pi_\theta(Y|Z, X)$ is maximized. Budget forcing aims at compressing Z.

In the classical information bottleneck, this problem is formulated as maximizing the information gain $I(Z;Y)$ while minimizing the redundancy with $I(X;Z)$, yielding the following optimization problem:

$$\min_{\pi_\theta(Z|X)} I(X;Z) - \beta I(Y;Z),$$

under the Markov assumption $Y \leftrightarrow X \leftrightarrow Z$ and the marginal probability constraint $\sum_z \pi_\theta(z|x) = 1, \forall x$. In information bottleneck literature, the mutual information $I(X;Z)$ is called rate or complexity, while $I(Y;Z)$ is called relevance or information. We use the terms *minimality* and *sufficiency* in this paper.

In the context of reasoning, the Markov chain $Y \leftrightarrow X \leftrightarrow Z$ does not hold, namely $\pi_\theta(Y|X, Z) \neq \pi_\theta(Y|X)$ because the dense attention mechanism breaks the Markov relation, and the response depends on both the query and the reasoning trace.

There is another subtle issue with the classical information bottleneck setup. The outcome of the optimization problem is $\pi_\theta(z|x)$. The Markov property enables us to generate y based on z using $p(y|z) = \sum_x p(y|x)p(x|z)$, which implicitly assumes the knowledge of the conditional probability $p(y|x)$ at decoding time. Without Markov property, this relation cannot be used. Besides, $p(y|x)$ is *not* available at the decoding time, and it is unknown in the context of reasoning. The goal of reasoning trace z is to enable the simulation of $p(y|x)$ using $\pi_\theta(z|x)\pi_\theta(y|z, x)$, which points toward a connection with *channel simulation* literature [18].

We would like to maximize the information gain of the reasoning trace Z with the knowledge of the query X. We can measure the gain using the conditional mutual information $I(Y;Z|X)$. The conditional information bottleneck version is as follows:

$$\min_{P(Z|X,Y)} I(X;Z) - \beta I(Y;Z|X),$$

where we need the following marginal probability constraints to be satisfied:

$$\sum_z P(z|x, y) = 1, \forall x, y.$$

If we cast the problem as a maximization problem, the final optimization problem in terms of $P(Z|X, Y)$ is as follows:

$$\begin{aligned} \max_{P(Z|X,Y)} & \sum_{x,y,z} P(x, y)P(z|x, y) \log P(y|x, z) - \beta \sum_{x,z} P(x)P(z|x) \log \frac{P(z|x)}{P(z)} \\ \text{s.t.} & \sum_z P(z|x, y) = 1, \forall x, y. \end{aligned} \tag{6}$$

The dependence on $P(z|x, y)$ is implicit in various distributions like $P(z|x), p(z)$ and $p(y|x, z)$, and we can solve this problem using an iterative algorithm, similar to Blahut-Arimoto algorithm, proposed in [38]. Given that z is from the space of reasoning traces, it is not tractable to use the same approach.

We would like to remark that the information bottleneck problem is an instance of lossy compression under log-loss distortion metric. In this sense, the conditional information bottleneck is akin to lossy compression with side information, which was studied by Wyner and Ziv [47] under different settings.

Practical Implementation. The information bottleneck optimization in deep learning is directly intractable, and the approximate bounds are used for training such variational information bottleneck [3]. In practice, for training LLMs, we do not directly optimize over $P(Z|Y,X)$ but rather optimize the model parameter θ.

The parameter θ controls the two probabilities $\pi_\theta(z|x)$ and $\pi_\theta(y|x,z)$, both the inference part of the AR generative model (instead of $P(Z|Y,X)$. We solve the following optimization problem:

$$\begin{aligned} \max_{\theta} \quad & \mathcal{L}_{\text{CIB}}(\theta) = I(Y;Z|X) - \beta I(X;Z) \\ \text{s.t.} \quad & P(y|x) = \sum_z \pi_\theta(y|x,z)\pi_\theta(z|x), \quad \forall x, y. \end{aligned} \tag{7}$$

The last constraint should be satisfied to have a valid probability distribution. Since all the probabilities $\pi_\theta(\cdot|\cdot)$ are parametrized to sum up to one, we do not need to explicitly add the constraint.

Note that we can write $P(z|x,y)P(y|x) = \pi_\theta(y|x,z)\pi_\theta(z|x)$. In other words, we can obtain a valid $P(z|x,y)$ from $\pi_\theta(y|x,z)$ and $\pi_\theta(z|x)$, and vice versa. Therefore, the above optimization problem is just a reparameterization of the original information bottleneck problem and yields the same optimal value.

There are some challenges with the above optimization problem. First, the conditional distribution $P(y|x)$ is unknown in general, and we have access to it only through the samples. Second, we should approximate the information theoretic quantities, namely the sufficiency term $I(Y;Z|X)$ and the minimality term $I(X;Z)$. We try to address these challenges below.

Sufficiency term. Consider the first term in the objective function. We can write it as a function of the optimization parameters $\pi_\theta(y|x,z), \pi_\theta(z|x)$ as follows:

$$\begin{aligned} I(Y;Z|X) &= \sum_{x,y,z} P(x,y)P(z|x,y)\log\frac{\pi_\theta(y|x,z)}{P(y|x)} \\ &= \sum_{x,y,z} P(x,y)\pi_\theta(z|x)\frac{\pi_\theta(y|x,z)}{P(y|x)}\log\frac{\pi_\theta(y|x,z)}{P(y|x)} \\ &\geq \sum_{x,y,z} P(x,y)\pi_\theta(z|x)\log\frac{\pi_\theta(y|x,z)}{P(y|x)} \end{aligned}$$

where we used $x\log x \geq \log x$ in the last step. Therefore, we can maximize the lower bound and approximate it further using the query-answer samples (x_i, y_i). The first term of the optimization problem is:

$$\sum_{i=1}^{m} \mathbb{E}_{Z\sim\pi_\theta(\cdot|x_i)}[\log \pi_\theta(y_i|x_i,Z)]. \tag{8}$$

We can also approximate the bound using variational approximation of [3]. We introduce a *verifier model* $Q_\rho(y|x,z)$ as variational parameter:

$$\begin{aligned} I(Y;Z|X) &= \sum_{x,y,z} P(x,y)P(z|x,y)\log\frac{\pi_\theta(y|x,z)}{P(y|x)} \\ &= \sum_{x,y,z} P(x,y)P(z|x,y)\log\frac{\pi_\theta(y|x,z)Q_\rho(y|z,x)}{P(y|x)Q_\rho(y|z,x)} \\ &= \sum_{x,y,z} P(x,y)P(z|x,y)\log\frac{Q_\rho(y|z,x)}{P(y|x)} + \mathbb{E}_{(X,Z)}D_{KL}(\pi_\theta(\cdot|X,Z)\|Q_\rho(\cdot|X,Z)) \\ &\geq \sum_{x,y,z} P(x,y)P(z|x,y)\log\frac{Q_\rho(y|z,x)}{P(y|x)} \\ &= \sum_{x,y,z} P(x,y)\pi_\theta(z|x)\frac{\pi_\theta(y|x,z)}{P(y|x)}\log\frac{Q_\rho(y|z,x)}{P(y|x)} \end{aligned}$$

Throughout the paper, we assume that there is always a unique answer to each query. Using this assumption, we can lower bound the last step as follows:

$$\sum_{x,y,z} P(x,y)\pi_\theta(z|x)\frac{\pi_\theta(y|x,z)}{P(y|x)}\log\frac{Q_\rho(y|z,x)}{P(y|x)}$$
$$\geq \sum_{x,y=y_{\text{true}}(x),z} P(x,y)\pi_\theta(z|x)\log Q_\rho(y|z,x),$$

where we used the assumption that $P(y|x) = \delta(y - y_{\text{true}}(x))$. Finally, we can maximize the following objective function for the sufficiency term:

$$\sum_{i=1}^{m} \mathbb{E}_{Z\sim\pi_\theta(\cdot|x_i)}[\log Q_\rho(y_i|x_i,Z)].$$

Minimality term. For the minimality term, we use a variational approximation to find a variational bound similar to [3]. This upper bound combined with the above lower bound provides a general lower bound on the conditional information bottleneck objective which we try to maximize. First note that:

$$I(X;Z) = \sum_{x,z} P(x)\pi_\theta(z|x)\log\frac{\pi_\theta(z|x)}{P(z)}.$$

There is a dependence on $P(z)$ which requires marginalization over x and is intractable. The derivation of the variational lower bound is quite standard:

$$\begin{aligned} I(X;Z) &= \sum_{x,z} P(x)\pi_\theta(z|x)\log\frac{\pi_\theta(z|x)}{P(z)} \\ &= \sum_{x,z} P(x)\pi_\theta(z|x)\log\frac{\pi_\theta(z|x)}{Q_\phi(z)} - D_{KL}(P(Z)\|Q_\phi(Z)) \\ &\leq \sum_{x,z} P(x)\pi_\theta(z|x)\log\frac{\pi_\theta(z|x)}{Q_\phi(z)}. \end{aligned}$$

The variational distribution $Q_\phi(\cdot)$ is supposed to capture the distribution over reasoning traces without conditioning on the prompt.

The final optimization problem consists of finding a policy $\pi_\theta(z|x)$ that maximizes the returns defined from the above approximate bounds using Q_ϕ and Q_ρ.

Marginal probability constraint. Let's consider again the constraint for the conditional information bottleneck:

$$P(y|x) = \sum_z \pi_\theta(y|x,z)\pi_\theta(z|x).$$

As we mentioned above, the conditional probability is unknown. Now, assume that for each query there is a unique answer: $P(y|x) = \delta(y - y_{\text{true}}(x))$. In this case, the marginal probability constraint holds only if $\pi_\theta(y|x,z)$ is also equal to $\delta(y - y_{\text{true}}(x))$, namely:

$$\pi_\theta(y_{\text{true}}(x)|x,z) = 1, \tag{9}$$

We do not need explicitly include this constraint in the optimization problem, because it amounts to maximizing the probability of the correct answer under $\pi_\theta(y|x,z)$ which is already part of the optimization problem.

B Additional Analysis

Efficiency gain. To quantify the trade-off between reasoning performance and computational cost, we define two metrics: the *Compression Factor* $= 1 - \ell_{\text{CIB}}/\ell_{\text{base}}$, measuring the relative reduction in completion length, and the *Accuracy Ratio* $= \mathcal{A}_{\text{CIB}}/\mathcal{A}_{\text{base}}$, normalizing performance against the baseline. Figure 4 (left) plots both across architectures and benchmarks. The upper half-plane

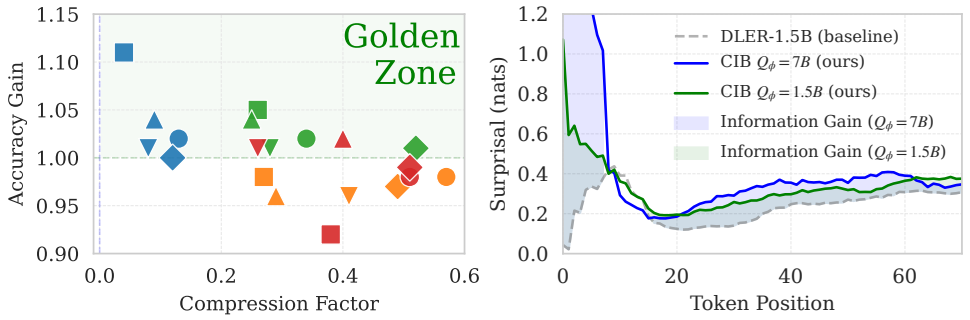

Figure 4: **Left: Efficiency Gain.** Efficiency gain of CIB across diverse benchmarks and models. Points falling in the upper half-plane ("Golden Zone") exhibit strictly superior efficiency, achieving higher information density with reduced computational cost. Colors denote models: $\text{CIB}^{DLER}_{Q_\phi=1.5B}$ (●), $\text{CIB}^{DLER}_{Q_\phi=7B}$ (●), $\text{CIB}^{DeepSc.}_{Q_\phi=1.5B}$ (●), $\text{CIB}^{DeepSc.}_{Q_\phi=7B}$ (●). Shapes denote tasks: MATH500 (●), AIME24 (■), AIME25 (▲), Minerva (♦), Olympiad (▼). **Right: Information Density Profile.** Token-wise surprisal evaluated against the baseline language prior. Lower surprisal indicates predictable, low-information filler.

("Golden Zone") holds models that are strictly superior in both speed and accuracy, while the bottom-right offers significant speedups where partial accuracy degradation is acceptable. Filtering for the Golden Zone therefore selects models that are "smarter" and faster rather than merely truncated. Note that each model's gains are normalized to its own baseline. For visual clarity, the figure focuses on DLER-1.5B and DeepScaleR across the five math benchmarks and both prior sizes.

Information density. To understand *how* CIB compresses, we analyze the token-wise surprisal $-\log p(z_t \mid z_{<t}, x)$ measured against a frozen reference model. In Figure 4 (right), the baseline profile (dashed gray) shows valleys of low surprisal (≈ 0.1 nats), i.e. highly predictable sequences carrying negligible unique information, whereas the CIB profiles (solid blue and green) maintain a higher floor ($\gtrsim 0.2$ nats). By penalizing cumulative low-utility transitions, the objective acts as a high-pass semantic filter that excises predictable filler while preserving high-entropy peaks. CIB thus compresses not by random truncation but by raising the average information rate, distilling reasoning traces to their essential computational content.

C CoT Qualitative Comparison: Pruning Reasoning Redundancy

To examine the nature of the compression induced by our semantic prior, we visualize qualitative differences between baseline traces and CIB-generated traces across a range of reasoning tasks (see Figures 5–7). We find that the semantic prior does not merely truncate outputs; instead, it changes *which* computations are expressed in the trace. Concretely, by imposing an information-cost on the reasoning tokens (via the prior surprisal) while preserving task success through the sufficiency objective, CIB penalizes computation that offers low marginal information regarding the target Y. This effect manifests through three recurring mechanisms:

- Induction of Algorithmic Generalization.
  Perhaps most notably, the information bottleneck biases the model toward theoretically superior solution paths. In geometric reasoning (Figure 5), while the baseline defaults to brute-force coordinate calculations via the Pythagorean theorem, the CIB model converges on a concise trigonometric identity ($\sin T = \cos R$). This suggests that minimizing the redundant computation under the semantic prior of the reasoning trace naturally selects for abstract, elegant proofs, as these represent the most compressed description of the transformation from prompt X to answer Y.
- Suppression of Stochastic Exploration and Verification Bloat.
  Baseline models typically adopt a high-entropy strategy characterized by reasoning redundancy, utilizing conversational scaffolding and unstructured exploration. For instance, in arithmetic search tasks (Figure 6), the baseline explicitly calculates invalid candidates (e.g.,

98^3) before testing the correct one. Similarly, in constraint satisfaction problems (Figure 7), the baseline engages in tautological self-checks (e.g., verifying that positive lengths satisfy $x > 0$). CIB eliminates these low-utility branches. By penalizing the cumulative surprisal of the chain, the policy shifts from "exploratory thinking" to "efficient execution," treating valid derivations as terminal states rather than triggering redundant self-doubt loops.

- Semantic Filtering of Syntactic Artifacts.
  CIB acts as a semantic filter that separates essential state information from syntactic artifacts. As shown in Figure 5, when presented with raw code metadata (Asymptote), the baseline expends significant budget on "verbal parsing"—reading the code aloud without progressing the state. The compressed policy bypasses this verbalization, extracting the underlying geometric conditions directly.

Input Prompt

Problem: In $\triangle RST$, shown, $\sin R = \frac{2}{5}$. What is $\sin T$?
[Asymptote Code Segment]: `pair R,S,T; T = (0,0); S = (2,0); R = (2,sqrt(21)); ...`

Baseline Response (Standard CoT) - 1685 tokens

Okay, so I have this triangle RST... The Asymptote code draws a right-angled triangle... The Asymptote code defines points T at (0,0)... **Triangle RST is right-angled at S.** So, angle at S is 90 degrees... Wait, the label is on (R+T)/2... **RT is the hypotenuse, length 5.**
Let me assign the sides... Given sin R = 2/5... **sin R = ST / RT = 2/5. RT is 5, so ST = 2.**
Now, since it's a right-angled triangle at S, we can use Pythagoras... **$RS^2 + 2^2 = 5^2$... RS = $\sqrt{21}$.**
Now, we need to find sin T... Wait, no. Wait, angle T is at point T... Wait, maybe I... **sin T = opposite / hypotenuse... RS / RT. sin T = $\sqrt{21}$/5.**

Ours: CIB - 154 tokens

First, observe that **triangle RST is a right-angled triangle** at point S. Given that **sin R = 2/5.** In a right-angled triangle, the sum of angles R and T is 90 degrees, so angle T is complementary to angle R. Using the sine of complementary angles, **sin T = cos R.**
Since sin R = 2/5, cos R = $\sqrt{1-(2/5)^2} = \sqrt{21/25} = \sqrt{21}/5$. Therefore, **sin T = $\sqrt{21}$/5.**

Figure 5: **Qualitative comparison on geometry reasoning. Top:** Prompt. **Middle:** the Baseline trace is dominated by redundant "verbal parsing" of the input code and repetitive self-correction loops (highlighted in red). **Bottom:** The CIB trace successfully filters this syntactic noise. Notably, the information constraint induces a shift in strategy: CIB bypasses the lengthy coordinate calculation favored by the baseline, converging instead on a concise trigonometric identity.

Input Prompt

Problem: When the two-digit integer "AB" is cubed, the value is $912,673$. What is $A + B$?

Baseline Response (Standard CoT) - 965 tokens

Okay, so I have this problem... I need to find A + B... Hmm... Wait, let me think. **AB is a two-digit integer... between 10 and 99.**
First, maybe I should find the cube root... I know that 100^3 is 1,000,000... Maybe 99^3 is about 970299... 98^3 is 98 * 98 * 98. Let me calculate that. 9604 * 98... 960400 - 19208 = 941192. So 98^3 is 941192, which is larger than 912673.
What about 97^3? Let me compute that. **97 * 97 is 9409... 9409 * 97 = 912,673.** Oh, wait! 97 cubed is 912673. So AB is 97.
Wait wait, let me double-check. 97 cubed is 912673... Yes, correct. **So AB is 97, so A is 9, B is 7. Therefore, A + B is 9 + 7 = 16.**

Ours: CIB - 210 tokens

To determine the value of A + B, we start by noting that **the two-digit integer AB is a number between 10 and 99.** When AB is cubed, the result is 912,673.
Taking the cube root of 912,673 gives an approximate value. The cube root of 912,673 is approximately 97, since **97^3 = 912,673.**
Thus, **AB is 97.** The digits A and B are 9 and 7, respectively. Adding them together, **A + B equals 9 + 7 = 16.**

Figure 6: **Qualitative comparison on arithmetic search. Top:** Prompt. **Middle:** Baseline model engages in inefficient trial-and-error, explicitly calculating the incorrect candidate 98^3 (highlighted in red) and engaging in redundant self-verification loops. **Bottom:** the CIB model (bottom) suppresses this exploratory computation, converging directly on the correct candidate (97) and reducing the token count by $\sim$78% without loss of accuracy.

Input Prompt

Problem: Two sides of a triangle are each 8 units long. If the third side has a whole number length, what is the greatest possible perimeter, in units, for the triangle?

Baseline Response (Standard CoT)

Okay, so I have this problem... Hmm, let me think about how to approach this. **Triangle inequality theorem... sum of any two sides $>$ third.**
$8 + 8 > c$**. That means** $16 > c$**...**c **can be at most 15.**
But wait, wait. Let me think again... The triangle inequality also requires... $8 + c > 8$. Well, that simplifies to $c > 0$, which is always true...
But wait, let me double-check. Let's plug c = 15 into the triangle inequalities... 1. $8 + 8 > 15$? Yes... 2. $8 + 15 > 8$? Yes... So all the inequalities are satisfied.
Wait, but hold on. Let me make sure I didn't miss any constraints... c can be 15... **Perimeter is 8 + 8 + 15 = 31 units.**

Ours: CIB

To determine the greatest possible perimeter... we apply the **triangle inequality theorem.** The sum of any two sides must exceed the third side. Let the third side be 'x'.
The inequalities are: 1. $8 + 8 > x \rightarrow x < 16$ 2. $8 + x > 8 \rightarrow x > 0$ (always true)
Thus, x must be less than 16 and a positive integer. **The maximum integer less than 16 is 15.** Therefore, the **maximum perimeter is 8 + 8 + 15 = 31.**

Figure 7: **Qualitative comparison on constraint satisfaction.Top:** Prompt. **Middle:** Baseline trace is characterized by "verification bloat." Despite correctly deriving the constraint ($c < 16$) early on, the model expends tokens checking tautologies ($8 + c > 8$) and re-verifying its own conclusions (highlighted in red). **Bottom:** CIB trace (bottom) retains the necessary constraint logic but eliminates the redundant self-auditing loops, trusting the derivation immediately.

D Uniform Prior Ablation: Length and Reasoning Composition

Both a uniform per-token penalty and the CIB semantic prior discourage verbose trajectories. Suppressing tokens that add cost without improving accuracy therefore does not, by itself, explain the semantic prior's benefit. We investigate this difference in the DeepScaler-1.5B and DLER-1.5B families through three analyses: a uniform-prior β_u sweep with per-value hyperparameter tuning, an accuracy comparison at comparable average completion length, and a paired trajectory analysis over 1432 samples across five benchmarks.

Length-dependent versus content-dependent cost. Under a uniform prior, trajectory cost depends only on token count: removing any k tokens yields the same cost reduction, regardless of their content. A semantic prior makes cost content-dependent, allowing the policy to optimize not only completion length but also the reasoning it retains. We test whether this distinction is reflected in trajectory composition by measuring redundant-reasoning markers and their density per token. Proportional shortening would leave marker density unchanged; a lower density indicates a change in composition beyond the reduction in length.

Surprisal weighting may also penalize load-bearing reasoning tokens. However, the penalty is balanced against the accuracy reward, so retaining such tokens can justify their cost. Higher accuracy at comparable average length provides evidence against indiscriminate removal of useful reasoning (Table 6 and Table 10).

D.1 DeepScaler-1.5B

Here, $CIB_{\beta^-,1.5B}$ and $CIB_{\beta^+,1.5B}$ denote the two DeepScaler-1.5B semantic-prior models reported in the paper.

Uniform-prior tuning. We sweep β_u over one order of magnitude (Table 5). Below this range, the penalty is negligible; above it, the penalty dominates and training collapses within the first few iterations. The uniform prior was more sensitive to other hyperparameters than the semantic prior. For each β_u, we therefore ran a small grid search over learning rate, KL-regularization strength, and other hyperparameters, and report the best checkpoint. Training used a maximum generation length of 8192 tokens, as in the semantic-prior runs; evaluation followed the paper's protocol, allowing up to 32K new tokens.

Table 5: **DeepScaler-1.5B uniform-prior sweep.** Accuracy / average completion length on five reasoning benchmarks.

Setting	AIME24	AIME25	GPQA	MATH500	Olympiad	mean
Uniform prior, β_u sweep						
$\beta_u = 5 \times 10^{-5}$	34.6 / 9261	25.8 / 8715	35.9 / 7446	85.8 / 3067	54.4 / 5871	47.3 / 6872
$\beta_u = 1 \times 10^{-4}$	37.1 / 9025	27.7 / 8582	32.3 / 7257	88.6 / 3193	54.6 / 5756	48.1 / 6763
$\beta_u = 3 \times 10^{-4}$	34.6 / 9050	26.5 / 8179	31.8 / 7528	86.4 / 2950	55.1 / 5613	46.9 / 6664
$\beta_u = 5 \times 10^{-4}$	36.9 / 6473	24.8 / 7002	31.8 / 6051	84.2 / 2270	52.2 / 4322	46.0 / 5224

D.1.1 Matched-length comparison

We selected the uniform checkpoint closest in average completion length to the semantic-prior models, choosing the highest-accuracy checkpoint among those at comparable length. The selected checkpoint, $\beta_u = 5 \times 10^{-4}$ (Table 5), averages 5224 tokens, compared with 5113 for $CIB_{\beta^-,1.5B}$ and 4870 for $CIB_{\beta^+,1.5B}$. It is thus closely matched to the former and 7% longer than the latter. No uniform checkpoint reached the 7B semantic-prior model's average length of 3529 tokens, so we exclude that model from the matched-length comparison. We report the results in Table 6

The tuned uniform penalty provides a strong baseline, but the semantic prior achieves a better accuracy–length trade-off. At slightly shorter average lengths, $CIB_{\beta^-,1.5B}$ and $CIB_{\beta^+,1.5B}$ improve mean accuracy by 3.9 and 2.8 points, respectively, with gains on all five benchmarks. AIME24 and AIME25 each contain only 30 problems, so their individual differences should be interpreted

Table 6: **DeepScaler-1.5B matched-length comparison with $CIB_{\beta^-,1.5B}$.** The uniform baseline uses $\beta_u = 5 \times 10^{-4}$, the highest-accuracy (↑) uniform checkpoint at comparable average length (↓).

Task	Uniform Acc. / Avg. Len.	$CIB_{\beta^-,1.5B}$ Acc. / Avg. Len.	Δ Acc. (↑)	$CIB_{\beta^+,1.5B}$ Acc. / Avg. Len.	Δ Acc. (↑)
AIME24	36.9 / 6473	40.2 / 7298	+3.3	38.2 / 6937	+1.3
AIME25	24.8 / 7002	27.8 / 6866	+3.0	27.6 / 6757	+2.8
GPQA	31.8 / 6051	38.4 / 4571	+6.6	35.5 / 4429	+3.7
MATH500	84.2 / 2270	87.6 / 2359	+3.4	87.2 / 2017	+3.0
Olympiad	52.2 / 4322	55.5 / 4472	+3.3	55.3 / 4211	+3.1
Mean	46.0 / 5224	**49.9** / 5113	**+3.9**	48.8 / **4870**	+2.8

cautiously. Gains persist on the three larger benchmarks and are largest on GPQA, where $CIB_{\beta^-,1.5B}$ is 6.6 points more accurate and 24% shorter.

D.1.2 Paired trajectory analysis

We compare completions paired by benchmark and sample index over 1432 samples. Unlike the matched-length comparison, this analysis uses the highest-accuracy uniform checkpoint, $\beta_u = 1 \times 10^{-4}$. Its longer completions produce a wider token gap than in subsubsection D.1.1, making length-normalized marker density essential for distinguishing changes in composition from shortening alone.

We use four marker categories as proxies for redundant reasoning loops:

- **backtrack:** *Wait, Hold on, Actually, Hmm*;
- **reverify:** *Let me check, Let me verify, Let me double-check, Let me re-examine, Let me recompute*;
- **alternative:** *Alternatively, Another way, Another approach, Or maybe*;
- **self-correct:** *That's wrong, I made a mistake, Let me reconsider, I think I made*.*

For each completion, we count markers by category and in total, then compute paired differences relative to the uniform completion. Entries in the trajectory-analysis tables are rounded independently; differences and totals may therefore differ by one unit in the last digit from arithmetic on the displayed values. We report the results in Table 7.

Table 7: **DeepScaler-1.5B average markers count per completion (↓): uniform prior versus $CIB_{\beta^-,1.5B}$.**

Metric	n samples	Uniform	$CIB_{\beta^-,1.5B}$	Δ (↓)	$CIB_{\beta^+,1.5B}$	Δ (↓)
backtrack	1432	40.19	20.27	-19.92	25.92	-14.27
reverify	1432	1.21	0.77	-0.45	0.81	-0.41
alternative	1432	22.43	12.59	-9.84	14.70	-7.73
self-correct	1432	1.02	0.60	-0.42	0.76	-0.27
All markers	1432	64.85	**34.22**	**-30.63**	42.18	-22.67

Both semantic-prior variants reduce every marker category (Table 7). For $CIB_{\beta^-,1.5B}$, the total marker count falls by 47.2%. Marker density also falls, from 12.2 to 7.5 per 1000 tokens (Table 8). Thus, the reduction in markers exceeds what proportional shortening alone would explain.

D.2 DLER-1.5B

In this section, $CIB_{\beta^-,1.5B}$ denotes the DLER-1.5B semantic-prior model reported in the paper. We repeat the DeepScaler procedure with the same training and evaluation setup. DLER's shorter baseline chains of thought allow us to explore slightly higher values of β_u.

Table 8: **DeepScaler-1.5B marker reduction relative to length for $CIB_{\beta^-,1.5B}$.**

Metric	n samples	Uniform	$CIB_{\beta^-,1.5B}$	% change
n_markers	1432	64.8	**34.2**	-47.2
markers_per_1k_tokens	1432	12.2	**7.5**	-38.4

Table 9: **DLER-1.5B uniform-prior sweep.** Accuracy / average completion length on five reasoning benchmarks.

Setting	AIME24	AIME25	GPQA	MATH500	Olympiad	mean
Uniform prior, β_u sweep						
$\beta_u = 9 \times 10^{-5}$	36.5 / 3508	25.3 / 3386	38.9 / 3260	85.6 / 2093	54.0 / 2726	48.1 / 2995
$\beta_u = 3 \times 10^{-4}$	33.5 / 3431	24.6 / 3193	38.9 / 3189	86.2 / 1981	54.0 / 2645	47.4 / 2888
$\beta_u = 5 \times 10^{-4}$	33.8 / 3376	25.0 / 3087	37.9 / 3125	85.6 / 1853	52.6 / 2623	47.0 / 2813
$\beta_u = 8 \times 10^{-4}$	35.4 / 2772	23.0 / 2627	35.4 / 2890	84.8 / 1723	50.7 / 2232	45.9 / 2449

D.2.1 Matched-length comparison

The same selection rule yields the uniform checkpoint at $\beta_u = 5 \times 10^{-4}$ (Table 9), averaging 2813 tokens versus 2789 for the semantic-prior model: a difference of less than 1%. As with DeepScaler, no uniform checkpoint reached the 7B semantic-prior model's average length, precluding a matched-length comparison at that scale. We report the results in Table 10.

Table 10: **DLER-1.5B matched-length comparison with $CIB_{\beta^-,1.5B}$.** The uniform baseline uses $\beta_u = 5 \times 10^{-4}$.

Task	Uniform Acc. / Avg. Len.	$CIB_{\beta^-,1.5B}$ Acc. / Avg. Len.	Δ Acc. ($\uparrow$)
AIME24	33.8 / 3376	35.4 / 3198	+1.6
AIME25	25.0 / 3087	25.3 / 3260	+0.3
GPQA	37.9 / 3125	39.4 / 3183	+1.5
MATH500	85.6 / 1853	88.4 / 1687	+2.8
Olympiad	52.6 / 2623	54.3 / 2616	+1.7
Mean	47 / 2813	**48.6 / 2789**	**+1.6**

At average completion lengths matched to within 1%, the semantic prior improves accuracy on all five benchmarks, with a mean gain of 1.6 points (Table 10). As above, the gains on GPQA, MATH500, and Olympiad are less subject to the small-sample caveat for AIME24 and AIME25.

D.2.2 Paired trajectory analysis

We apply the same pairing procedure and marker categories to 1432 samples, again using the highest-accuracy uniform checkpoint for the trajectory analysis. Results are reported in Table 11–12.

Table 11: **DLER-1.5B mean marker counts per completion: uniform prior versus $CIB_{\beta^-,1.5B}$.**

Metric	n samples	Uniform	$CIB_{\beta^-,1.5B}$	Δ
backtrack	1432	21.19	7.40	-13.79
reverify	1432	0.23	0.20	-0.03
alternative	1432	2.91	1.75	-1.16
self-correct	1432	0.10	0.10	-0.00
All markers	1432	24.43	**9.45**	**-14.98**

Table 12: **DLER-1.5B marker reduction relative to length for $CIB_{\beta^-,1.5B}$.**

Metric	n samples	Uniform	$CIB_{\beta^-,1.5B}$	% change
n_markers	1432	24.43	**9.45**	-61.32%
markers_per_1k_tokens	1432	10.58	**5.29**	-49.99%

The mean marker count falls from 24.43 to 9.45, primarily through reduced backtracking. Marker density falls from 10.58 to 5.29 per 1000 tokens, a reduction of approximately 50%. As in DeepScaler, markers become less frequent relative to length, rather than merely fewer in absolute number.

D.3 Mechanistic Interpretation

Across both model families, the semantic prior improves accuracy on every benchmark at comparable average completion length. The paired analyses further show that the number of redundant-reasoning markers declines faster than length, supporting selective compression rather than proportional shortening. These observations are consistent with a content-dependent penalty that suppresses markers associated with redundant reasoning blocks while balancing their removal against accuracy.

Markers remain present, and we observed no reduction in the policy's exploration behavior (Figure 3 in Section 5.2 of the main paper). The β parameter in the CIB objective controls the trade-off between compression and accuracy.

E Training Details

To ensure reproducibility, we provide the full set of hyperparameters and infrastructure details used in our experiments. Our implementation relies on the `trl` library (version 0.26.2) for Group Relative Policy Optimization (GRPO) and `lighteval` (version 0.8.1) for robust evaluation.

E.1 Hyperparameters

We fine-tuned all models using the hyperparameters listed in Table 13.

Table 13: **GRPO Training Hyperparameters.** All experiments share these settings unless otherwise noted.

Hyperparameter	**Value**
Optimizer	AdamW
Learning Rate	1×10^{-6}
LR Scheduler	constant
Warmup Ratio	0.03
Batch Size (Global)	128
Number of Generations per Prompt	16
Temperature	0.8
Max Completion Length	4096 for DLER and 8192 for Deepscaler
Max Gradient Norm	1.0
KL Penalty Coefficient (β_{KL})	$5.e^{-4}$
CIB Regularization Weight (β^-, β^+)	$\{5.e^{-5}, 1.5e^{-4}\}$
Number of steps	150

E.2 Evaluation Setup

We utilized `lighteval` for all downstream benchmarks.

- **Inference Engine:** vLLM (version 0.10.2).
- **Sampling Strategy:** We used temperature $T = 0.6$, $\text{top}_p = 0.95$, 32K max completion length, and 16 generations per prompt.

- **Hardware:** Training was performed on a node with $8\times$ NVIDIA H100 (80GB) GPUs.

F Results from literature

We report additional results from [46] in Table 14.

Table 14: Compression results for DeepScaler-1.5B [46].

Model	MATH500		AIME24		AMC-23		OlympiadBench	
	Acc.↑	Tok.↓	Acc.↑	Tok.↓	Acc.↑	Tok.↓	Acc.↑	Tok.↓
DeepScaler-1.5B	85.8	3280	35.5	9246	74.2	6416	54.6	5974
ThinkPrune-I2k	85.5	1707	34.9	5095	74.3	2913	54.7	3498
ThinkPrune-4k	86.6	2042	35.5	6488	76.3	3839	55.7	4010
HAPO	84.4	2370	31.4	7702	70.3	4301	51.4	4571
AutoThink	84.9	1635	36.2	7201	67.8	3658	52.5	4085
Thinkless	81.3	2944	28.9	9143	65.7	5276	50.2	6057
LAPO-D	86.4	2365	37.6	5945	77.6	3655	56.1	4499
LAPO-I	86.3	2168	38.1	5371	78.3	3765	56.3	4024

G The list of datasets and models

Table 15: The list of datasets.

Dataset	**Link**	**License**
MATH500 [21]	`HuggingFace`	MIT
AIME 2024 [28]	`HuggingFace`	Copyright MAA
AIME 2025 [29]	`HuggingFace`	Copyright MAA
Minerva Math [17]	`HuggingFace`	MIT
OlympiadBench [9]	`HuggingFace`	MIT
AMC 2023 [27]	`HuggingFace`	Copyright MAA
GPQA-Diamond [32]	`HuggingFace`	CC BY 4.0
MMLU-Pro [43]	`HuggingFace`	MIT
LiveCodeBench [13]	`GitHub`	MIT
DeepScaleR dataset [24]	`HuggingFace`	MIT

Table 16: The list of models.

Model	**Link**	**License**
DLER-R1-1.5B [22]	`HuggingFace`	NSCLv1 (research only)
DLER-R1-7B [22]	`HuggingFace`	NSCLv1 (research only)
DeepScaleR-1.5B-Preview [24]	`HuggingFace`	MIT
Qwen2.5-Base-0.5B [50]	`HuggingFace`	Apache 2.0
Qwen2.5-Base-1.5B [50]	`HuggingFace`	Apache 2.0
Qwen2.5-Base-3B [50]	`HuggingFace`	Qwen Research
Qwen2.5-Base-7B [50]	`HuggingFace`	Apache 2.0
L3L1-1.5B-Exact [1]	`HuggingFace`	MIT
OpenReasoning-Nemotron-1.5B [2, 30]	`HuggingFace`	CC-BY-4.0

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: To the best of our knowledge the claims made in the introduction accurately reflect the scope and the contributions of the paper as well as important assumptions.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations in the work are discussed in the limitations subsection.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We provide the full set of assumptions, derivations, and proofs in section 3 and Appendix A.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the details needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide all the information needed to reproduce the main experimental results in the section 5, Appendix B, and Appendix E.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Our organization requires legal review of released source code which is not feasible at the submission time due to the time constraints of the conference deadline. However, we will release the code upon acceptance of the paper.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: We provided detailed information on all the training and test details in section 5 and Appendix E.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We provide mean and the standard error of the mean (SEM) for the experimental results.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide the information on the computer resources in Appendix E.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The paper conforms to the NeurIPS Code of Ethics

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: To the best of our knowledge there is no societal impact of the work performed.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [No]

    Justification: The paper poses no additional risks of misuse in comparison to the open source models used in the experiments.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes].

    Justification: The list of assets used in the paper along with URLs and license is given in Appendix G.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: Our paper does not release new assets

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.