MAGO: BEYOND FIXED HYPERPARAMETERS WITH MULTI-OBJECTIVE PARETO OPTIMIZATION FOR HYBRID LLM REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) with advanced step-by-step reasoning capabilities have achieved remarkable performance in complex problem-solving through chain-of-thought (CoT) reasoning. However, uniformly applying elaborate reasoning to all queries creates substantial computational inefficiency, as many problems can be solved directly without extended reasoning chains. Current hybrid reasoning approaches rely on static hyperparameters and heuristic single-objective optimization, leading to suboptimal trade-offs and poor adaptation to varying task complexities. To address these limitations, we propose a multi-objective adaptive generation optimization (MAGO) framework, which integrates multi-objective optimization with dynamic adaptive weighting into hybrid reasoning. MAGO optimizes three competing objectives simultaneously: accuracy (maintaining solution correctness), efficiency (minimizing computational costs through appropriate mode selection), and calibration (ensuring mode selection aligns with model capabilities). The framework employs Pareto frontier maintenance with correlation-aware optimization to automatically explore the full trade-off space, avoiding the spatial constraints that limit fixed-weight approaches to narrow cone-shaped regions of the objective space. Unlike existing methods requiring manual hyperparameter tuning, MAGO's Pareto optimization dynamically adapts weights based on task complexity and training progress, achieving principled and adaptive decision-making across varying problem complexities. Comprehensive evaluation on mathematical reasoning benchmarks including AIME, Minerva Algebra, MATH-500, and GSM-8K shows $2.2\times$ to $3\times$ token-efficiency gains and relative accuracy improvements of 0.6% to 9.4% over heuristic baselines, while remaining competitive with the strongest task-specific models. Additional experiments on CommonsenseQA and MedQA further confirm the framework's generalizability beyond mathematics, achieving 1 to 2% higher accuracy and approximately $2\times$ efficiency improvement without additional fine-tuning.

1 INTRODUCTION

Recent breakthroughs in step-by-step reasoning capabilities have enabled LLMs to achieve unprecedented performance in complex problem-solving. Reasoning-enabled models such as DeepSeek-R1 (DeepSeek-AI, 2025) and Claude (Anthropic, 2025) employ CoT reasoning (Wei et al., 2022) to decompose complex problems into manageable sub-steps, thereby simulating human cognitive processes (Nye et al., 2021; Jung et al., 2022). This paradigm has proven particularly effective in mathematical reasoning (Hendrycks et al., 2021; Cobbe et al., 2021) and logical inference tasks (Saha et al., 2020; Wang et al., 2023).

However, uniformly applying elaborate reasoning to all queries creates significant efficiency problems in practical deployment scenarios. Large-scale deployment scenarios must handle diverse query types ranging from simple factual questions requiring direct answers to complex multi-step problems necessitating extensive reasoning (Rajpurkar et al., 2018; Khashabi et al., 2020). Indiscriminate use of reasoning models for all inputs leads to substantial computational waste, as reasoning models generate hundreds to thousands of tokens for problems that could be solved with direct answers, resulting in 5 to 20 times higher resource consumption compared to non-reasoning approaches (Kaplan et al., 2020; Hoffmann et al., 2022; Suzgun et al., 2022; Fu et al., 2023).

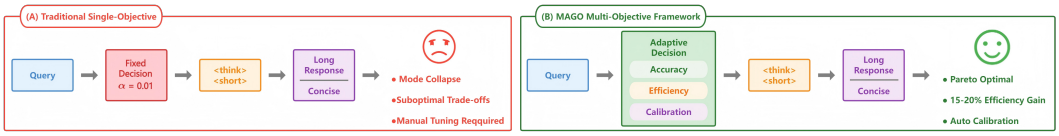

Figure 1: (A) Traditional single-objective approaches and (B) MAGO's multi-objective framework.

To address the substantial computational costs and resource consumption inherent in reasoning-enabled models, current research has concentrated on several key directions to improve inference efficiency. Hybrid reasoning mode selection approaches develop systems that dynamically choose between detailed reasoning and concise response generation through learnable control mechanisms (Fang et al., 2025; Zelikman et al., 2024; Raposo et al., 2024), utilizing specialized optimization algorithms for adaptive mode switching. Test-time compute scaling techniques allocate computational resources dynamically during inference to optimize the trade-off between accuracy and efficiency (Snell et al., 2024; Zhang et al., 2025; Lyu et al., 2025), enabling models to achieve better performance through adaptive inference-time computation rather than larger model parameters. Token-budget-aware reasoning methods explicitly incorporate computational cost constraints into the reasoning process (Han et al., 2024), developing frameworks that balance reasoning depth with predefined computational budgets. However, these methods often produce suboptimal solutions that excel in one aspect (such as accuracy or efficiency) while sacrificing others.

To address these challenges, we propose the MAGO framework, a theoretically grounded approach that reformulates hybrid reasoning as a multi-objective optimization problem. MAGO incorporates dynamic weight adaptation mechanisms that adjust with training progress and implements Pareto frontier maintenance (Deb et al., 2002; Miettinen, 1999) with correlation-aware weight selection to support more refined reasoning decisions. This method eliminates the need for manual hyperparameter tuning while achieving mathematically sound trade-offs among three competing objectives: accuracy (maintaining solution correctness), efficiency (minimizing computational costs), and decision calibration (ensuring mode selection aligns with the model's actual capabilities) (see Figure 1). The main contributions of this paper are as follows:

- We identify two performance gaps in existing hybrid reasoning systems: (1) static weight configurations lead to model under-performance across different scenarios, and (2) strong correlations between objectives cause multi-objective optimization to under-perform.
- We propose MAGO, a multi-objective optimization framework addressing these gaps through: (1) reformulating hybrid reasoning as a multi-objective optimization problem, (2) using Pareto optimization for dynamic weight selection, and (3) achieving end-to-end integration from training to deployment with zero inference overhead.
- Our framework achieves 2.2x to 3x computational efficiency improvements while simultaneously improving accuracy by 0.6% to 9.4% across mathematical reasoning benchmarks. Cross-domain evaluation on CommonsenseQA and MedQA demonstrates generalizability beyond mathematics without fine-tuning.

2 Motivation

In this section, we introduce static weight challenges and multi-objective optimization challenges.

Challenge #1: Static Weight. Current hybrid reasoning approaches (Fang et al., 2025; Shao et al., 2024) rely on fixed hyperparameters that fail to adapt to varying task complexities across different queries and training datasets. We make a series of attempts across various α values on mathematical reasoning benchmarks to explore this limitation. The results reveal three critical limitations of static weighting schemes. First, different α values cause severe mode selection imbalances (Figure 2A): α = 0.0001 leads to over 90% short-mode usage, sacrificing accuracy on complex problems, while α = 0.01 results in over 80% think-mode usage, negating efficiency gains. Second, optimal α varies significantly across datasets (Figure 2B), and no single fixed weight achieves consistent performance across diverse problem types. Third, exhaustive hyperparameter search for optimal α values is computationally prohibitive, requiring independent model training for each configuration with costs scaling linearly with search space size. These limitations demonstrate that static approaches can-

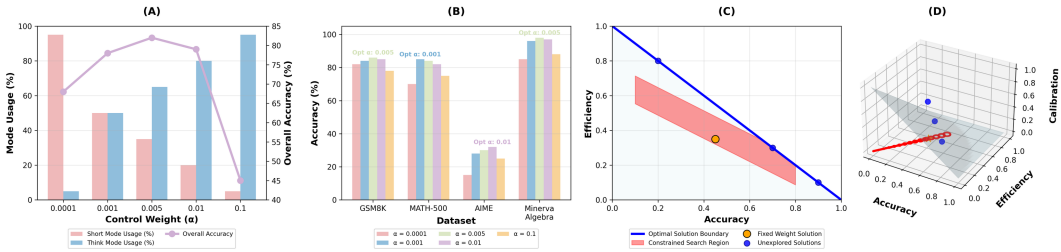

Figure 2: Static weight limitations in hybrid reasoning optimization. (A) Mode selection imbalance across control weight α values with accuracy performance. (B) Dataset-specific α sensitivity across mathematical reasoning benchmarks. (C) Fixed weight search constraints in objective space. (D) Cone-shaped optimization limitations in multi-objective landscape.

not accommodate the inherent variability in problem complexity and dataset characteristics while remaining computationally feasible. More details can be found in Appendix A.1.

Challenge #2: Multi-objective Optimization. The hybrid reasoning problem involves three competing objectives: accuracy, efficiency, and decision calibration. These objectives exhibit interdependencies creating optimization challenges. High accuracy often requires longer reasoning chains, creating tension with efficiency goals. Decision calibration considerations may favor conservative mode selection strategies, potentially affecting both accuracy and efficiency outcomes (Song et al., 2024; Wilde et al., 2024; Albeaik et al., 2024). Traditional single-weight approaches constrain optimization to narrow regions within the objective space, as illustrated in Figure 2C. Fixed weight values restrict the search trajectory to predetermined directions, preventing exploration of alternative regions with superior solutions. This spatial constraint confines optimization to limited cone-shaped regions (Figure 2D), missing optimal solutions in unexplored objective space areas. The limitation becomes pronounced when objectives exhibit different gradient scales and convergence rates, causing premature convergence to spatially constrained local optima rather than exploring the full solution landscape. More details can be found in Appendix A.2.

3 Background

In this section, we present reinforcement learning and multi-objective pareto optimization.

Reinforcement Learning. GRPO (Shao et al., 2024) provides the foundation for training hybrid reasoning models through reinforcement learning. In this framework, x represents an input query or problem that requires the model to generate a response. Given control tokens $\mathcal{C} = \{\texttt{<short>}, \texttt{<think>}\}$, the hybrid reasoning model is parameterized as a policy π_θ:

$$\pi_\theta(c, a|x) = \pi_\theta(c|x) \cdot \pi_\theta(a|x, c), \tag{1}$$

where $c \in \mathcal{C}$ denotes the reasoning mode selection and a represents the generated response sequence. The sequence $a_i = (a_{i,0}, \ldots, a_{i,T_i})$ has length $T_i + 1$, where $a_{i,0} \in \mathcal{C}$ is the control token and $(a_{i,1}, \ldots, a_{i,T_i})$ form the response (Shao et al., 2024).

The standard GRPO objective treats all tokens uniformly through a single normalization factor:

$$J_{\text{GRPO}}(\theta) = \mathbb{E}_{x,a_i}\left[\frac{1}{G}\sum_{k=1}^{G}\left(\frac{1}{T_k+1}\left[L_{k,0}(\theta) + \sum_{j=1}^{T_k} L_{k,j}(\theta)\right] - \beta D_{\text{KL}}[\pi_\theta(\cdot|x)\|\pi_{\text{ref}}(\cdot|x)]\right)\right], \tag{2}$$

where $L_{k,0}(\theta)$ and $L_{k,j}(\theta)$ represent control and response token losses, respectively. This creates two imbalances: the control token is overshadowed by T_k response tokens, and longer sequences suppress control gradients via the shared normalization $\frac{1}{T_k+1}$. DeGRPO (Fang et al., 2025) introduces separate normalization and a weight parameter α to balance mode selection against response accuracy, preventing mode collapse. More details can be found in Appendix A.3.

Multi-objective Pareto Optimization. Multi-objective optimization addresses problems with multiple competing objectives that cannot be simultaneously optimized. Rather than seeking a single optimal solution, the goal of Pareto optimization is to find the set of Pareto-optimal solutions:

$$\mathcal{P} = \{x^* \in \mathcal{X} : \nexists x \in \mathcal{X}, \mathbf{f}(x) \preceq \mathbf{f}(x^*), \mathbf{f}(x) \neq \mathbf{f}(x^*)\}, \tag{3}$$

where $\mathbf{f}(x) = [f_1(x), f_2(x), \ldots, f_m(x)]$ represents the objective vector, and $\preceq$ denotes Pareto dominance (Deb et al., 2023; Feng et al., 2021). Traditional single-objective approaches using fixed weight combinations $\sum_i \lambda_i f_i(x)$ often fail to capture the full trade-off space, as they restrict optimization to predetermined directions in the objective space. Multi-objective methods enable exploration of diverse trade-offs by adapting weights dynamically based on the problem characteristics and solution quality. More details can be found in Appendix A.4.

4 METHOD

In this section, we introduce the problem formulation and then present our solutions, including the MAGO framework, Pareto frontier maintenance, and end-to-end integration.

4.1 PROBLEM FORMULATION

In order to address the *Challenge #1* mentioned in previous sections, we formulate hybrid reasoning training as a dynamic adaptive optimization problem:

$$J(\theta) = \mathbb{E}_{x,a_i}\left[\frac{1}{G}\sum_{k=1}^{G}\left(\underbrace{m(x)}_{\text{adaptive}} L_{k,0}(\theta) + \frac{1}{T_k}\sum_{j=1}^{T_k} L_{k,j}(\theta) - \beta D_{\text{KL}}[\pi_\theta(\cdot|x)\|\pi_{\text{ref}}(\cdot|x)]\right)\right], \tag{4}$$

where $m(x)$ represents an adaptive weighting mechanism that adjusts based on input characteristics. Unlike existing approaches that rely on fixed hyperparameters, $m(x)$ dynamically adapts to balance competing training objectives without requiring manual tuning or hyperparameter search.

4.2 MULTI-OBJECTIVE ADAPTIVE GENERATION OPTIMIZATION

To realize the adaptive weighting mechanism $m(x)$ introduced in Equation 4, we propose MAGO that dynamically balances competing objectives. The framework integrates three objectives:

$$m_{\text{MAGO}}(x) = \beta_1 \cdot S_{\text{accuracy}}(x) + \beta_2 \cdot S_{\text{efficiency}}(x) + \beta_3 \cdot S_{\text{calibration}}(x), \tag{5}$$

where $(\beta_1, \beta_2, \beta_3)$ are dynamically adapted weights that automatically balance the three competing objectives without manual tuning, and the three task-specific objectives are defined below:

Accuracy Objective. The accuracy objective $S_{\text{accuracy}}(x)$ measures the correctness of responses generated under different reasoning modes:

$$S_{\text{accuracy}}(x) = \mathbb{E}_{(c,a)\sim\pi_\theta(c,a|x)}[\mathbb{I}(\phi(a) = y^*)], \tag{6}$$

where $\mathbb{I}(\cdot)$ is the indicator function, y^* is the ground-truth answer, and $\phi(a)$ extracts the final answer from the response sequence a. This function parses the generated response to identify the concluding numerical or textual answer, enabling direct comparison with the ground truth regardless of reasoning mode length.

Efficiency Objective. The efficiency objective $S_{\text{efficiency}}(x)$ captures the potential for computational savings through appropriate mode selection by measuring the expected response efficiency:

$$S_{\text{efficiency}}(x) = \mathbb{E}_{(c,a)\sim\pi_\theta(c,a|x)}\left[1 - \frac{|a|}{T_{\max}}\right], \tag{7}$$

where $|a|$ denotes the token length of the generated response sequence a, and $T_{\max}$ represents the maximum allowed sequence length. The normalization term $\frac{|a|}{T_{\max}}$ measures the relative computational cost, and subtracting from 1 converts this to an efficiency score where values approaching 1 indicate highly efficient responses. This expectation is computed by sampling responses and calculating their average normalized efficiency.

Calibration Objective. The decision calibration objective addresses a critical challenge in hybrid reasoning: ensuring that the model's mode selection decisions are well-calibrated with its problem-solving capabilities. Specifically, when the model chooses the short reasoning mode, it should be confident that it can solve the problem correctly without extended reasoning. Conversely, when it selects the think mode, this should indicate that the problem requires more elaborate reasoning for solution. Poor calibration occurs when the model overconfidently chooses short mode for difficult problems or unnecessarily defaults to think mode for simple problems it could solve directly.

The decision calibration objective $S_{\text{calibration}}(x)$ ensures that mode selection decisions align with the model's actual capability on the specific input by measuring decision calibration quality:

$$S_{\text{calibration}}(x) = 1 - \mathbb{E}_{(c,a)\sim\pi_\theta(c,a|x)}[|P_{\text{model}}(\text{correct}|x,c) - \mathbb{I}(\phi(a) = y^*)|]. \tag{8}$$

To compute the model's confidence estimate, we first extract the raw confidence score from the final answer tokens. Let L_{answer} denote the logits over the answer vocabulary at the final token position. The raw confidence score is defined as:

$$\text{RawConf}(a) = \max(\text{softmax}(L_{\text{answer}})), \tag{9}$$

which represents the model's highest probability assignment among all possible answer tokens. We then discretize this continuous confidence score into predefined intervals:

$$b = \text{Bin}(\text{RawConf}(a)) = \lfloor \text{RawConf}(a) \times N_{\text{bins}} \rfloor, \tag{10}$$

where N_{bins} is the number of confidence bins (e.g., 5 or 10).

The model's calibrated confidence estimate is then computed using statistical calibration based on historical performance:

$$P_{\text{model}}(\text{correct}|x,c) = \text{HistoricalAccuracy}(c,b), \tag{11}$$

where $\text{HistoricalAccuracy}(c,b)$ returns the empirical accuracy for mode c in confidence bin b:

$$\text{HistoricalAccuracy}(c,b) = \frac{\sum_{t\in\mathcal{H}(c,b)} \mathbb{I}(\text{correct}_t)}{|\mathcal{H}(c,b)|}, \tag{12}$$

where $\mathcal{H}(c,b)$ represents the set of historical samples with mode c and confidence bin b, and $\mathbb{I}(\text{correct}_t)$ indicates whether sample t produced the correct answer.

The historical statistics are maintained using exponential decay to prioritize recent performance:

$$\text{HistoricalAccuracy}_{t+1}(c,b) = \lambda \cdot \text{HistoricalAccuracy}_t(c,b) + (1-\lambda) \cdot \mathbb{I}(\text{correct}_{t+1}), \tag{13}$$

where $\lambda \in (0,1)$ is the decay factor. This approach leverages the model's intrinsic confidence distribution while correcting for systematic overconfidence or underconfidence patterns through empirical calibration, requiring no additional neural components while providing more reliable confidence estimates than raw token probabilities. More details can be found in Appendix A.5.

4.3 Pareto Frontier

The Pareto frontier mechanism provides the mathematical foundation for dynamic weight adaptation in MAGO to address *Challenge #2* mentioned in previous sections. We formalize the multi-objective optimization problem as maintaining an evolving set of weight configurations $\mathcal{F}_t = \{\boldsymbol{\beta}^{(1)}, \boldsymbol{\beta}^{(2)}, ..., \boldsymbol{\beta}^{(k)}\}$, where each $\boldsymbol{\beta}^{(i)} = [\beta_1^{(i)}, \beta_2^{(i)}, \beta_3^{(i)}]$ represents a distinct combination of weights for the three competing objectives (accuracy, efficiency, and calibration). By maintaining a diverse set of non-dominated weight combinations, the Pareto optimization framework avoids the cone entrapment problem that constrains fixed-weight approaches to narrow regions of the objective space, enabling principled adaptation to varying task requirements.

At each iteration t, we evaluate the performance of weight configurations using the training batch $\mathcal{B}_t$. For a given weight vector $\boldsymbol{\beta}^{(i)}$, we define the objective vector based on batch-level performance:

$$\mathbf{S}_t(\boldsymbol{\beta}^{(i)}) = \left[\frac{1}{|\mathcal{B}_t|} \sum_{x\in\mathcal{B}_t} S_{\text{accuracy}}(x), \frac{1}{|\mathcal{B}_t|} \sum_{x\in\mathcal{B}_t} S_{\text{efficiency}}(x), \frac{1}{|\mathcal{B}_t|} \sum_{x\in\mathcal{B}_t} S_{\text{calibration}}(x) \right]_{\boldsymbol{\beta}^{(i)}}, \tag{14}$$

where $|\mathcal{B}_t|$ denotes the batch size, each component represents the average performance of the corresponding objective over the current batch, evaluated under the policy π_θ trained with weight configuration $\boldsymbol{\beta}^{(i)}$.

The Pareto frontier is maintained as the set of non-dominated weight configurations:

$$\mathcal{F}_t = \{\boldsymbol{\beta}^{(i)} \mid \nexists \boldsymbol{\beta}^{(j)} \in \mathcal{S}_t : \mathbf{S}_t(\boldsymbol{\beta}^{(j)}) \succ \mathbf{S}_t(\boldsymbol{\beta}^{(i)})\}, \tag{15}$$

where $\mathcal{S}_t$ represents the set of all evaluated weight configurations up to iteration t, superscripts (i) and (j) index different weight vectors in the frontier, and $\succ$ denotes Pareto dominance relation.

To address objective correlations that lead to cone entrapment, we introduce a correlation-aware weight selection mechanism. For each training batch $\mathcal{B}_t$ at iteration t, we compute the empirical correlation matrix between the three objectives:

$$\mathbf{C}_t[i,j] = \frac{\sum_{x\in\mathcal{B}_t}(S^{(i)}(x) - \mu_t^{(i)})(S^{(j)}(x) - \mu_t^{(j)})}{\sqrt{\sum_{x\in\mathcal{B}_t}(S^{(i)}(x) - \mu_t^{(i)})^2 \sum_{x\in\mathcal{B}_t}(S^{(j)}(x) - \mu_t^{(j)})^2}}, \tag{16}$$

where $S^{(i)}(x)$ denotes the i-th objective function (accuracy, efficiency, or calibration) evaluated on input x, and $\mu_t^{(i)} = \frac{1}{|\mathcal{B}_t|}\sum_{x\in\mathcal{B}_t} S^{(i)}(x)$ represents the batch mean of objective i. This correlation structure guides the selection of weight combinations from the current frontier, ensuring that highly correlated objectives receive balanced attention while conflicting objectives maintain proper balance.

The weight selection process employs a correlation-adaptive scoring function $\Psi_t(\boldsymbol{\beta})$ that evaluates the quality of each weight configuration in the current Pareto frontier and penalizes configurations leading to high correlation between conflicting objectives:

$$\Psi_t(\boldsymbol{\beta}) = \sum_{i=1}^{3} \beta_i \hat{S}_t^{(i)} - \beta_{\text{corr}} \sum_{i<j} |\mathbf{C}_t[i,j]| \cdot |\beta_i - \beta_j|, \tag{17}$$

where $\hat{S}_t^{(i)}$ represents the moving average of the i-th objective performance over recent iterations, and $\beta_{\text{corr}} > 0$ is a hyperparameter controlling the penalty strength for correlated objectives. The first term rewards weight configurations that emphasize well-performing objectives, while the second term $|\beta_i - \beta_j|$ penalizes unbalanced weight allocations when objectives i and j are highly correlated, encouraging more uniform distribution across correlated objectives. The optimal weight vector for the current iteration is selected as:

$$\boldsymbol{\beta}_t^* = \arg\max_{\boldsymbol{\beta}\in\mathcal{F}_t} \Psi_t(\boldsymbol{\beta}). \tag{18}$$

To prevent premature convergence and ensure frontier diversity, we employ an exploration mechanism that generates new candidate solutions through guided perturbation:

$$\boldsymbol{\beta}^{\text{new}} = \boldsymbol{\beta}_t^* + \epsilon_t \cdot \mathbf{d}, \tag{19}$$

where $\boldsymbol{\beta}_t^*$ is the currently selected optimal weight vector, $\mathbf{d}$ is sampled uniformly from the constraint surface $\{\mathbf{d} \in \mathbb{R}^3 : \|\mathbf{d}\|_2 = 1, \sum_{i=1}^3 d_i = 0\}$ to preserve weight normalization, and ϵ_t is scaled based on the current frontier diversity measure:

$$\epsilon_t = \epsilon_0 \cdot \exp\left(-\frac{D(\mathcal{F}_t)}{D_{\text{target}}}\right), \tag{20}$$

where $\epsilon_0 > 0$ is the base exploration rate hyperparameter, $D_{\text{target}} > 0$ is the target diversity threshold hyperparameter, and $D(\mathcal{F}_t) = \frac{1}{|\mathcal{F}_t|^2}\sum_{\boldsymbol{\beta}^{(i)},\boldsymbol{\beta}^{(j)}\in\mathcal{F}_t} \|\boldsymbol{\beta}^{(i)} - \boldsymbol{\beta}^{(j)}\|_2$ measures the average pairwise Euclidean distance among frontier solutions (Deb et al., 2002).

The frontier update mechanism integrates newly evaluated candidate solutions and maintains non-dominance:

$$\mathcal{F}_{t+1} = \text{NonDominated}(\mathcal{F}_t \cup \{\boldsymbol{\beta}^{\text{new}}\}) \cap \text{DiversityFilter}(\cdot, \tau_{\text{div}}), \tag{21}$$

where $\text{DiversityFilter}(\cdot, \tau_{\text{div}})$ ensures minimum pairwise distance τ_{div} between frontier solutions to prevent clustering. In implementation, the number of frontier vectors $|\mathcal{F}_t|$ grows gradually in early training and stabilizes around 20–25, remaining below the upper bound $|\mathcal{F}_{\max}| = 30$. When the limit is reached, dominated or redundant vectors are pruned through cosine-similarity filtering to preserve representative diversity. More details about algorithm convergence can be found in Appendix A.6.

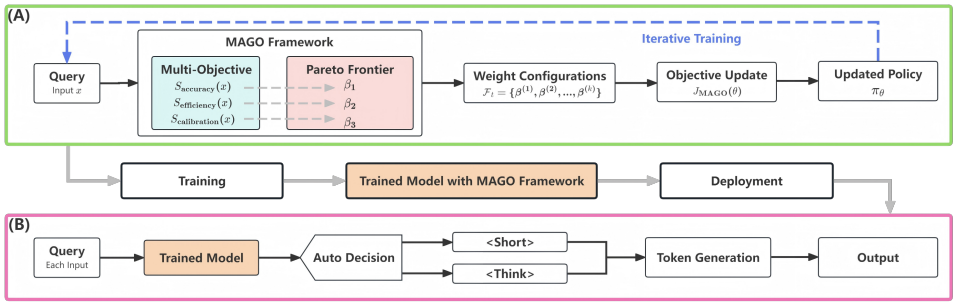

Figure 3: MAGO framework efficiently implements end-to-end integration from training to deployment and inference. (A) Training pipeline with multi-objective optimization and frontier updates. (B) Inference pipeline with learned adaptive mode selection.

4.4 End-to-End Integration

Training and Deployment. MAGO integrates into the hybrid reasoning training framework by replacing static weight parameters with dynamic multi-objective optimization (Figure 3A). At each training iteration, the system selects an optimal weight vector $\boldsymbol{\beta}_t^* = [\beta_1^*, \beta_2^*, \beta_3^*]$ from the current Pareto frontier $\mathcal{F}_t$ using correlation-aware selection (Eq. 18). The selected weights instantiate the adaptive weighting function $m_{\text{MAGO}}(x; \boldsymbol{\beta}_t^*) = \beta_1^* S_{\text{accuracy}}(x) + \beta_2^* S_{\text{efficiency}}(x) + \beta_3^* S_{\text{calibration}}(x)$, which determines the control token weight for the current batch in the training objective:

$$J(\theta) = \mathbb{E}_{x,a_i}\left[\frac{1}{G}\sum_{k=1}^{G}\left(m_{\text{MAGO}}(x;\boldsymbol{\beta}_t^*)L_{k,0}(\theta) + \frac{1}{T_k}\sum_{j=1}^{T_k} L_{k,j}(\theta) - \beta D_{\text{KL}}[\pi_\theta(\cdot|x)\|\pi_{\text{ref}}(\cdot|x)]\right)\right]. \tag{22}$$

In training (Figure 3A), the framework iteratively performs policy updates using the selected weights, evaluates objective performance on the current batch, and maintains the Pareto frontier through non-dominated sorting in this closed-loop process. During deployment (Figure 3B), the trained model automatically selects between `<short>` and `<think>` modes with zero inference overhead through learned decision-making, followed by standard token generation. More details can be found in Appendix A.7.

Training Process and Reward Design. We use a minimal reward function in training which encourages efficiency while maintaining correctness:

$$r(a, y^*, c) = \begin{cases} 1.0, & \text{if } c = \texttt{<short>} \text{ and } \phi(a) = y^*, \\ 1.0 - \gamma, & \text{if } c = \texttt{<think>} \text{ and } \phi(a) = y^*, \\ -1.0, & \text{if } \phi(a) \neq y^*, \end{cases} \tag{23}$$

where $\phi(a)$ extracts the final answer and $0 < \gamma < 1$ creates preference for efficient correct responses (Fang et al., 2025). Additional details including the relative advantage computation and token-level loss formulations are provided in Appendix A.8.

Framework Summary. While MAGO framework introduces training overhead, this cost is amortized across millions of inference queries, yielding substantial operational savings. The framework enables automatic adaptation to changing model capabilities and data characteristics, providing principled trade-offs between accuracy, efficiency, and calibration without manual tuning, operating entirely during training with zero additional inference parameters or computation.

5 Experiments

5.1 Implementation Details

Experimental Setup. We employ DeepSeek-R1-Distill-Qwen-1.5B (Guo et al., 2025) as the base model for hybrid reasoning training. To construct paired long-short response data for warm-up distillation, we leverage DeepSeek-R1-671B (Guo et al., 2025) to generate extended reasoning chains and

Qwen2.5-Math-1.5B-Instruct (Yang et al., 2024a) for concise responses. The training corpus comprises approximately 40K samples aggregated from DeepScaleR (Luo et al., 2025), OpenR1 (Face, 2025), OpenThoughts-114K (Team, 2025), and additional open-source mathematical reasoning corpora (Jebali et al., 2024; Langlais et al., 2025). To demonstrate scalability, we conduct experiments across Qwen2.5 series backbones of varying sizes (1.5B, 7B, 14B, 32B parameters) (Yang et al., 2024a). All experiments are conducted on 4 to 8 NVIDIA H100 GPUs depending on model size.

Training Configuration. Training involves supervised fine-tuning (1 epoch) followed by MAGO reinforcement learning (600 steps), implemented using VeRL (Jiang et al., 2025) and Megatron (Shoeybi et al., 2019). We optimize using AdamW (AbuKaraki et al., 2024) with learning rate 1×10^{-6}, batch size 128, weight decay 0.01, and momentum $\beta = (0.9, 0.999)$. Context length is 16K during warm-up and 24K during reinforcement learning. MAGO hyperparameters: correlation penalty $\beta_{\text{corr}} = 0.1$, exploration rate $\epsilon_0 = 0.05$, diversity threshold $\tau_{\text{div}} = 0.2$, maximum frontier size $|\mathcal{F}_{\max}| = 30$, calibration bins $N_{\text{bins}} = 10$, decay factor $\lambda = 0.95$ for historical accuracy, and reward preference $\gamma = 0.1$ favoring correct short responses.

Evaluation Benchmarks and Baselines. We evaluate on six benchmarks: AIME 2024 (Ji et al., 2025b), Minerva Algebra (Hendrycks et al., 2021), MATH-500 (Lightman et al., 2023), and GSM-8K (Cobbe et al., 2021) for mathematical reasoning, CommonsenseQA (Talmor et al., 2019) and MedQA-USMLE (Jin et al., 2021) for cross-domain generalization. All benchmarks report Pass@1 accuracy and token usage per query. We compare against three baseline categories: (1) *Base LLMs*: DeepSeek-R1-1.5B (Guo et al., 2025), Qwen2.5-1.5B-Instruct, and Qwen2.5-Math-1.5B-Instruct (Yang et al., 2024a); (2) *Shortened CoT*: Model Merging (Team et al., 2025) with coefficients (0.5, 0.6, 0.7) and CoT-Valve (Ma et al., 2025) with $\alpha \in \{4, 6, 8\}$; (3) *Hybrid Reasoning*: DeGRPO (Fang et al., 2025) with fixed $\alpha = 0.001$, random router, and Qwen2.5-7B router (Ong et al., 2024). Additional details are provided in Appendix A.9.

5.2 Result

Multi-Objective Optimization Evaluation. We first evaluate MAGO on the 1.5B backbone. Across mathematical reasoning benchmarks, MAGO yields $2.2\times$ to $3\times$ token-efficiency gains and 0.6% to 9.4% relative accuracy improvements over heuristic baselines, with consistent improvements across all evaluated tasks. Table 1 shows that MAGO achieves superior token efficiency (7,164 vs. 18,063+ baseline tokens on AIME) and competitive or superior accuracy on most benchmarks, including AIME (0.2741) and MATH-500 (0.8247), while remaining close to the best scores on Minerva Algebra (0.9483 vs. 0.9577) and GSM-8K (0.8469 vs. 0.8572). Unlike baseline methods that require dataset-specific hyperparameter tuning and router-based approaches that struggle with complex datasets, MAGO's Pareto optimization framework automatically calibrates reasoning strategies to achieve optimal efficiency–accuracy trade-offs, demonstrating the effectiveness of principled multi-objective optimization over heuristic approaches in hybrid reasoning systems. To validate scalability, we further apply MAGO to larger backbones (7B, 14B, and 32B). As model capacity increases, Pass@1 improves consistently across all benchmarks while average token usage per query decreases slightly, indicating that MAGO's Pareto optimization generalizes effectively to larger-scale models without increasing inference cost. More details can be found in Appendix A.10.

Mode Collapse in RL. Our Pareto optimization prevents mode collapse by maintaining balanced reasoning mode selection throughout training, avoiding the extreme preference for short responses that characterizes vanilla GRPO. Figure 4 (A) illustrates the *Mode Collapse* issue in standard GRPO, where the model develops an excessive preference for short outputs during training, with the number of think samples dropping precipitously to near zero within 120 training steps. In contrast, the proposed framework demonstrates significantly more stable training dynamics, maintaining a balanced distribution between think and short samples throughout the process. The vanilla GRPO's rapid collapse to predominantly short-mode usage (below 10 think samples) indicates a failure to properly balance competing objectives of accuracy and efficiency. Our Pareto-based optimization prevents this catastrophic collapse by maintaining diverse weight configurations that ensure neither reasoning mode is abandoned, enabling adaptive strategy selection based on query complexity rather than converging to suboptimal modes.

Table 1: Comparison of MAGO against baseline reasoning methods on mathematical benchmarks.

Models	Type	AIME 2024		Minerva Algebra		MATH-500		GSM8K	
		Pass@1	#Tokens	Pass@1	#Tokens	Pass@1	#Tokens	Pass@1	#Tokens
DeepSeek-R1-1.5B (Guo et al., 2025)	Base LLM	0.2800	18063	0.9577	3029	0.8608	5675	0.8347	1919
Q-1.5B (Yang et al., 2024a)		0.0200	1300	0.7771	933	0.5168	855	0.7022	466
QMath-1.5B (Yang et al., 2024a)		0.1133	1128	0.9184	586	0.7604	721	0.8572	447
Merging-0.5 (Team et al., 2025)	Short CoT	0.1333	8636	0.9292	834	0.7740	1524	0.8332	601
Merging-0.6 (Team et al., 2025)		0.1733	10615	0.9321	1091	0.7900	3000	0.8381	747
Merging-0.7 (Team et al., 2025)		0.1667	15854	0.9398	1834	0.8108	4347	0.8458	1201
CoT-Valve $\alpha = 8$ (Ma et al., 2025)		0.2000	10692	0.8079	1903	0.7060	3723	0.7726	773
CoT-Valve $\alpha = 6$ (Ma et al., 2025)		0.1933	17245	0.9468	2656	0.8024	5167	0.7970	1009
CoT-Valve $\alpha = 4$ (Ma et al., 2025)		0.2267	17722	0.9439	2965	0.8036	5820	0.8108	1396
Router Random (Fang et al., 2025)	Hybrid	0.1300	8093	0.9032	1736	0.7484	3096	0.8205	1086
Router Q-7B (Ong et al., 2024)		0.1480	9296	0.9049	795	0.7781	2748	0.8587	563
DeGRPO-Qwen-1.5B (Fang et al., 2025)	Hybrid	0.2506	7262	0.9216	1228	0.8037	2644	0.8418	649
MAGO-Qwen-1.5B (Ours)	Pareto	0.2741	7164	0.9483	1174	0.8247	2578	0.8469	633
MAGO-Qwen-7B (Ours)	Pareto	0.2960	6890	0.9562	1102	0.8424	2426	0.8611	592
MAGO-Qwen-14B (Ours)		0.3112	6724	0.9621	1041	0.8538	2368	0.8723	571
MAGO-Qwen-32B (Ours)		0.3254	6587	0.9689	992	0.8652	2294	0.8834	552

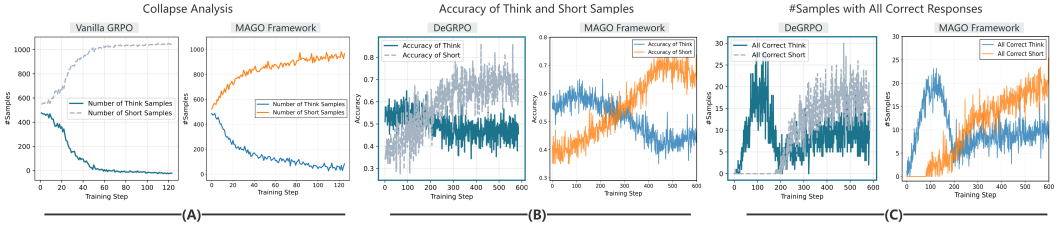

Figure 4: Training dynamics comparison between vanilla GRPO, DeGRPO, and MAGO frameworks. (A) Mode collapse analysis showing sample distribution over training steps. (B) Accuracy evolution for think and short reasoning modes during training. (C) Number of samples achieving correct responses for both reasoning modes.

Table 2: Cross-domain generalization on CommonsenseQA.

Model	Accuracy (%) ↑	Tokens / Query ↓	Token Reduction
DeGRPO	73.1	312	-
CoT-Valve	73.8	298	1.05
MAGO (ours)	**74.9**	**152**	**2.05**

The U-Shape Learning Curve. Figure 4 (B) reveals that our approach achieves more balanced training dynamics across 600 steps, with both reasoning modes converging smoothly after approximately 300 steps. While DeGRPO exhibits high volatility with fluctuations in accuracy for both modes, the proposed method demonstrates stable convergence patterns with reduced variance. The think mode maintains consistent performance around 0.6-0.7 accuracy while the short mode gradually improves from 0.4 to 0.5, contrasting sharply with DeGRPO's chaotic dynamics where accuracy fluctuates wildly between 0.3 and 0.8. Figure 4 (C) demonstrates superior sample efficiency, showing that the model quickly learns to activate short mode while ensuring correctness. The intersection point between think and short correct responses occurs later in training (around step 400), indicating more thorough exploration of reasoning mode trade-offs before settling on optimal strategies.

Cross-Domain Generalization. To evaluate the generalization ability of MAGO beyond mathematical reasoning, we perform additional experiments on CommonsenseQA, a benchmark that assesses everyday reasoning and contextual understanding. The objective is to examine whether our proposed Pareto-based adaptive optimization, trained only on mathematical reasoning data, can effectively transfer to a different reasoning domain without further fine-tuning. The same inference settings described in Section 5.1 are adopted for all methods. Representative hybrid reasoning baselines, including DeGRPO and CoT-Valve, are used for comparison. The experimental results are presented in Table 2. All results are averaged over three random seeds to ensure stability. MAGO achieves 74.9% accuracy, outperforming DeGRPO and CoT-Valve by 1.8% and 1.1%, respectively, while reducing the average number of generated tokens from 312 to 152, corresponding to a 2.05× improvement in efficiency. These findings demonstrate that MAGO's Pareto-based adaptive optimization generalizes effectively across reasoning domains and maintains a stable balance between

accuracy and computational efficiency. We also evaluate MAGO on MedQA-USMLE (Jin et al., 2021), a medical question answering benchmark, where MAGO achieves over $2.0\times$ efficiency improvement while maintaining competitive accuracy. More details can be found in Appendix A.18.

Computational Complexity. We analyze the computational and memory complexity introduced by the multi-objective optimization process. Let $|\mathcal{B}|$ denote the batch size, $M = 3$ the number of objectives, and $|\mathcal{F}_t|$ the number of maintained frontier vectors, with an upper bound $|\mathcal{F}_{\max}| = 30$.

Per-step time cost consists of several components. Computing objective statistics and scores is $\mathcal{O}(|\mathcal{B}| \cdot M)$. Correlation estimation among objectives is $\mathcal{O}(|\mathcal{B}| \cdot M^2)$. Updating the Pareto frontier requires non-dominated sorting and diversity filtering, which is $\mathcal{O}(|\mathcal{F}_t|^2 \cdot M)$, and remains constant in practice since $|\mathcal{F}_t| \leq |\mathcal{F}_{\max}|$. The guided perturbation and projection to the simplex is $\mathcal{O}(M)$. Therefore, the overall additional time complexity per step is $\mathcal{O}(|\mathcal{B}| \cdot M^2 + |\mathcal{F}_{\max}|^2 \cdot M)$, which is dominated by the batch term and treated as a small constant overhead due to the tight cap on $|\mathcal{F}_{\max}|$.

The memory cost is linear in the frontier size, $\mathcal{O}(|\mathcal{F}_{\max}| \cdot M)$, corresponding to at most 90 scalars in our configuration, plus $\mathcal{O}(M \cdot N_{\text{bins}})$ for calibration bins. All operations are performed in the objective space and are independent of model parameters, ensuring scalability to larger backbones.

6 Related Work

Reasoning (Hybrid and Efficient). Recent hybrid reasoning advances combine multiple paradigms for efficiency. Chain-of-thought and program-aided reasoning integrate natural language with code (Gao et al., 2022; Ranaldi et al., 2024), while self-refinement methods iteratively improve chains (Madaan et al., 2023; Ji et al., 2025a). Tree-of-thoughts structures reasoning as search (Yao et al., 2023; Pandey et al., 2025), adaptive frameworks select strategies by complexity (Zhou et al., 2023; Tu et al., 2025), and multi-path reasoning aggregates diverse chains (Zhu et al., 2024; Zhang et al., 2024c). Compression (Omri et al., 2025; Han et al., 2024) and selective generation (Jo et al., 2022; Yang et al., 2024b) reduce tokens while maintaining accuracy. However, these lack principled frameworks for jointly optimizing strategy selection and efficiency across diverse distributions.

Effective Reasoning (Single Methods). Single-paradigm optimizations enhance reasoning without hybridization. Prompt compression preserves semantics with 20x ratios (Jiang et al., 2023), knowledge distillation transfers capabilities to smaller models (Shridhar et al., 2023), and speculative decoding accelerates inference (Leviathan et al., 2022). Structured pruning removes redundant steps (Tao et al., 2023; Men et al., 2024), early-exit uses confidence thresholds (Tang et al., 2023; Xu et al., 2025), token-level optimization skips steps (Lee et al., 2024), and cache-based approaches reuse patterns (Yang et al., 2025a;b). These optimize singular objectives, missing opportunities.

Multi-Objective Optimization (MOO). MOO in language models balances competing goals. Pareto-optimal solutions identify accuracy-efficiency trade-offs (Mukherjee et al., 2024; Huang et al., 2024), weighted scalarization combines objectives (Yang et al., 2024c; Li & Ma, 2018), and RL optimizes multiple rewards (Zhang et al., 2024b). Constraint-based methods ensure safety (Zhang et al., 2024a; Peng et al., 2025), dynamic adjustment adapts priorities (Low & Kumar, 2024; Krishna & Vali, 2025), preference learning captures values (Dai et al., 2023; Shen et al., 2025), and evolutionary algorithms handle trade-offs (Bai et al., 2023; Li et al., 2024). However, MOO in inference mode selection remains underexplored, missing context-aware optimization opportunities.

7 Conclusion

We present MAGO, a multi-objective adaptive generation optimization framework that integrates Pareto frontier maintenance with correlation-aware weight selection for hybrid reasoning in LLMs. Our framework combines three competing objectives (accuracy, efficiency, and calibration) through dynamic weight adaptation using Pareto frontier maintenance and correlation-aware selection. This principled approach eliminates hyperparameter tuning while preventing the mode collapse observed in existing reinforcement learning methods. Experiments show that MAGO delivers $2.2\times$ to $3\times$ token-efficiency gains along with 0.6% to 9.4% relative accuracy improvements over heuristic methods on mathematical reasoning tasks. Cross-domain evaluation on CommonsenseQA and MedQA further confirms the framework's transferability beyond mathematics without additional fine-tuning.

Ethics Statement

This work adheres to the ICLR Code of Ethics. Our research focuses on developing multi-objective optimization for adaptive reasoning in large language models. We identify the following ethical considerations:

Privacy. No personally identifiable information is collected or processed.

Environmental Impact. We report detailed computational requirements in Appendix A.9.

Potential Harms. Our optimization framework could potentially be applied to harmful applications. We emphasize the importance of responsible deployment and adherence to AI safety guidelines.

Reproducibility Statement

To facilitate reproduction of our results:

Code. Complete implementation including training scripts and evaluation code will be released upon paper acceptance. For review purposes, we provide pseudocode in Appendix.

Experimental Details. Hyperparameters and experimental setup are fully specified in Appendix A.9. Hardware specifications are provided in Appendix A.9.

Data. We use publicly available datasets: AIME 2024, Minerva Algebra, MATH-500, and GSM-8K for mathematical reasoning evaluation; CommonsenseQA and MedQA-USMLE for cross-domain evaluation; DeepScaleR, OpenR1, and OpenThoughts-114K for training.

References

Anas AbuKaraki, Tawfi Alrawashdeh, Sumaya Abusaleh, Malek Zakarya Alksasbeh, Bilal Alqudah, Khalid Alemerien, and Hamzah Alshamaseen. Pulmonary edema and pleural effusion detection using efficientnet-v1-b4 architecture and adamw optimizer from chest x-rays images. *Computers, materials & continua*, 80(1), 2024.

Saer Albeaik, Sebastian Raschka, and Andrew D White. Pareto optimization to accelerate multi-objective virtual screening. *Digital Discovery*, 3(3):578–588, 2024.

Anthropic. Introducing claude 3.7 sonnet: Extended thinking. `https://www.anthropic.com/news/visible-extended-thinking`, 2025. Accessed: 2025-01-20.

Hui Bai, Ran Cheng, and Yaochu Jin. Evolutionary reinforcement learning: A survey. *ArXiv*, abs/2303.04150, 2023. doi: 10.34133/icomputing.0025.

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

Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. Safe rlhf: Safe reinforcement learning from human feedback. *ArXiv*, abs/2310.12773, 2023. doi: 10.48550/arXiv.2310.12773.

Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2): 182–197, 2002.

Kalyanmoy Deb, Karthik Sindhya, and Varun Ojha. On generalized dominance structures for multi-objective optimization. *Mathematical and Computational Applications*, 28(5):100, 2023. doi: 10.3390/mca28050100.

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Hugging Face. Open r1: A fully open reproduction of deepseek-r1, January 2025. URL https://github.com/huggingface/open-r1.

Gongfan Fang, Xinyin Ma, and Xinchao Wang. Thinkless: Llm learns when to think. *arXiv preprint arXiv:2505.13379*, 2025.

Wenqing Feng, Dunwei Gong, and Zekuan Yu. Multi-objective evolutionary optimization based on online perceiving pareto front characteristics. *Information Sciences*, 581:912–931, 2021.

Yao Fu, Hao Peng, Litu Ou, Ashish Sabharwal, and Tushar Khot. Specializing smaller language models towards multi-step reasoning. In *International Conference on Machine Learning*, pp. 10421–10430. PMLR, 2023.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. Pal: Program-aided language models. *ArXiv*, abs/2211.10435, 2022. doi: 10.48550/arXiv.2211.10435.

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

Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyun Zhao, Shiqing Ma, and Zhenyu Chen. Token-budget-aware llm reasoning. *ArXiv*, abs/2412.18547, 2024. doi: 10.48550/arXiv.2412.18547.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Yuxiao Huang, Sheng hao Wu, Wenjie Zhang, Jibin Wu, Liang Feng, and Kay Chen Tan. Autonomous multi-objective optimization using large language model. *IEEE Transactions on Evolutionary Computation*, 2024. doi: 10.1109/tevc.2025.3561001.

Mohamed Salah Jebali, Anna Valanzano, Malathi Murugesan, Giacomo Veneri, and Giovanni De Magistris. Leveraging the regularizing effect of mixing industrial and open source data to prevent overfitting of llm fine tuning. In *International Joint Conference on Artificial Intelligence 2024 Workshop on AI Governance: Alignment, Morality, and Law*, 2024.

Shihao Ji, Zihui Song, Fucheng Zhong, Jisen Jia, Zhaobo Wu, Zheyi Cao, and Tianhao Xu. Mygo multiplex cot: A method for self-reflection in large language models via double chain of thought thinking. *ArXiv*, abs/2501.13117, 2025a. doi: 10.48550/arXiv.2501.13117.

Yunjie Ji, Xiaoyu Tian, Sitong Zhao, Haotian Wang, Shuaiting Chen, Yiping Peng, Han Zhao, and Xiangang Li. Am-thinking-v1: Advancing the frontier of reasoning at 32b scale. *arXiv preprint arXiv:2505.08311*, 2025b.

Dongfu Jiang, Yi Lu, Zhuofeng Li, Zhiheng Lyu, Ping Nie, Haozhe Wang, Alex Su, Hui Chen, Kai Zou, Chao Du, et al. Verltool: Towards holistic agentic reinforcement learning with tool use. *arXiv preprint arXiv:2509.01055*, 2025.

Huiqiang Jiang, Qianhui Wu, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. Llmlingua: Compressing prompts for accelerated inference of large language models. pp. 13358–13376, 2023. doi: 10.48550/arXiv.2310.05736.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421, 2021.

DaeJin Jo, Taehwan Kwon, Eun-Sol Kim, and Sungwoong Kim. Selective token generation for few-shot natural language generation. *ArXiv*, abs/2209.08206, 2022. doi: 10.48550/arXiv.2209.08206.

Jaehun Jung, Lianhui Qin, Sean Welleck, Faeze Brahman, Chandra Bhagavatula, Ronan Le Bras, and Yejin Choi. Maieutic prompting: Logically consistent reasoning with recursive explanations. *arXiv preprint arXiv:2205.11822*, 2022.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. Unifiedqa: Crossing format boundaries with a single qa system. *arXiv preprint arXiv:2005.00700*, 2020.

Mallu Shiva Rama Krishna and D Khasim Vali. Adweh: A dynamic prioritized workflow task scheduling approach based on the enhanced harris hawk optimization algorithm. *IEEE Access*, 2025.

Pierre-Carl Langlais, Carlos Rosas Hinostroza, Mattia Nee, Catherine Arnett, Pavel Chizhov, Eliot Krzystof Jones, Irène Girard, David Mach, Anastasia Stasenko, and Ivan P Yamshchikov. Common corpus: The largest collection of ethical data for llm pre-training. *arXiv preprint arXiv:2506.01732*, 2025.

Jung Hyun Lee, June Yong Yang, Byeongho Heo, Dongyoon Han, and Kang Min Yoo. Token-supervised value models for enhancing mathematical reasoning capabilities of large language models. *ArXiv*, abs/2407.12863, 2024. doi: 10.48550/arXiv.2407.12863.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. Fast inference from transformers via speculative decoding. pp. 19274–19286, 2022. doi: 10.48550/arXiv.2211.17192.

Erchao Li and Xiang-Qi Ma. Dynamic multi-objective optimization algorithm based on prediction strategy. *Journal of Discrete Mathematical Sciences and Cryptography*, 21:411 – 415, 2018. doi: 10.1080/09720529.2018.1453625.

Jiahui Li, Geng Sun, Qingqing Wu, Dusit Niyato, Jiawen Kang, Abbas Jamalipour, and Victor CM Leung. Collaborative ground-space communications via evolutionary multi-objective deep reinforcement learning. *IEEE Journal on Selected Areas in Communications*, 2024.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*, 2023.

Siow Meng Low and Akshat Kumar. Safe reinforcement learning with learned non-markovian safety constraints. *ArXiv*, abs/2405.03005, 2024. doi: 10.48550/arXiv.2405.03005.

Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Tianjun Zhang, Li Erran Li, et al. Deepscaler: Surpassing o1-preview with a 1.5 b model by scaling rl. *Notion Blog*, 2025.

Fuyuan Lyu, Qiyuan Zhang, Zexu Sun, Lei Wang, et al. Reasoning on a budget: A survey of adaptive and controllable test-time compute in llms. *arXiv preprint arXiv:2507.02076*, 2025.

Xinyin Ma, Guangnian Wan, Runpeng Yu, Gongfan Fang, and Xinchao Wang. Cot-valve: Length-compressible chain-of-thought tuning. *arXiv preprint arXiv:2502.09601*, 2025.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36:46534–46594, 2023.

Xin Men, Mingyu Xu, Qingyu Zhang, Bingning Wang, Hongyu Lin, Yaojie Lu, Xianpei Han, and Weipeng Chen. Shortgpt: Layers in large language models are more redundant than you expect. *ArXiv*, abs/2403.03853, 2024. doi: 10.48550/arXiv.2403.03853.

Kaisa Miettinen. *Nonlinear multiobjective optimization*, volume 12. Springer Science & Business Media, 1999.

Subhojyoti Mukherjee, Anusha Lalitha, Sailik Sengupta, Aniket Deshmukh, and B. Kveton. Multi-objective alignment of large language models through hypervolume maximization. *ArXiv*, abs/2412.05469, 2024. doi: 10.48550/arXiv.2412.05469.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, et al. Show your work: Scratchpads for intermediate computation with language models. 2021.

Yasmine Omri, Parth Shroff, and Thierry Tambe. Token sequence compression for efficient multi-modal computing. *ArXiv*, abs/2504.17892, 2025. doi: 10.48550/arXiv.2504.17892.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M Waleed Kadous, and Ion Stoica. Routellm: Learning to route llms with preference data, 2024. URL https://arxiv.org/abs/2406.18665.

Tushar Pandey, A. Ghukasyan, O. Goktas, and Santosh Kumar Radha. Adaptive graph of thoughts: Test-time adaptive reasoning unifying chain, tree, and graph structures. *ArXiv*, abs/2502.05078, 2025. doi: 10.48550/arXiv.2502.05078.

Siyuan Peng, Zimeng Huangfu, Wenyun Xie, Zhijing Yang, and Feiping Nie. Dual constraint based semi-supervised nonnegative matrix factorization for multi-view clustering. *Knowledge-Based Systems*, pp. 114357, 2025.

Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*, 2018.

Leonardo Ranaldi, Giulia Pucci, Barry Haddow, and Alexandra Birch. Empowering multi-step reasoning across languages via program-aided language models. *In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 12171–12187, 2024. doi: 10.18653/v1/2024.emnlp-main.678.

David Raposo, Sam Ritter, Blake Richards, Timothy Lillicrap, Peter Conway Humphreys, and Adam Santoro. Mixture-of-depths: Dynamically allocating compute in transformer-based language models. *arXiv preprint arXiv:2404.02258*, 2024.

Swarnadeep Saha, Sayan Ghosh, Shashank Srivastava, and Mohit Bansal. Prover: Proof generation for interpretable reasoning over rules. *arXiv preprint arXiv:2010.02830*, 2020.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Yuan Shen, Luzhen Tang, Huixiao Le, Shufang Tan, Yueying Zhao, Kejie Shen, Xinyu Li, Torsten Juelich, Qiong Wang, Dragan Gašević, et al. Aligning and comparing values of chatgpt and human as learning facilitators: A value-sensitive design approach. *British Journal of Educational Technology*, 2025.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*, 2019.

Kumar Shridhar, Alessandro Stolfo, and Mrinmaya Sachan. Distilling reasoning capabilities into smaller language models. *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 7059–7073, 2023.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

Qiuye Song, Maximilian Balandat, Benjamin Letham, Eytan Bakshy, David Zhao, and Andrew Gordon Wilson. Optimal scalarizations for sublinear hypervolume regret. *arXiv preprint arXiv:2307.03288*, 2024. Published at NeurIPS 2024.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, et al. Challenging big-bench tasks and whether chain-of-thought can solve them. *arXiv preprint arXiv:2210.09261*, 2022.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4149–4158, 2019.

Peng Tang, Pengkai Zhu, Tian Li, Srikar Appalaraju, Vijay Mahadevan, and R. Manmatha. Deed: Dynamic early exit on decoder for accelerating encoder-decoder transformer models. *ArXiv*, abs/2311.08623, 2023. doi: 10.48550/arXiv.2311.08623.

Chaofan Tao, Lu Hou, Haoli Bai, Jiansheng Wei, Xin Jiang, Qun Liu, Ping Luo, and Ngai Wong. Structured pruning for efficient generative pre-trained language models. pp. 10880–10895, 2023. doi: 10.18653/v1/2023.findings-acl.692.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*, 2025.

Open Thoughts Team. Open thoughts, January 2025.

Songjun Tu, Jiahao Lin, Qichao Zhang, Xiangyu Tian, Linjing Li, Xiangyuan Lan, and Dongbin Zhao. Learning when to think: Shaping adaptive reasoning in r1-style models via multi-stage rl. *ArXiv*, 2025.

Lei Wang, Wanyu Xu, Yihuai Lan, Zhiqiang Hu, Yunshi Lan, Roy Ka-Wei Lee, and Ee-Peng Lim. Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models. *arXiv preprint arXiv:2305.04091*, 2023.

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

Nils Wilde, Stephen L Smith, and Javier Alonso-Mora. Scalarizing multi-objective robot planning problems using weighted maximization. *IEEE Robotics and Automation Letters*, 9(3):2503–2510, 2024.

Jiaming Xu, Jiayi Pan, Yongkang Zhou, Siming Chen, Jinhao Li, Yaoxiu Lian, Junyi Wu, and Guohao Dai. Specee: Accelerating large language model inference with speculative early exiting. *ArXiv*, abs/2504.08850, 2025. doi: 10.48550/arXiv.2504.08850.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024a.

Huan Yang, Renji Zhang, Mingzhe Huang, Weijun Wang, Yin Tang, Yuanchun Li, Yunxin Liu, and Deyu Zhang. Kvshare: An llm service system with efficient and effective multi-tenant kv cache reuse. *ArXiv*, 2025a. doi: https://doi.org/10.48550/arXiv.2503.16525.

Jingbo Yang, Bairu Hou, Wei Wei, Yujia Bao, and Shiyu Chang. Kvlink: Accelerating large language models via efficient kv cache reuse. *ArXiv*, abs/2502.16002, 2025b. doi: 10.48550/arXiv.2502.16002.

Kailai Yang, Zhiwei Liu, Qianqian Xie, Jimin Huang, Erxue Min, and Sophia Ananiadou. Selective preference optimization via token-level reward function estimation. *ArXiv*, abs/2408.13518, 2024b. doi: 10.48550/arXiv.2408.13518.

Rui Yang, Xiaoman Pan, Feng Luo, Shuang Qiu, Han Zhong, Dong Yu, and Jianshu Chen. Rewards-in-context: Multi-objective alignment of foundation models with dynamic preference adjustment. *ArXiv*, abs/2402.10207, 2024c. doi: 10.48550/arXiv.2402.10207.

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

Eric Zelikman, Georges Harik, Yijia Shao, Varuna Jayasiri, Nick Haber, and Noah D Goodman. Quiet-star: Language models can teach themselves to think before speaking. *arXiv preprint arXiv:2403.09629*, 2024.

Qiyuan Zhang, Shu Leng, Xiaoteng Ma, Qihan Liu, Xueqian Wang, Bin Liang, Yu Liu, and Jun Yang. Cvar-constrained policy optimization for safe reinforcement learning. *IEEE Transactions on Neural Networks and Learning Systems*, 36:830–841, 2024a. doi: 10.1109/TNNLS.2023.3331304.

Qiyuan Zhang, Fuyuan Lyu, Zexu Sun, Lei Wang, Weixu Zhang, Zhihan Guo, Yufei Wang, Niklas Muennighoff, Irwin King, Xue Liu, and Chen Ma. What, how, where, and how well? a survey on test-time scaling in large language models, 2025.

Shun Zhang, Zhenfang Chen, Sunli Chen, Yikang Shen, Zhiqing Sun, and Chuang Gan. Improving reinforcement learning from human feedback with efficient reward model ensemble. *ArXiv*, abs/2401.16635, 2024b. doi: 10.48550/arXiv.2401.16635.

Ziqi Zhang, Cunxiang Wang, Xiong Xiao, Yue Zhang, and Donglin Wang. Nash cot: Multi-path inference with preference equilibrium. *ArXiv*, pp. 14572–14587, 2024c. doi: 10.48550/arXiv.2407.07099.

Jianpeng Zhou, Wanjun Zhong, Yanlin Wang, and Jiahai Wang. Adaptive-solver framework for dynamic strategy selection in large language model reasoning. *Inf. Process. Manag.*, 62:104052, 2023. doi: 10.48550/arXiv.2310.01446.

Jiace Zhu, Yingtao Shen, Jie Zhao, and An Zou. Path-consistency: Prefix enhancement for efficient inference in llm. *ArXiv*, abs/2409.01281, 2024. doi: 10.48550/arXiv.2409.01281.

A Appendix

All appendices are provided in the supplementary text.