REASONING AS COMPRESSION: UNIFYING BUDGET FORCING VIA THE CONDITIONAL INFORMATION BOTTLENECK

Fabio Valerio Massoli, Andrey Kuzmin, 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 reducing cost via fine-tuning with heuristic length penalties, suppress 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 (RL) 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-based approaches, we introduce a semantic prior that measures token cost by surprisal under a language model prior. Empirically, our CIB objective prunes cognitive bloat while preserving fluency and logic, improving accuracy at moderate compression and enabling aggressive compression with minimal accuracy drop.

1 INTRODUCTION

Chain-of-Thought (CoT) prompting (Wei et al., 2022) 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 "flat tax" 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*. We propose a unified framework based on the Information Bottleneck (IB) principle (Tishby et al., 1999), positing that an ideal reasoning chain is the minimal sufficient statistic of the prompt required to predict the answer. We identify that standard IB (Tishby et al., 1999) cannot be naively applied to transformers due to a theoretical inconsistency we term the "Attention Paradox": 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 paradox by modeling CoT generation under the Conditional Information Bottleneck (CIB) as *Source Coding with Side Information*. As a result, a novel RL objective naturally arises from the CIB framework. Instead of a uniform length

*Qualcomm AI Research is an initiative of Qualcomm Technologies, Inc.

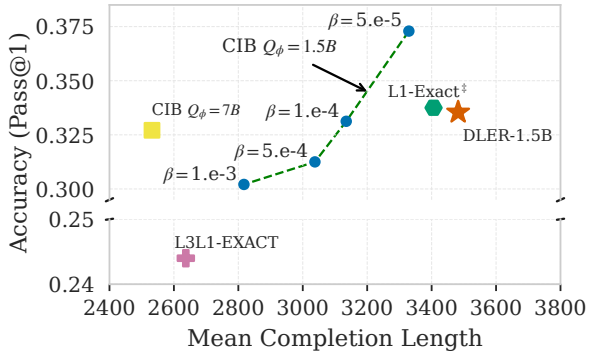

Figure 1: **Pareto frontier for AIME24.** The β weight from CIB objective confers fine-grained control over the accuracy-compression trade-off. A stronger prior ($Q_\phi = 7B$, yellow square) allows for stronger compression compared to a smaller one ($Q_\phi = 1.5B$, blue circles). As a reference, we report the baseline model (DLER (Shih-Yang Liu et al., 2025), red star), the L3L1-EXACT (Aggarwal & Welleck, 2025) model snapshot (purple cross), and our implementation of L1-Exact length penalty from the same paper (green hexagon).

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 (see Figure 1).
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 Paradox" via the Conditional Information Bottleneck, 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 structured as outlined below. Section 2 describes the prior works and their differences to our method. Section 3 explains the "Attention Paradox", presents CIB mathematically, introduces the semantic prior, and relation of CIB to existing budget forcing methods. Section 5 contains experimental results and ablations.

2 Related Work

2.1 Budget Forcing and Efficient Reasoning

Recent studies suggest that optimal reasoning compute should scale with problem complexity Zhang et al. (2025), yet unconstrained models often exhibit excessive verbosity even on simple tasks Muennighoff et al. (2025). This has motivated "Budget Forcing" strategies spanning training and inference, including reward shaping with length costs Aggarwal & Welleck (2025), and hard truncation Shih-Yang Liu et al. (2025). More granular approaches include difficulty-aware allocation Cheng et al. (2025) and reference-guided budgeting Wu et al. (2025); Li et al. (2025b); Luo et al. (2025a), sometimes tracking history Huang et al. (2025a) or decomposing costs per-token Jiang et al. (2025). Inference-only methods steer generation via auxiliary predictors Li et al. (2025a); Han et al. (2025) or employ early-exit decoding Mao et al. (2025); Wang et al. (2025b). Alternative paradigms replace verbose CoT with concise drafting Xu et al. (2025); Renze & Guven (2024), selective reasoning policies Wang et al. (2025a), or trace compression via token pruning and skipping Xia et al. (2025); Choi et al. (2025); Cui et al. (2025); Cheng & Van Durme (2024). Wang et al. (2024a) further propose budget-aware evaluation metrics. 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.

2.2 Information Theory in Large Language Models

The IB principle Tishby et al. (1999) was proposed as a framework for analyzing deep learning Shwartz-Ziv & Tishby (2017), followed by various discussions Saxe et al. (2018), applications in reasoning and robustness Huang et al. (2025b), and hallucination detection Wang et al. (2024b). 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 Conditional Information Bottleneck framework.

3 Methodology

In this section, we formalize efficient reasoning as an optimization problem within the CIB framework. First, we expand on the concept of "Attention Paradox" and briefly introduce the CIB approach. Subsequently, we define the theoretical objective and the probability space. We then rigorously derive computable variational bounds for both the sufficiency and minimality terms, resolving the intractability of the true distributions. Finally, we present our rewards for training LLMs. In what follows, we refer to X, Z, and Y, as the prompt, the CoT, and the ground truth answer, respectively. We refer the reader to Appendix A for details.

The Attention Paradox The standard Information Bottleneck (IB) principle (Tishby et al., 1999) 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 Large Language Model (LLM)s. 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 Paradox**. 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 paradox, 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 the LLM policy $\pi_\theta(\cdot|\cdot)$ to re-parameterize the above optimization problem.

3.1 Problem Formulation

We consider a reasoning task defined by a dataset distribution $P_{\mathcal{D}}(X, Y)$, where X represents a problem prompt and Y represents 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.

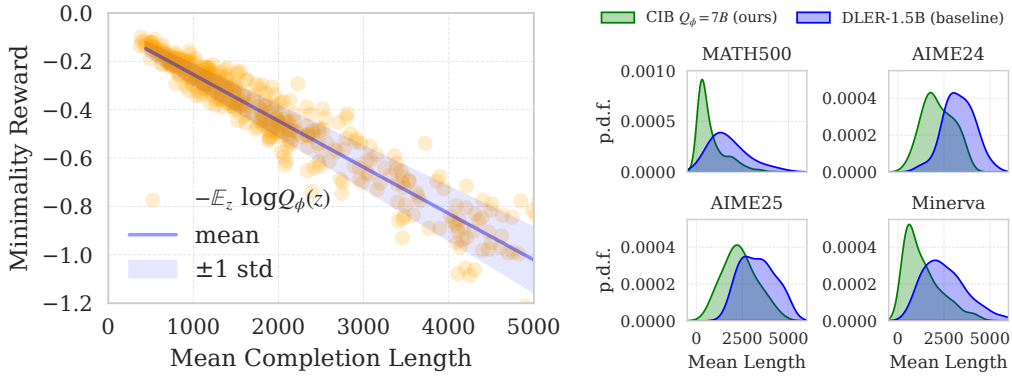

Figure 2: (a) **Minimality reward as a function of the completion length.** We observe a consistent negative correlation between the completion length and the minimality reward used during RL training. The shadow blue region shows the $\pm 1\sigma$ band representing the spread of the information cost for the token chosen within CoTs with similar length. (b) **Lengths Distribution.** Compared 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 (within $\lesssim 1.4\%$ – see Table 1)

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:

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

where $\mu \geq 0$ controls the rate-distortion trade-off. To derive our final reward function, we rewrite Equation 3 as a maximization problem, rather than a minimization one. Therefore, our objective becomes $\max_\theta \mathcal{L}_{\text{CIB}}(\theta) = \max_\theta I(Z;Y \mid X) - \beta I(X;Z)$, where β gives direct control on the trade-off between accuracy and compression level (see Figure 1). See Appendix A for the detailed discussion on the derivation. In what follows, we discuss how we can optimize the above bound.

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

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

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 for correct answers and 0 for the wrong ones. As we show, a finite, stable surrogate for the log-verifier objective is the following accuracy reward

$$r_{\text{acc}}(x,y,z) := \mathbb{1}(\widehat{y}(x,z) = y)\,, \tag{4}$$

3.3 Deriving the Minimality Term (Information Cost)

We aim to minimize the MI $I(X;Z)$ to penalize redundancy in the CoT. However, computing $P(Z)$ is not tractable. As we show in appendix A, the following variational upper bound can be used:

$$I(X;Z) = \mathbb{E}_{X,Z}\left[\log \frac{\pi_\theta(Z \mid X)}{P(Z)}\right] \le \mathbb{E}_{X,Z}\left[-\log Q_\phi(Z)\right] - H(Z \mid X)] \tag{5}$$

where $Z \sim \pi_\theta(\cdot|Z)$. To effectively penalize information specific to X (redundancy), $Q_\phi(Z)$ must be an **unconditional prior** that does not observe the prompt 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.

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.

3.4 Reward Modeling

Combining the bounds, we aim to maximize the following objective:

$$\mathcal{L}_{\text{CIB}} = \mathbb{E}_{(X,Y)\sim P_{\mathcal{D}}, Z\sim\pi_\theta}\Big[\log \widetilde{Q}_\rho(Y|X,Z) + \beta \sum_{t=1}^{T} \log Q_\phi(z_t \mid z_{<t})\Big], \tag{6}$$

where the first term represents the accuracy score from the verifier, $\widetilde{Q}_\rho$, as previously stated, while Q_ϕ is chosen as prior distribution. This objective effectively assigns a "value-added tax" to every token. The cost $-\log Q_\phi$ penalizes tokens that are high surprisal to the blind prior or verbose, while the accuracy term justifies the cost for tokens that resolve the answer. Thus, we can define our reward model as:

$$R(X,Y,Z) \;:=\; r_{\text{acc}}(X,Y,Z) + \beta r_{\text{min}}(X,Z), \tag{7}$$

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. In this formulation, accuracy remains the primary objective, while r_{min} acts as a semantic regularizer controlled by the coefficient β. This effectively assigns a "value-added tax" 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. We maximize the expected reward using Group Relative Policy Optimization (GRPO) (Shao et al., 2024).

Ultimately, although our framework can be instantiated as a particular class of reward models for RL-based training, its central contribution is a general and highly flexible recipe for optimizing reasoning efficiency. By varying the implementations of the verifier and the prior models, practitioners can explore a broad design space and tailor these components to the requirements of specific downstream tasks and deployment constraints.

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.

4.1 Recovering Length Penalties

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:

$$-\log Q(Z) = -\sum_{t=1}^{T} \log(\frac{1}{|V|}) = T \cdot \log|V| \tag{8}$$

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. □

Proposition 4.2. Target-length penalties, such as LCPO-Exact (Aggarwal & Welleck, 2025), 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:

$$Q_{\text{LCPO}}(Z) \propto e^{-|n_{gold} - n_y|} \tag{9}$$

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

5.1 Training

We conduct extensive experiments to demonstrate the benefit of our method on compressing CoT in state-of-the-art (SOTA) reasoning models. We consider two model families: DLER-{1.5B, 7B} (Shih-Yang Liu et al., 2025) and Deepscaler-1.5B (Luo et al., 2025c). We apply our CIB objective to penalize verbose completions using GRPO with a group size of 16 and group-scaled rewards. To maximize training stability, we filter the DeepScaleR dataset (Luo et al., 2025b) to remove prompts with zero group reward standard deviation. For the prior, we use a Qwen2.5-Base-{1.5B, 7B} model. Note that the prior is used at training time only, thus without imposing any additional cost at inference time.

5.2 Evaluation

We evaluate our models and baselines on five math reasoning benchmarks: Math500 (Lightman et al., 2023), AIME24 (Mathematical Association of America, 2024), AIME25 (Mathematical Association of America, 2025), Minerva (Lewkowycz et al., 2022), and OlympiadBench (He et al., 2024). Following the protocol in Shih-Yang Liu et al. (2025), we use vLLM as the inference engine (temperature 0.6, $\text{top}_p = 0.95$, max tokens 32K, 16 generations/prompt) and report pass@1 accuracy. Further training and evaluation details are provided in Appendix E.

5.3 Model Choice

We focus on two families of models. To the best of our knowledge, Deepscaler-1.5B (Luo et al., 2025c) and DLER-{1.5B,7B} (Shih-Yang Liu et al., 2025) represent SOTA concerning small language models. Specifically, Deepscaler achieves higher average performance compared to DLER-1.5B

Table 1: **Performance Results.** Accuracy and average completion length across five different benchmarks. For each benchmark we highlight in bold the best performance (within a max drop in average accuracy of 1.5%). The last columns reports average values for accuracy and completion length across all the benchmarks. Token reduction is highlighted in green. Each reduction is computed with respect to the proper baseline. Concerning the Deepscaler-1.5B, DLER-{1.5B, 7B}, and L3L1-{1.5B, 7B}-{EXACT, MAX} baselines, we used the models publicly available on huggingface. The symbols β^- and β^+ represent two choices for the β parameter in the CIB objective corresponding to $5.e^{-5}$ and $1.5e^{-4}$, respectively.

Model	MATH500 Acc.↑	MATH500 Tok.↓	AIME24 Acc.↑	AIME24 Tok.↓	AIME25 Acc.↑	AIME25 Tok.↓	Minerva Acc.↑	Minerva Tok.↓	Olympiad Acc.↑	Olympiad Tok.↓	Average Acc.↑	Average Tok.↓
Deepscaler-1.5B	85.8	3190	38.1	9269	25.2	8901	19.9	6375	54.7	5875	44.7	6722
CIB$^{Q_\phi=1.5B}_{\beta^-}$	87.6	2359	39.2	7156	**27.3**	6725	20.1	4512	**55.4**	4473	45.9	5045
CIB$^{Q_\phi=1.5B}_{\beta^+}$	**87.8**	2106	**40.2**	6857	26.3	6672	**20.3**	4144	55.3	4208	**46.0**	4797
CIB$^{Q_\phi=7B}$	84.0	**1548**	35.2	**5720**	25.8	**5362**	19.6	3097	55.2	**4029**	44.0	**3951**
L3L1-1.5B$_{\text{EXACT}}$	85.0	1932	24.4	2637	20.2	2604	19.8	2124	49.0	2368	39.7	2333
L3L1-1.5B$_{\text{MAX}}$	84.2	1610	23.1	2657	20.4	2579	19.9	**1999**	50.9	2078	39.7	2185
DLER-1.5B	86.4	1873	33.5	3482	23.8	3284	20.9	2585	53.9	2741	43.7	2793
L1-Exact‡	86.6	1751	33.8	3406	24.4	3086	21.2	2276	53.9	2576	44.0	2619
CIB$^{Q_\phi=1.5B}_{\beta^-}$	86.8	1724	**37.3**	3329	**25.8**	3136	**21.6**	2345	**54.2**	2616	**45.1**	2630
CIB$^{Q_\phi=1.5B}_{\beta^+}$	**88.2**	1621	34.2	3334	25.2	3123	21.0	2304	54.0	2543	44.5	2585
CIB$^{Q_\phi=7B}$	85.0	**805**	32.7	**2532**	22.9	**2326**	20.4	**1326**	51.6	**1630**	42.5	**1724**
DLER-7B	93.6	1159	48.5	3261	36.9	3264	**27.2**	1872	**65.0**	2851	54.2	2481
L1-Exact‡	92.0	726	46.5	2530	30.8	2621	26.2	1148	62.0	1833	51.5	1772
CIB$^{Q_\phi=7B}_{\beta^-}$	**94.0**	1033	**49.4**	3003	**37.1**	3113	**26.7**	1643	**64.3**	2632	**54.3**	2285
CIB$^{Q_\phi=7B}_{\beta^+}$	92.2	**678**	48.3	**2617**	35.6	**2580**	25.8	**849**	62.6	**1711**	52.9	**1687**

* The higher reduction in the average number of reasoning tokens comes at the cost of a very significant degradation in accuracy.
‡ Our implementation of the L1-Exact reward function (Aggarwal & Welleck, 2025).

while being more verbose. Moreover, Deepscaler represents the base model for L3L1, or LCPO, models Aggarwal & Welleck (2025), thus offering a fair comparison against our approach. Given that, DLER already reported Pareto dominance compared to other "Budget Forcing" methods (Shih-Yang Liu et al., 2025), we report a comparison with all other methods in Appendix F.

5.4 CoT Compression

Before training, we verify that the proposed minimality reward provides a usable learning signal. As shown in Figure 2(a), the minimality reward defined in section 3 exhibits a pronounced negative correlation with completion length, indicating that longer generations incur systematically higher cost. We also observe a limited dispersion around the mean at a given length. Such a dispersion indicates that the reward is not merely a function of length, but also depends on the specific token sequence.

We successfully compress CoT across all benchmarks. Figure 2(b) illustrates the significant shift toward shorter, denser reasoning chains for the CIB-tuned DLER-1.5B model compared to the baseline. As detailed in Table 1, the CIB objective enables precise control over the accuracy–efficiency trade-off via the regularization coefficient β. We identify two distinct operating regimes: *conservative compression* (β^-), which yields moderate token reduction with negligible accuracy loss, and *aggressive compression* (β^+), which achieves high reduction (up to $\approx 41\%$) with a maximal average performance drop of $\lesssim 1.5\%$. This tunability allows users to traverse the Pareto frontier (see Figure 1) and customize model behavior for specific downstream constraints, such as memory- or latency-constrained edge devices. We further observe that the capacity of the reference prior Q_ϕ

plays a critical role in optimization. Using a larger prior (7B) yields superior compression at similar accuracy compared to a smaller prior (1.5B), as the stronger model provides a sharper estimate of semantic redundancy (surprisal). However, we note a slight average accuracy degradation (up to 1.4%) when scaling the prior without re-tuning. We emphasize that this gap could likely be closed by specific hyperparameter optimization for the 7B prior; due to resource limitations, our experiments utilized the hyperparameters optimized for the 1.5B prior. Additional ablation results are provided in Appendix D.

5.5 Comparison to prior work

We provide a comprehensive set of results in Table 1. We fine-tuned DLER and DeepScaleR models using CIB with two distinct regularization coefficients: β^-, which targets moderate compression while strictly preserving accuracy, and β^+, which prioritizes higher compression factors. We also ablate the influence of the prior model size using Qwen-2.5-Base (1.5B and 7B). To benchmark performance against state-of-the-art budget forcing, we compare against two distinct baselines. First, we evaluate publicly available L1-compressed models (Aggarwal & Welleck, 2025) initialized from DeepScaleR-Preview (rows "L3L1-1.5B-{EXACT,MAX}"), aligning all inference settings (temperature, generations, context length) to match our protocol. Second, to control for base model differences in the 7B regime, we implemented an L1-based length penalty baseline (row "L1‡") applied directly to DLER-7B under identical training budgets and starting checkpoints. The results demonstrate that CIB achieves Pareto optimal performance compared to length-based methods. A critical distinction emerges when comparing our approach to the L3L1 baselines on DeepScaleR-1.5B—same starting checkpoint as our CIB models. While L3L1 models achieve higher raw compression rates, this efficiency comes at a steep cost to reliability: they exhibit an average performance drop of 5% relative to the base model, with degradations up to 15% on AIME24. In contrast, our CIB approach demonstrates significantly greater stability, limiting the average accuracy loss to at most 0.7% (max 2.9% on AIME24). This validates that the semantic objective selectively preserves high-utility reasoning, avoiding the brittle failure modes of naive length penalties. This advantage is further confirmed by the results on the 7B models. While L3L1 baselines continue to trade accuracy for length, our CIB models surpass them in *both* dimensions. Crucially, when compared against the controlled "L1-Exact‡" baseline on DLER-7B, CIB achieves the optimal trade-off: it reaches a compression factor of up to 32% while maintaining higher average accuracy than the L1-penalized equivalent. This confirms that the efficiency gains of our semantic objective scale effectively to larger models, systematically outperforming standard length penalties even when the base architecture and training budget are held constant.

6 Conclusions

In this work, we address the challenge of efficient reasoning in LLMs by reframing "Budget Forcing" from an information-theoretic perspective. We identified the "Attention Paradox"—a structural inconsistency in applying standard Information Bottleneck principles to transformer architectures—and proposed a Conditional Information Bottleneck framework to resolve it. Our empirical results on mathematical reasoning benchmarks demonstrate that penalizing tokens based on their semantic information content yields a more favorable trade-off between CoT length and accuracy than naive length-based penalties. By tuning the regularization coefficient β, we demonstrate that it is possible to traverse the Pareto frontier, achieving significant reductions in token budget (up to 41%) with minimal degradation in reasoning performance ($\lesssim 1.5\%$). Furthermore, our analysis indicates that the quality of the reference prior matters: stronger priors provide better estimates of redundancy, allowing for more aggressive compression with minimal loss in performance. These findings suggest that efficient inference requires moving beyond a "flat tax" on token count toward metrics that value computation based on its utility. While our method introduces a dependency on a reference model during training, it offers a principled path toward deploying capable reasoning models in resource-constrained environments.

Ultimately, although our framework can be instantiated as a particular class of reward models for RL-based training, its central contribution is a general and flexible recipe for optimizing reasoning efficiency. By varying the implementations of the verifier and the prior, practitioners can explore a broad design space and tailor these components to the requirements of specific downstream tasks and deployment constraints.

REFERENCES

Pranjal Aggarwal and Sean 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.

Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. In *International Conference on Learning Representations*, 6 February 2017.

Jeffrey Cheng and Benjamin Van Durme. Compressed chain of thought: Efficient reasoning through dense representations. *arXiv preprint arXiv:2412.13171*, 2024. URL https://arxiv.org/pdf/2412.13171.

Zhengxiang Cheng, Dongping Chen, Mingyang Fu, and Tianyi Zhou. Optimizing length compression in large reasoning models. *arXiv preprint arXiv:2506.14755*, 2025.

Sunguk Choi, Yonghoon Kwon, and Heondeuk 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.pdf.

Yingqian Cui, Pengfei He, Jingying Zeng, Hui Liu, Xianfeng Tang, Zhenwei Dai, Yan Han, Chen Luo, Jing Huang, Zhen Li, Suhang Wang, Yue Xing, Jiliang Tang, and Qi He. Stepwise perplexity-guided refinement for efficient chain-of-thought reasoning in large language models. *arXiv preprint arXiv:2502.13260*, 2025. URL https://aclanthology.org/2025.findings-acl.956.pdf.

Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyu Zhao, Shiqing Ma, and Zhenyu 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.pdf.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Zhou, Lei Hou, Juanzi Li, and Maosong 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)*, pp. 14138–14166, 2024. URL https://aclanthology.org/2024.acl-long.762.

Chengyu Huang, Zhengxin Zhang, and Claire Cardie. Hapo: Training language models to reason concisely via history-aware policy optimization. *arXiv preprint arXiv:2505.11225*, 2025a.

Y. Huang et al. Revisiting llm reasoning via information bottleneck. *arXiv preprint arXiv:2507.18391*, 2025b.

Shuyang Jiang, Yusheng Liao, Ya Zhang, Yanfeng Wang, and Yu Wang. Overthinking reduction with decoupled rewards and curriculum data scheduling. *arXiv preprint arXiv:2509.25827*, 2025.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 3843–3857, 2022. URL https://arxiv.org/abs/2206.14858.

Cheuk Ting Li. Channel simulation: Theory and applications to lossy compression and differential privacy. *Found. Trends® Commun. Inf. Theory*, 21(6):847–1106, 17 December 2024.

Junyan Li, Chuang Gan, et al. Steering llm thinking with budget guidance. *arXiv preprint arXiv:2506.13752*, 2025a. NVIDIA & UMass Amherst.

Zheng Li, Qingxiu Dong, Jingyuan Ma, Di Zhang, Kai Jia, and Zhifang Sui. Selfbudgeter: Adaptive token allocation for efficient llm reasoning. *arXiv preprint arXiv:2505.11274*, 2025b.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023. doi: 10.48550/arXiv.2305.20050. URL https://arxiv.org/abs/2305.20050.

Haotian Luo, Li Shen, Haiying He, Yibo Wang, Shiwei Liu, Wei Li, Naiqiang Tan, Xiaochun Cao, and Dacheng Tao. O1-pruner: Length-harmonizing fine-tuning for o1-like reasoning pruning. *arXiv preprint arXiv:2501.12570*, 2025a.

Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Tianjun Zhang, Erran Li, Raluca Ada Popa, and Ion Stoica. Deepscaler: Effective rl scaling of reasoning models via iterative context lengthening, 2025b. URL https://arxiv.org/abs/2509.25176.

Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Tianjun Zhang, Erran Li, Raluca Ada Popa, and Ion Stoica. Deepscaler: Surpassing o1-preview with a 1.5b model by scaling rl. https://huggingface.co/agentica-org/DeepScaleR-1.5B-Preview, 2025c.

Minjia Mao, Bowen Yin, Yu Zhu, and Xiao Fang. Early stopping chain-of-thoughts in large language models. *arXiv preprint arXiv:2509.14004*, 2025. URL https://arxiv.org/pdf/2509.14004.

Mathematical Association of America. American invitational mathematics examination (aime) 2024, 2024. URL https://maa.org/. Problems I and II.

Mathematical Association of America. American invitational mathematics examination (aime) 2025, 2025. URL https://maa.org/. Problems I and II.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 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/.

Matthew Renze and Erhan 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/2401.05618v1.pdf.

Andrew Michael Saxe, Yamini Bansal, Joel Dapello, Madhu Advani, Artemy Kolchinsky, Brendan Daniel Tracey, and David Daniel Cox. On the information bottleneck theory of deep learning. In *International Conference on Learning Representations*, 2018.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y.K. Li, Y. Wu, and Daya 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/abs/2402.03300.

Ximing Lu Shih-Yang Liu, Xin Dong et al. Dler: Doing length penalty right - incentivizing more intelligence per token via reinforcement learning. *arXiv preprint arXiv:2502.xxxxx*, 2025. doi: 10.48550/arXiv.2502.xxxxx. URL https://arxiv.org/abs/2502.xxxxx.

Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *arXiv:1703.00810 [cs]*, 2 March 2017.

Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. In *Allerton Conference*, 1999.

Jiaqi Wang, Kevin Qinghong Lin, James Cheng, and Mike Zheng Shou. Think or not? selective reasoning via reinforcement learning for vision-language models. *arXiv preprint arXiv:2505.16854*, 2025a. URL https://arxiv.org/pdf/2505.16854.

Junlin Wang, Siddhartha Jain, Ben Athiwaratkun, Dejiao Zhang, Baishakhi Ray, and Varun Kumar. Reasoning in token economies: Budget-aware evaluation of LLM reasoning strategies. In *Proceedings of EMNLP 2024*, 2024a. URL https://aclanthology.org/2024.emnlp-main.1112.pdf.

X. Wang et al. Understanding chain-of-thought in llms through information theory. *arXiv preprint arXiv:2411.11984*, 2024b.

Xi Wang, James McInerney, Lequn Wang, and Nathan Kallus. Entropy after </think> for reasoning model early exiting. *arXiv preprint arXiv:2509.26522*, 2025b. URL https://arxiv.org/pdf/2509.26522.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022.

Xingyu Wu, Yuchen Yan, Shangke Lyu, Linjuan Wu, Yiwen Qiu, Yongliang Shen, Weiming Lu, Jian Shao, Jun Xiao, and Yueting Zhuang. Lapo: Internalizing reasoning efficiency via length-adaptive policy optimization. *arXiv preprint arXiv:2507.15758*, 2025.

Aaron D Wyner and Jacob Ziv. The rate-distortion function for source coding with side information at the decoder. *IEEE Transactions on Information Theory*, 1976.

Heming Xia, Chak Tou Leong, Wenjie Wang, Yongqi Li, and Wenjie Li. Tokenskip: Controllable chain-of-thought compression in LLMs. *arXiv preprint arXiv:2502.12067*, 2025. URL https://arxiv.org/pdf/2502.12067.

Silei Xu, Wenhao Xie, Lingxiao Zhao, and Pengcheng He. Chain of draft: Thinking faster by writing less. *arXiv preprint arXiv:2502.18600*, 2025. URL https://arxiv.org/pdf/2502.18600.

Junyu Zhang, Yifan Sun, Tianang Leng, Jingyan Shen, Liu Ziyin, Paul Pu Liang, and Huan 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 is defined as:
$H(X) := \mathbb{E}_{X\sim P_X}(-\log P(X)), H(Y|X) := \mathbb{E}_{(X,Y)\sim P_{XY}}(-\log P(Y|X)), I(X;Y) := H(X) - H(X|Y).$

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 Li (2024).

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:

$$\max_{P(Z|X,Y)} \sum_{x,y,z} P(x, y)P(z|x, y) \log P(y|x, z) - \beta \sum_{x,z} P(x)P(z|x) \log \frac{P(z|x)}{P(z)}$$
$$\text{s.t.} \sum_z P(z|x, y) = 1, \forall x, y. \tag{10}$$

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 Tishby et al. (1999). 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 Wyner & Ziv (1976) 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 Alemi et al. (2017). 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:

$$\begin{aligned} \max_{\theta} \quad & \mathcal{L}_{\text{CIB}}(\theta) = I(Y;Z|X) - \beta I(X;Z) \\ \text{s.t.} \quad & P(y|x) = \sum_z \pi_\theta(y|x,z)\pi_\theta(z|x), \quad \forall x, y. \end{aligned} \tag{11}$$

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.

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:

$$\sum_{i=1}^{m} \mathbb{E}_{Z\sim\pi_\theta(Z|x_i)}[\log \pi_\theta(y_i|x_i,Z)]. \tag{12}$$

We can also approximate the bound using variational approximation of Alemi et al. (2017). 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:

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

Minimality term. For the minimality term, we use a variational approximation to find a variational bound similar to Alemi et al. (2017). 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:

$$\pi_\theta(y_{\text{true}}(x)|x,z) = 1, \tag{13}$$

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 FROM LOG-VERIFIER TO A 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

$$\widetilde{Q}_\rho(y \mid x, z) := \varepsilon + (1-\varepsilon)\,\mathbb{1}(\widehat{y}(x,z) = y)\,, \tag{14}$$

where $\varepsilon \in (0, 1)$ and $\widehat{y}$ is the predicted answer. Then

$$\log \widetilde{Q}_\rho(y \mid x, z) = \log \varepsilon - \log \varepsilon \, \mathbb{1}(\widehat{y}(x, z) = y) \, . \tag{15}$$

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

$$r_{\mathrm{acc}}(x, y, z) := \mathbb{1}(\widehat{y}(x, z) = y) \, , \tag{16}$$

which is a finite, stable surrogate for the log-verifier objective.

C CoT Qualitative Comparison: Pruning Cognitive Bloat

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 3–5). 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 3), 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 "cognitive bloat," utilizing conversational scaffolding and unstructured exploration. For instance, in arithmetic search tasks (Figure 4), the baseline explicitly calculates invalid candidates (e.g., 98^3) before testing the correct one. Similarly, in constraint satisfaction problems (Figure 5), 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 3, 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.

C.1 Efficiency Gain

To quantify the trade-off between reasoning performance and computational cost, we define two metrics, namely, *Compression Factor* (C_f): $C_f = 1 - \ell_{\text{CIB}}/\ell_{\text{base}}$, measuring the relative reduction in completion length between the base model, ℓ_{base}, and our models, ℓ_{CIB}, and *Accuracy Ratio* (A_r): $A_r = \mathcal{A}_{\text{CIB}}/\mathcal{A}_{\text{base}}$, normalizing performance against the baseline. Figure 6 visualizes these metrics across architectures and benchmarks. The upper half-plane ("Golden Zone") represents the ideal outcome: models that are strictly superior to the baseline in both speed ($C_f > 0$) and accuracy ($A_r \geq 1$). Models in the bottom-right offer significant speedups for specific low-latency applications where partial accuracy degradation is permissible. By filtering for the Golden Zone, we select models that are "smarter" and faster, rather than those that simply truncate reasoning. When looking at Figure 6, one must keep in mind that gains for each model are normalized to its own baseline.

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 3: **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.

C.2 Qualitative CoT Comparison

To validate the assumption that our objective targets "cognitive bloat" rather than essential logic, we analyzed reasoning traces across arithmetic and symbolic tasks. We observe that CIB systematically eliminates conversational scaffolding, redundant verification loops, and tautological checks. Unlike naive truncation, the semantic prior fundamentally alters the reasoning topology, preserving the "computational bridge" while filtering transitions that offer low marginal information regarding Y. Detailed case studies are provided in Appendix C (Figure 3–5).

C.3 Information Density Analysis

To quantify the mechanics of compression, we analyze the *information density* of the reasoning traces. We define information density as the token-wise surprisal, $-\log p(z_t|z_{<t}, x)$, measured relative to a frozen reference model. In standard CoT, this density is typically low and heterogeneous: critical logical operations (high surprisal) are diluted by extensive linguistic scaffolding and repetitive self-correction (low surprisal). Figure 7 illustrates this phenomenon. The baseline profile (dashed gray) is characterized by "valleys" with low surprisal (≈ 0.1 nats), indicating sequences that are highly predictable and thus carry negligible unique information regarding the target. In contrast, the CIB profile (solid blue and green) exhibits a higher floor ($\gtrsim 0.2$ nats). By penalizing cumulative low-utility transitions, the objective functions as a high-pass semantic filter, excising the predictable filler while preserving the high-entropy peaks. This confirms that CIB achieves compression not by random truncation, but by maximizing the average information rate of the channel, effectively distilling the reasoning trace down to its essential computational bridge.

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 4: **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.

D Additional Results

Figure 8 - 11 show the Inference Time Compute (ITC) behavior of our CIB models compared to the baselines on two math reasoning benchmarks, namely, AIME24 and AIME15. Notably, CIB-compressed models exhibit on par or better scaling behavior than baselines. Especially, when bounding the maximum generation length to 2K, Figure 10, or 3K, Figure 10, we see that CIB trained model with the large prior ($Q_\phi = 7B$) achieves superior scaling performance.

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 2.

E.2 Evaluation Setup

We utilized `lighteval` for all downstream benchmarks.

- **Inference Engine:** vLLM (version 0.10.2).

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 5: **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.

Table 2: **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 8196 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

- **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.

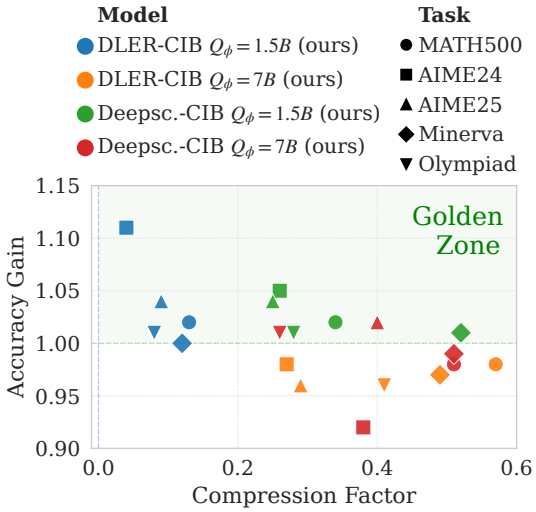

Figure 6: **Meta-Generalization: Robustness Across Benchmarks.** Efficiency gain of CIB across diverse benchmarks and models. Points falling in the upper half-plane ("Golden Zone") exhibit strictly superior efficiency, achieving higher information density with reduced computational cost.

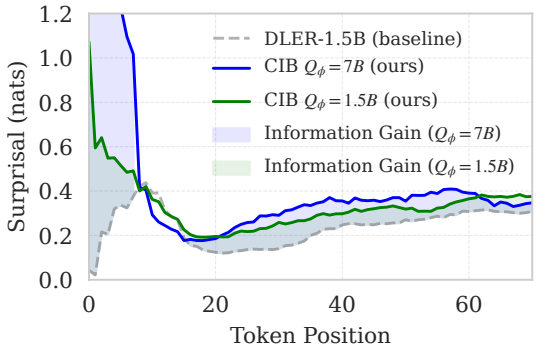

Figure 7: **Information Density Profile**. Token-wise surprisal evaluated against the baseline language prior. A lower value of the surprisal corresponds to predictable linguistic filler and cognitive bloat. CIB models maintain a consistently higher information floor ($\gtrsim$ 0.2 nats) confirming that the compression is semantic rather than arbitrary.

F RESULTS FROM LITERATURE

We report additional results from Wu et al. (2025) in Table 3.

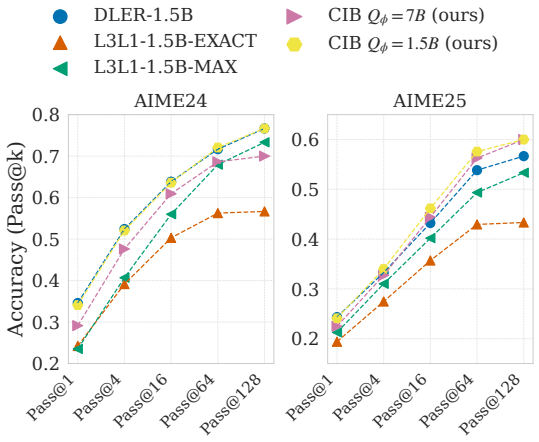

Figure 8: **Inference Time Compute.** Pass@k accuracy for different values of k, with a maximum completion length of 8K tokens.

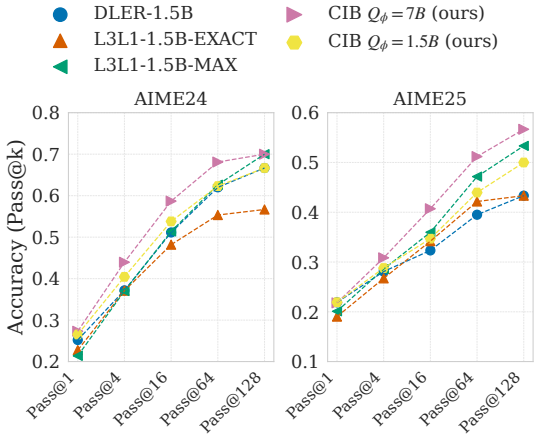

Figure 9: **Inference Time Compute.** Pass@k accuracy for different values of k, with a maximum completion length of 3K tokens.

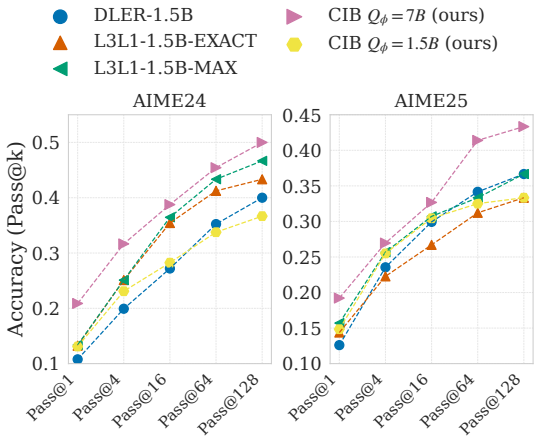

Figure 10: **Inference Time Compute.** Pass@k accuracy for different values of k, with a maximum completion length of 2K tokens.

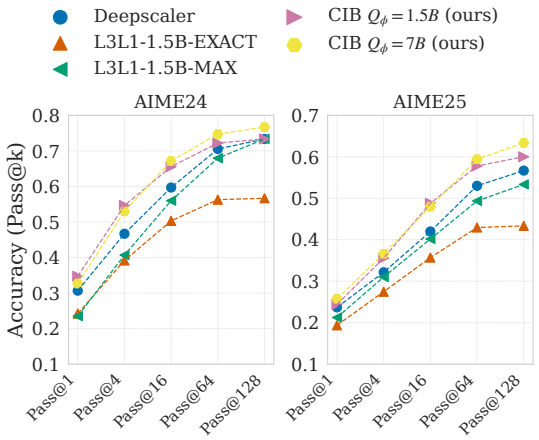

Figure 11: **Inference Time Compute.** Pass@k accuracy for different values of k, with a maximum completion length of 8K tokens.

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

Table 3: Compression results for DeepScaler-1.5B Wu et al. (2025).