PRELIMINARY PREPRINT VERSION – DO NOT CITE
This is not the version of record. The final version will appear in the Proceedings of IJCAI-ECAI 2026.

Streamlining Long-Chain Reasoning via Differentiable Hierarchical Fusion

Chuangen Gao[1,2], **Wenlun Zhang**[3], **Shang Wang**[4*] and **Shuyang Gu**[5†]
[1]Key Laboratory of Computing Power Network and Information Security, Ministry of Education, Shandong Computer Science Center (National Supercomputer Center in Jinan), Qilu University of Technology (Shandong Academy of Sciences), Jinan, China
[2]Shandong Provincial Key Laboratory of Industrial Network and Information System Security, Shandong Fundamental Research Center for Computer Science, Jinan, China
[3]Keio University
[4]Independent Researcher
[5]Department of Computer Information Systems, College of Business Administration, Texas A & M University-Central Texas, 1001 Leadership Place, Killeen, TX 76549, USA
gaochuangen@qlu.edu.cn, wenlun_zhang@keio.jp, email.shangwang@gmail.com, s.gu@tamuct.edu

IJCAI-ECAI 2026 PREPRINT

Abstract

While large-scale reasoning models have achieved remarkable performance gains by scaling test-time computation through extended Chain-of-Thought (CoT) sequences, their practical utility is severely constrained by protracted inference lengths and high computational latency. In this paper, we present Differentiable Hierarchical Fusion (DHF), a novel framework that merges reasoning models with efficient base models via differentiable optimization to produce concise, accurate outputs. We introduce a dual-factor adaptive weighting mechanism to capture intra-block (Attention vs. MLP) variance and inter-block (shallow vs. deep layers) importance hierarchies, thereby addressing key limitations of static merging heuristics. Specifically, DHF optimizes fusion coefficients using gradient descent on a loss function that jointly minimizes cross-entropy (for accuracy) and response length (for conciseness). Furthermore, we enhance an outlier-aware initialization strategy to seed coefficients based on activation density and construct multi-calibration datasets to improve the model's generalization ability. Comprehensive evaluations on Qwen and LLaMA models across six reasoning benchmarks show that DHF reduces the average response length by 55% while boosting accuracy by 2.8%–6% compared to existing state-of-the-art merging methods.

1 Introduction

The evolution of Large Language Models (LLMs) has recently shifted toward Large Reasoning Models (LRMs) like DeepSeek-R1 [DeepSeek-AI *et al.*, 2025] and OpenAI o3 [OpenAI, 2025], which utilize reinforcement learning to elicit sophisticated Chain-of-Thought (CoT) behaviors [Wei *et al.*, 2022]. However, the pursuit of enhanced logical depth often comes at a significant cost, manifesting as high computational overhead and protracted latency [Wu *et al.*, 2025]. A primary driver of this inefficiency is the tendency for models to engage in "overthinking" [Chen *et al.*, 2025], a phenomenon where the model generates redundant or unnecessarily exhaustive reasoning paths for tasks that could be resolved with far less cognitive effort. This lack of adaptive computation not only inflates inference costs but also hinders the deployment of LRMs in real-time applications, necessitating a more nuanced balance between reasoning rigor and efficiency.

Model fusion [Ilharco *et al.*, 2023] has emerged as a promising paradigm for balancing accuracy and efficiency by synthesizing the strengths of base LLMs and reasoning LLMs. However, existing techniques encounter three critical bottlenecks when distilling long-form reasoning into concise outputs. First, established metric-based approaches, such as TIES-Merging [Yadav *et al.*, 2023] and DARE [Yu *et al.*, 2024], and even activation-informed methods like AIM [Nobari *et al.*, 2025] often suffer from **performance collapse** due to weight and activation noise. These methods rely on static heuristics or noisy statistics derived from limited calibration data, which frequently results in destructive interference that degrades the model's core logic during the fusion process. Second, current frameworks overlook the **heterogeneous weight distributions** inherent in transformer architectures. By applying uniform merging coefficients across the Attention and MLP modules, they overlook the divergent variance patterns exhibited by these components when transitioning from base to reasoning-heavy weights, resulting in suboptimal parameter integration. Third, traditional merging lacks a **dynamic feedback mechanism** to refine parameters based on downstream performance. Relying on fixed global coefficients or hand-crafted importance scores prevents these methods from navigating the complex Pareto frontier between reasoning depth and response conciseness, ultimately failing to achieve a truly adaptive fusion.

In this paper, we introduce **Differentiable Hierarchical**

*Corresponding author
†Corresponding author

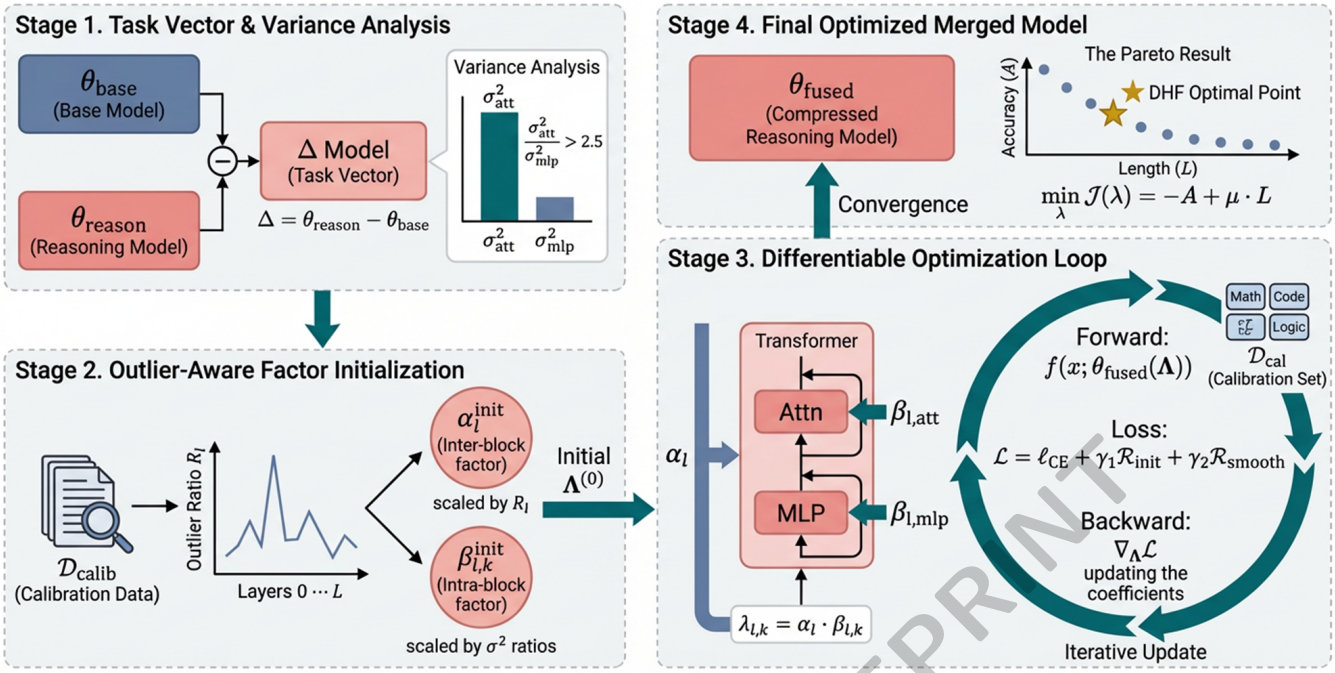

Figure 1: Overview of our Differentiable Hierarchical Fusion (DHF) method. **First**, we compute the task vector between reasoning and base models. **Then**, we perform outlier-aware initialization for dual merge factors with calibration datasets. **Next**, we apply differentiable fusion using hierarchical scaling and a combined loss function. **Finally**, we obtain we obtain the compressed reasoning model through Pareto optimization.

Fusion (DHF), a framework that casts the merging of long-to-short reasoning models as a differentiable optimization problem. As illustrated in Figure 1, our approach is motivated by two critical empirical insights derived from a systematic analysis of weight distributions in reasoning versus base models. First, we observe a significant disparity in variance profiles within transformer blocks: Attention mechanisms exhibit substantially higher weight variance (σ^2_{att}) between base and reasoning models compared to MLP sections (σ^2_{mlp}), with variance ratios frequently exceeding 2.5. This suggests that reasoning-specific knowledge is primarily localized within attention parameters, necessitating finer-grained control during the fusion process. Second, we find a hierarchical functional divergence across the model architecture: shallow layers provide the structural foundation essential for linguistic coherence, whereas deeper layers govern complex reasoning logic and verification steps. Leveraging these insights, we establish a **dual-factor adaptive weighting scheme** that optimizes intra-block coefficients $\beta_{l,k}$ for component-level fusion and inter-block coefficients α_l for layer-wise importance. This is achieved through a differentiable learning framework that minimizes cross-entropy loss on a multi-domain calibration set, following the optimization efficiency established in recent literature [Yang *et al.*, 2024]. To ensure robust convergence, we employ an **activation outlier-aware initialization** strategy, seeding coefficients based on the concentration of high-magnitude activations to preserve critical reasoning pathways. Furthermore, we employ multi-calibration set construction by sampling diverse reasoning problems from mathematics, coding, and logical domains to enhance the generalizability of learned fusion coefficients.

Experimental Results: We evaluate the performance of DHF on different mathematical reasoning benchmarks, including GSM8K, MATH 500, and AIME24. On the Qwen2.5 series, the DHF shows a mean accuracy improvement of 2.8% over four competitive merging baselines, including Task Arithmetic, TIES, DARE, and AIM. Over LLaMA-3.1-8B, the mean accuracy improvement reaches 4.4%, while simultaneously reducing average response lengths by 55%. Leading results are obtained on the large-scale OlympiadBench and CollegeMath datasets. This effectively bridges the gap between efficiency and performance, enabling practical deployment of reasoning capabilities in resource-constrained environments.

2 Related Work

2.1 Efficient Long-to-Short Reasoning

The emergence of LRMs has introduced the "overthinking" bottleneck, where models expend excessive tokens on trivial logic [Chen *et al.*, 2025]. Current mitigation strategies generally fall into three categories. **Inference-time heuristics**, such as prompt-based constraints (Concise CoT [Renze and Guven, 2024]) or explicit token budgets [Han *et al.*, 2025], offer immediate but brittle solutions that often compromise accuracy on non-trivial tasks. **Architectural and routing methods**, including RouteLLM [Ong *et al.*,

IJCAI-ECAI 2026 PREPRINT

2025] and Sketch-of-Thought [Aytes *et al.*, 2025], dynamically allocate resources but introduce additional latency via auxiliary models or symbolic processing. **Training-centric approaches** utilize length-penalized reinforcement learning [Luo *et al.*, 2025; Aggarwal and Welleck, 2025] or supervised fine-tuning on distilled concise datasets [Kang *et al.*, 2025; Munkhbat *et al.*, 2025]. While effective, these require prohibitive computational resources. DHF departs from these paradigms by *internalizing* the efficiency-accuracy trade-off within the model's weight space. Unlike prompt-based or routing methods, DHF requires no inference-time overhead. Unlike RL-based methods, it achieves competitive conciseness through a brief, differentiable optimization on a tiny calibration set, effectively bridging the gap between training-free merging and full-scale fine-tuning.

2.2 Model Fusion

Model fusion (also known as model merging) enables the fusion of specialized capabilities into a single backbone without the cost of multi-task training [Wortsman *et al.*, 2022]. Foundational techniques like Task Arithmetic [Ilharco *et al.*, 2023] utilize linear combinations of task vectors, while subsequent works like TIES-Merging [Yadav *et al.*, 2023] and DARE [Yu *et al.*, 2024] focus on mitigating parameter interference through sign election and delta-pruning. However, these **weight-centric methods** ignore the runtime activation dynamics that define model behavior. To address this, **activation-informed methods** [Li *et al.*, 2025a; Li *et al.*, 2025b] ACM [Yao *et al.*, 2025] use calibration data to weight important parameters. AdaMerging [Yang *et al.*, 2024] further advanced this by introducing differentiable layer-wise coefficients. Despite their progress, these methods treat transformer layers as monolithic entities, ignoring the significant variance between attention and feed-forward components. DHF extends the differentiable merging framework by introducing a **hierarchical dual-factor weighting** scheme. By optimizing both inter-block (layer) and intra-block (component) coefficients, and utilizing outlier-aware initialization [Zhu *et al.*, 2025], DHF captures the fine-grained structural nuances of reasoning models that uniform merging methods overlook.

3 Methodology

3.1 Formulation for Reasoning Model Merging

We begin by formalizing the long-to-short reasoning merging problem. Given a base model θ_{base} (efficient but less capable in complex reasoning) and a reasoning model θ_{reason} (capable but generating verbose outputs), we define the task vector as:

$$\Delta = \theta_{reason} - \theta_{base} \tag{1}$$

The standard merging approach in Task Arithmetic [Ilharco *et al.*, 2023] produces a fused model through a global linear interpolation:

$$\theta_{fused} = \theta_{base} + \lambda\Delta \tag{2}$$

where $\lambda \in [0, 1]$ is a scalar coefficient controlling the strength of reasoning capability integration. When $\lambda = 0$, we recover the base model with concise outputs; when $\lambda = 1$, we obtain the full reasoning model with verbose chains.

In the context of long-to-short reasoning, our goal is to identify configurations of merging coefficients that simultaneously minimize output length L while maximizing task accuracy A. This can be formulated as a bi-objective optimization:

$$\min_{\lambda} \quad \mathcal{J}(\lambda) = -A(\theta_{fused}(\lambda)) + \mu \cdot L(\theta_{fused}(\lambda)) \tag{3}$$

where μ is a trade-off hyperparameter. However, uniform global λ fails to capture the heterogeneous importance of different model components, leading to sub-optimal solutions.

3.2 Dual-Factor Differentiable Framework

Through systematic analysis of weight distributions in reasoning versus base models, we observe two critical structural patterns that motivate our hierarchical approach.

Intra-block Weight Variance. To quantify the structural distribution of reasoning capabilities, we analyze the variance of the task vectors within each transformer block l. For the attention (att) and feed-forward (mlp) components, the variance is defined as:

$$\sigma_c^2(l) = \text{Var}(W_{reason,c}^{(l)} - W_{base,c}^{(l)}), \quad c \in \{\text{att, mlp}\}$$

Empirical analysis of Qwen2.5-7B and DeepSeek-R1-Distill-Qwen-7B reveals a consistent disparity: $\sigma_{att}^2/\sigma_{mlp}^2 > 2.5$. This indicates that reasoning-specific knowledge is predominantly concentrated in the attention mechanisms. Consequently, uniform merging within a block is suboptimal; while attention weights require an aggressive integration of reasoning logic, MLP weights should remain closer to the base model to preserve efficiency.

Inter-block Importance Hierarchy. Analyzing layer-wise activation statistics on reasoning tasks reveals a non-uniform distribution of reasoning-critical operations. Shallow layers (first 25% of depth) exhibit higher activation stability and serve primarily structural roles in token representation, while middle-to-deep layers (50%-90% depth) show pronounced outlier activations correlated with reasoning steps. The final layers perform answer extraction and formatting, requiring less reasoning depth. This hierarchical importance pattern indicates that different layers should contribute differently to the merged model.

Based on these observations, we decompose the global merging coefficient into a hierarchical structure. For each layer $l \in \{1, \ldots, L\}$ and component $k \in \{\text{att, mlp}\}$, we define component-specific coefficients through factorization:

$$\lambda_{l,k} = \alpha_l \cdot \beta_{l,k} \tag{4}$$

where $\alpha_l \in [0, 1]$ represents the **inter-block factor** capturing layer-wise importance, and $\beta_{l,k} \in [0.5, 1.5]$ represents the **intra-block factor** modulating component-specific sensitivity relative to the layer average. The constraint $E_k[\beta_{l,k}] = 1$ ensures that intra-block factors act as multiplicative refinements rather than global scaling.

IJCAI-ECAI 2026 PREPRINT

Algorithm 1 Differentiable Hierarchical Fusion (DHF)

Input: Base model θ_{base}, Reasoning model θ_{reason}
Calibration set $\mathcal{D}_{cal}$, Initialization set $\mathcal{D}_{cal}$
Output: Optimized fused model θ_{fused}

// Stage 1: Delta Computation
Compute task vectors: $\Delta = \theta_{reason} - \theta_{base}$
Compute component variances: $\sigma^2_{att}(l), \sigma^2_{mlp}(l)$ for all l

// Stage 2: Outlier-Aware Initialization
for each layer $l = 1, \ldots, L$ **do**
  Compute activation outlier ratio R_l using $\mathcal{D}_{init}$
  Initialize: $\alpha_l^{init} = \lambda_{min} + \lambda_{scale} \cdot R_l / \max_j R_j$
  Initialize: $\beta_{l,att}^{init}, \beta_{l,mlp}^{init}$ using variance ratios
end for
Set $\Lambda^{(0)} = \{\alpha_l^{init}, \beta_{l,k}^{init}\}$

// Stage 3: Differentiable Optimization
for iteration $t = 1$ to T_{max} **do**
  Sample mini-batch $(x, y) \sim \mathcal{D}_{cal}$
  Construct $\theta_{fused}(\Lambda^{(t)})$ using Eq. 4
  Compute loss: $\mathcal{L} = \ell_{CE}(f(x; \theta_{fused}), y) + \gamma_1 \mathcal{R}_{init} + \gamma_2 \mathcal{R}_{smooth}$
  Compute gradients: $\nabla_\Lambda \mathcal{L}$
  Update: $\Lambda^{(t+1)} = \Lambda^{(t)} - \eta \nabla_\Lambda \mathcal{L}$
  Project $\alpha_l \in [0, 1]$, $\beta_{l,k} \in [0.5, 1.5]$
  if $|\mathcal{L}^{(t)} - \mathcal{L}^{(t-5)}| < \epsilon$ **then**
    Break *// Converged*
  end if
end for

// Stage 4: Final Model Construction
Construct final θ_{fused} with optimized Λ
return θ_{fused}

The fused weights for component k in layer l are then computed as:

$$W_{fused}^{(l,k)} = W_{base}^{(l,k)} + \lambda_{l,k} \cdot (W_{reason}^{(l,k)} - W_{base}^{(l,k)}) \quad (5)$$

This dual-factor formulation provides fine-grained control: α_l modulates overall reasoning depth integration across layers, while $\beta_{l,k}$ adjusts the balance between attention and MLP within each block. Together, they enable the merged model to selectively absorb reasoning logic in high-variance attention components while maintaining the base model's structural efficiency in MLP components.

3.3 Outlier-Aware Initialization

The convergence speed and final quality of differentiable optimization depend critically on initialization. Random or uniform initialization often requires hundreds of iterations and may converge to poor local minima. We leverage recent insights on activation outliers in LLMs [Zhu *et al.*, 2025; Lin *et al.*, 2024] to develop an informed initialization strategy.

Activation Outlier Analysis. We define activation outliers relative to a threshold τ, which is proportional to the mean activation magnitude within a layer. For each layer l, we calculate the outlier ratio R_l as follows:

$$R_l = \frac{|a \in A^{(l)} : |a| > \tau|}{|A^{(l)}|} \quad (6)$$

This metric serves as a proxy for information density; layers with higher outlier ratios typically exhibit more pronounced reasoning-specific patterns where the base and reasoning models diverge. By identifying these critical layers, we can prioritize the preservation of specialized characteristics that are essential for maintaining logical integrity.

Initialization Strategy. We initialize the inter-block factors α_l proportionally to the normalized outlier ratios to balance reasoning preservation and structural efficiency:

$$\alpha_l^{init} = \lambda_{min} + \lambda_{scale} \cdot \frac{R_l}{\max_j R_j} \quad (7)$$

where λ_{min} and λ_{scale} are constants that bound the initial fusion strength. This ensures that layers with high outlier concentrations begin the optimization closer to the reasoning model's weight space, while layers with lower outlier density favor the efficiency of the base model.

For intra-block factors, we initialize based on the relative variance of the components. This allows us to assign more aggressive initial merging coefficients to components that exhibit higher variance, as these typically require more nuanced integration. The initial factors for the attention and feed-forward components are formulated as:

$$\beta_{l,att}^{init} = 1 + \eta \cdot \frac{\sigma^2_{att}(l)}{\sum_k \sigma^2_k(l)}, \quad \beta_{l,mlp}^{init} = \zeta - \beta_{l,att}^{init} \quad (8)$$

where η modulates the sensitivity to variance shifts and ζ maintains a constant expected value for the coefficients within each block. This warm-start strategy guides the differentiable framework toward a stable and high-performing region of the parameter space from the outset.

Multi-Domain Calibration Protocol. To ensure that the learned fusion coefficients generalize across diverse reasoning scenarios, we construct a multi-domain calibration set $\mathcal{D}_{cal}$. We curate this set by sampling problems from a broad spectrum of domains, including mathematical reasoning, coding tasks, and logical inference. We employ stratified sampling across multiple difficulty levels to ensure a balanced representation of reasoning patterns. This diversity prevents the optimization from overfitting to specific task patterns and ensures that the resulting fusion generalizes well to unseen scenarios. By focusing on a balanced distribution of data, we achieve a stable hierarchy of layer importance that enhances the overall reliability of the fused model.

3.4 Differentiable Optimization

Unlike traditional merging methods that set coefficients through heuristics or grid search, we treat the complete set of fusion parameters $\Lambda = \{\alpha_l, \beta_{l,k}\}_{l=1,\ldots,L}^{k \in \{\text{att,mlp}\}}$ as learnable variables optimized through gradient descent. We define a task-specific loss function over a multi-calibration set $\mathcal{D}_{cal}$ containing diverse reasoning problems:

$$\mathcal{L}(\Lambda) = E_{(x,y)\sim\mathcal{D}_{cal}} [\ell_{CE}(f(x; \theta_{fused}(\Lambda)), y)] \quad (9)$$

IJCAI-ECAI 2026 PREPRINT

where ℓ_{CE} denotes cross-entropy loss, $f(\cdot;\theta)$ represents the model's output distribution, and $\theta_{fused}(\Lambda)$ is constructed according to Equation 4. This formulation directly optimizes for task performance rather than proxy metrics [Liu *et al.*, 2018].

The optimization proceeds through gradient descent with respect to the fusion coefficients:

$$\Lambda^{(t+1)} = \Lambda^{(t)} - \eta \nabla_{\Lambda} \mathcal{L}(\Lambda^{(t)}) \tag{10}$$

where η is the learning rate. The gradient $\nabla_{\Lambda}\mathcal{L}$ is computed through backpropagation by treating the weight construction process as a differentiable operation. Specifically, for a given coefficient $\lambda_{l,k}$, the gradient is:

$$\frac{\partial \mathcal{L}}{\partial \lambda_{l,k}} = \sum_{i,j} \frac{\partial \mathcal{L}}{\partial W^{(l,k)}_{fused,ij}} \cdot (W^{(l,k)}_{reason,ij} - W^{(l,k)}_{base,ij}) \tag{11}$$

This differentiable path ensures that fusion coefficients are adjusted based on their actual contribution to task loss reduction. Layers and components that improve reasoning accuracy when merged more aggressively will receive gradient signals encouraging higher λ values, while those that maintain accuracy with conservative merging will be naturally down-weighted.

To prevent overfitting to the calibration set and ensure stable convergence, we incorporate two regularization terms. First, we add an L2 penalty encouraging coefficients to remain close to initial values:

$$\mathcal{R}_{init}(\Lambda) = \sum_{l,k} (\lambda_{l,k} - \lambda^{init}_{l,k})^2 \tag{12}$$

Second, we enforce smoothness across adjacent layers to prevent abrupt transitions:

$$\mathcal{R}_{smooth}(\Lambda) = \sum_{l=1}^{L-1} (\alpha_l - \alpha_{l+1})^2 \tag{13}$$

The complete optimization objective becomes:

$$\min_{\Lambda} \quad \mathcal{L}(\Lambda) + \gamma_1 \mathcal{R}_{init}(\Lambda) + \gamma_2 \mathcal{R}_{smooth}(\Lambda) \tag{14}$$

where γ_1 and γ_2 are regularization coefficients set to 0.01 and 0.005 respectively based on validation performance.

3.5 Overall DHF Workflow

Algorithm 1 presents the complete DHF procedure, integrating all components into a coherent workflow. The method begins with delta computation and outlier analysis for initialization, followed by iterative optimization of fusion coefficients through gradient descent on the calibration set.

4 Experiments

4.1 Experimental Setup

Evaluations. We evaluate DHF using two prominent model families. For the Qwen family, we merge Qwen2.5-Math-7B (base model optimized for mathematical problem-solving) with DeepSeek-R1-Distill-Qwen-7B (reasoning model with long CoT capabilities). For the LLaMA family, we merge LLaMA-3.1-8B-Instruct with DeepSeek-R1-Distill-LLaMA-8B. Evaluation is performed across six diverse reasoning benchmarks: GSM8K [Cobbe *et al.*, 2021] (grade school math), MATH 500 [Lightman *et al.*, 2024] (challenging competition mathematics), Minerva Math [Lewkowycz *et al.*, 2022] (STEM problems), OlympiadBench [He *et al.*, 2024] (olympiad-level problems), CollegeMath [Tang *et al.*, 2024] (undergraduate-level mathematics), and AIME24 (American Invitational Mathematics Examination 2024). These benchmarks span difficulty levels from elementary to expert and cover diverse mathematical domains. We compare DHF against state-of-the-art merging methods, including Task Arithmetic (TA) [Ilharco *et al.*, 2023], TIES-Merging [Yadav *et al.*, 2023], DARE [Yu *et al.*, 2024], Twin Merge [Lu *et al.*, 2024], AIM [Nobari *et al.*, 2025], and ACM [Yao *et al.*, 2025].

Implementation Details. The calibration set $\mathcal{D}_{cal}$ contains 256 examples sampled from multiple domains (Sec. 3.4) to ensure generalizability. We employ an outlier-aware initialization strategy: inter-block factors α_l are initialized based on layer-wise outlier ratios R_l with constants $\lambda_{min} = 0.4$ and $\lambda_{scale} = 0.5$, while intra-block factors $\beta_{l,k}$ are scaled according to the relative variance of component activations. Optimization uses the Adam optimizer with learning rate $\eta = 1 \times 10^{-3}$, batch size 8, and runs for a maximum of 100 iterations with early stopping when loss change falls below $\epsilon = 1 \times 10^{-4}$. Regularization coefficients are set to $\gamma_1 = 0.01$ and $\gamma_2 = 0.005$. All experiments use greedy decoding for fair comparison; response length is measured as the number of generated tokens before the final answer. Further details are provided in the Appendix.

4.2 Main Results

Table 1 presents comprehensive results on Qwen2.5-7B models across all six benchmarks. DHF consistently achieves the best balance between accuracy and response length reduction. The results demonstrate several important findings: **(1) Superior accuracy-efficiency trade-off:** DHF achieves 52.6% average accuracy, outperforming the best baseline (ACM) by 3.1 absolute points while simultaneously providing greater length reduction (55% vs. 47%). This represents a 6.3% relative improvement in accuracy over ACM and demonstrates that differentiable optimization successfully identifies superior points in the accuracy-length trade-off space. **(2) Consistent improvements across benchmarks:** DHF shows gains across all six benchmarks, with particularly notable improvements on challenging datasets like MATH (+4.4 over ACM), AIME (+3.3), and OlympiadBench (+3.4). This indicates that the learned coefficients effectively preserve complex reasoning capabilities while eliminating verbosity. **(3) Bridging the gap to reasoning models:** DHF recovers 90.2% of the full reasoning model's accuracy (52.6% vs. 58.3%) while using only 45% of its output length. In contrast, the best baseline (ACM) recovers only 84.9% of accuracy with 53% of length. This demonstrates DHF's effectiveness in maintaining reasoning depth with dramatically improved efficiency. Table 2 shows results on LLaMA-3.1-8B models, confirming that DHF's benefits generalize across architectures. DHF maintains its advantage on LLaMA models, achieving 60.1%

IJCAI-ECAI 2026 PREPRINT

PRELIMINARY PREPRINT VERSION – DO NOT CITE
This is not the version of record. The final version will appear in the Proceedings of IJCAI-ECAI 2026.

Method	Accuracy (%)						Avg. Acc	Len. Red.
	GSM8K	MATH	Minerva	Olympiad	College	AIME		
Base Model	$57.4_{\pm 0.2}$	$45.2_{\pm 0.3}$	$22.8_{\pm 0.1}$	$28.5_{\pm 0.2}$	$35.2_{\pm 0.3}$	$13.3_{\pm 0.1}$	33.7	-
Reasoning Model	$89.5_{\pm 0.1}$	$87.8_{\pm 0.2}$	$36.0_{\pm 0.2}$	$48.3_{\pm 0.3}$	$45.0_{\pm 0.2}$	$43.3_{\pm 0.4}$	58.3	0%
Task Arithmetic	$78.5_{\pm 0.3}$	$68.2_{\pm 0.4}$	$28.4_{\pm 0.2}$	$38.7_{\pm 0.3}$	$37.6_{\pm 0.2}$	$26.7_{\pm 0.3}$	46.4	42%
TIES-Merging	$79.2_{\pm 0.2}$	$69.8_{\pm 0.3}$	$29.1_{\pm 0.1}$	$39.5_{\pm 0.2}$	$38.2_{\pm 0.3}$	$28.3_{\pm 0.2}$	47.4	45%
DARE	$76.8_{\pm 0.4}$	$67.5_{\pm 0.4}$	$27.8_{\pm 0.3}$	$37.2_{\pm 0.4}$	$36.8_{\pm 0.4}$	$25.0_{\pm 0.3}$	45.2	38%
Twin Merge	$77.9_{\pm 0.3}$	$68.9_{\pm 0.2}$	$28.7_{\pm 0.2}$	$38.9_{\pm 0.3}$	$37.4_{\pm 0.2}$	$27.5_{\pm 0.4}$	46.6	41%
AIM	$80.5_{\pm 0.2}$	$71.2_{\pm 0.3}$	$30.3_{\pm 0.2}$	$40.8_{\pm 0.2}$	$39.5_{\pm 0.1}$	$30.0_{\pm 0.3}$	48.7	48%
ACM	$81.2_{\pm 0.2}$	$72.1_{\pm 0.2}$	$30.8_{\pm 0.1}$	$41.2_{\pm 0.3}$	$39.8_{\pm 0.2}$	$31.7_{\pm 0.2}$	49.5	47%
DHF (Ours)	$\mathbf{84.1_{\pm 0.2}}$	$\mathbf{76.5_{\pm 0.1}}$	$\mathbf{33.2_{\pm 0.2}}$	$\mathbf{44.6_{\pm 0.2}}$	$\mathbf{42.3_{\pm 0.2}}$	$\mathbf{35.0_{\pm 0.1}}$	**52.6**	**55%**

Table 1: Main results on Qwen2.5-7B models. DHF achieves the highest average accuracy while providing substantial length reduction. Accuracy values are reported with standard deviation (±) over three seeds. "Len. Red." indicates percentage reduction in average response length compared to the reasoning model.

Method	GSM8K	MATH	Minerva	Avg	Len. Red.
LLaMA-3.1-8B	$83.1_{\pm 0.2}$	$44.7_{\pm 0.3}$	$22.1_{\pm 0.2}$	50.0	-
R1-LLaMA-8B	$88.2_{\pm 0.1}$	$72.5_{\pm 0.2}$	$35.8_{\pm 0.2}$	65.5	0%
SLERP	$84.2_{\pm 0.3}$	$57.6_{\pm 0.4}$	$25.9_{\pm 0.3}$	55.9	50%
DARE	$84.0_{\pm 0.4}$	$57.2_{\pm 0.4}$	$25.5_{\pm 0.3}$	55.6	48%
Task Arithmetic	$84.5_{\pm 0.3}$	$58.3_{\pm 0.3}$	$26.4_{\pm 0.2}$	56.4	52%
TIES-Merging	$85.1_{\pm 0.2}$	$59.8_{\pm 0.3}$	$27.3_{\pm 0.2}$	57.4	54%
AIM	$85.8_{\pm 0.2}$	$61.2_{\pm 0.2}$	$28.1_{\pm 0.2}$	58.4	56%
DHF (Ours)	$\mathbf{86.9_{\pm 0.2}}$	$\mathbf{63.7_{\pm 0.1}}$	$\mathbf{29.6_{\pm 0.2}}$	**60.1**	**58%**

Table 2: Extended results on LLaMA-3.1-8B models. DHF consistently outperforms traditional merging methods (SLERP, DARE, TIES) and recent baselines (AIM) across all mathematical reasoning benchmarks while maximizing length reduction.

Configuration	Avg Acc	Len. Red.	Iter.
DHF (Full)	52.6	55%	68
w/o Intra-block β	49.8	54%	72
w/o Inter-block α	48.3	52%	85
w/o Outlier Init	51.2	55%	156
w/o Multi-Calib	50.5	54%	71
w/o Regularization	51.8	56%	124
Uniform λ (Learned)	48.9	51%	92
Random Init	50.3	53%	189

Table 3: Ablation study on Qwen2.5-7B across all benchmarks. "Iter." shows average iterations to convergence.

average accuracy (2.8% absolute improvement over AIM) with 58% length reduction. This confirms that the dual-factor framework and differentiable optimization generalize beyond specific model families.

4.3 Ablation Studies

Component Analysis. Table 3 examines the impact of removing key components from the full DHF framework. The results reveal several insights: **(1) Intra-block factors are crucial:** Removing $\beta_{l,k}$ (treating all components uniformly within blocks) causes a 2.8-point accuracy drop. This validates our observation that attention and MLP components require different merging strategies due to their distinct weight variance patterns. **(2) Inter-block hierarchy**

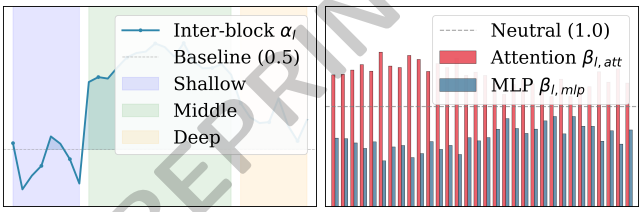

Figure 2: Learned fusion coefficients across layers. (a) Inter-block factors α_l show higher values in middle-to-deep layers where reasoning occurs. (b) Intra-block factors reveal that attention components ($\beta_{l,att}$) consistently receive higher coefficients than MLP components ($\beta_{l,mlp}$), confirming our hypothesis about component-level variance.

matters: Removing α_l (using uniform layer importance) degrades accuracy by 4.3 points, the largest single-component impact. This confirms that different layers contribute heterogeneously to reasoning capabilities and require adaptive weighting. **(3) Outlier initialization accelerates convergence:** Without outlier-aware initialization, convergence requires 2.3× more iterations (156 vs. 68) while achieving slightly lower accuracy. This demonstrates that leveraging activation statistics provides an effective warm start for optimization. **(4) Multi-calibration improves generalization:** Using a single-domain calibration set reduces average accuracy by 2.1 points, indicating that diversity in calibration data is essential for learning coefficients that generalize across reasoning tasks.

Coefficient Visualization. Figure 2 visualizes the learned coefficients α_l and $\beta_{l,k}$ for Qwen2.5-7B. The visualization reveals clear patterns: (1) α_l follows a U-shaped curve with lower values in shallow layers (0.4-0.5), peaking in middle layers (0.8-0.9), then decreasing slightly in final layers (0.6-0.7). This aligns with the expectation that reasoning logic is primarily encoded in middle-to-deep layers. (2) $\beta_{l,att}$ consistently exceeds $\beta_{l,mlp}$ across all layers (average ratio 1.15:0.85), confirming that attention mechanisms carry more reasoning information and benefit from stronger integration.

Hyperparameter Sensitivity. We analyze sensitivity to key hyperparameters: **(1) Calibration Set Size.** Figure 3

IJCAI-ECAI 2026 PREPRINT

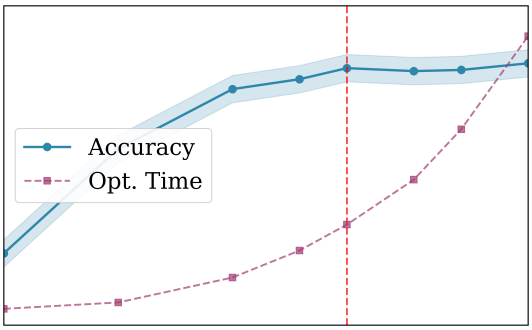

Figure 3: Average accuracy versus calibration set size. Performance plateaus after 256 examples, indicating high data efficiency.

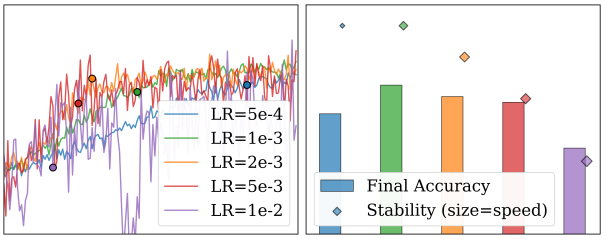

Figure 4: Learning rate sensitivity analysis: (a) Convergence curves showing validation accuracy over optimization iterations for different learning rates; (b) Final accuracy versus learning rate, with $\eta = 1 \times 10^{-3}$ achieving optimal performance.

shows performance as calibration set size varies from 64 to 512 examples. Accuracy improves rapidly from 64 (50.8%) to 256 examples (52.6%), then plateaus with minimal gains beyond 256 (52.7% at 512). This demonstrates that DHF is highly data-efficient, requiring only a small calibration set for effective coefficient learning. **(2) Learning Rate.** Figure 4 examines the sensitivity of our method to different learning rates. The optimal learning rate of $\eta = 1 \times 10^{-3}$ provides the best balance, achieving high validation accuracy (Fig. 4a) with fast and stable convergence, while also yielding the best final performance (Fig. 4b). Higher rates risk instability and reduced accuracy, while lower rates converge too slowly without gains in final performance.

Computational Efficiency. The results in Table 5 demonstrate that DHF provides a superior efficiency-to-performance ratio compared to existing baselines. While heuristic methods like TIES and DARE are faster, they fail to capture the complex reasoning capabilities of the Long-Thought model, resulting in a significant accuracy gap. Conversely, while full-parameter SFT distillation achieves respectable accuracy, it requires nearly seven times the optimization time and double the VRAM of our approach. Notably, DHF recovers approximately 99.5% of the reasoning accuracy of the Long-Thought model while reducing the output verbosity by 84.3%. This suggests that the 35-minute optimization process successfully identifies the critical "thought-dense" layers, allowing the model to retain high-level logic without the overhead of redundant tokens. The 38 GB VRAM usage further indicates

Method	Optimization Time	Peak VRAM	Avg. Length	Mean Acc.
Base (Short-Res)	-	-	182 tokens	72.4%
Base (Long-Thought)	-	-	1,245 tokens	78.3%
TIES Merging	1 min	16 GB	412 tokens	74.1%
DARE	1 min	16 GB	398 tokens	74.6%
Full SFT (Distill)	480 min	80 GB	215 tokens	76.5%
DHF (Ours)	**35 min**	**38 GB**	**195 tokens**	**77.9%**

Table 5: Comparison of computational efficiency and reasoning performance. All experiments conducted on a single NVIDIA A100 (80GB). Accuracy represents the average across GSM8K and MATH benchmarks.

that our implementation is accessible for researchers with limited compute resources, bridging the gap between simple weight averaging and expensive fine-tuning.

Method	Response Length Reduction (%)	Average Accuracy (%)	Category
DHF (Ours)	45	52.6	Proposed method
AIM	47	49.5	Baseline
ACM	53	49.5	Baseline
TIES-Merging	55	47.4	Baseline
Task Arithmetic	58	46.4	Baseline
Base Model	18	33.7	Lower bound reference
Reasoning Model	100	58.3	Upper bound reference

Table 6: Accuracy-efficiency trade-off operating points for different model merging methods.

Analysis of Length-Accuracy Trade-off. We investigate how different merging strengths affect the accuracy-length trade-off by varying the initialization of α_l and re-optimizing. Table 6 shows that DHF achieves a superior Pareto frontier compared to baseline methods. At any target response length, DHF consistently delivers higher accuracy, and conversely, for any target accuracy level, DHF requires fewer tokens. This confirms that differentiable optimization successfully identifies better trade-off points than heuristic methods.

5 Conclusions

We present Differentiable Hierarchical Fusion (DHF) for streamlining long-chain reasoning. DHF addresses the critical challenge of reducing inference verbosity while maintaining reasoning accuracy by treating fusion as a differentiable optimization problem. Our key innovation is the dual-factor adaptive weighting scheme that captures both intra-block component variance and inter-block layer importance, enabling fine-grained control over the merging process. Our DHF achieves significant reductions in response length while improving accuracy. The efficiency and effectiveness of DHF make it particularly suitable for scenarios where both computational cost and response quality matter [Li *et al.*, 2024a; Li *et al.*, 2024c; Li *et al.*, 2024d; Li *et al.*, 2024b].

Acknowledgments

This work is supported by the Shandong Provincial Natural Science Foundation under No. ZR2022QF010, and also supported by the "Jinan Science and Technology Plan 'Open Competition' Project" (Grant No. 202527031)

References

[Aggarwal and Welleck, 2025] Pranjal Aggarwal and Sean Welleck. L1: Controlling how long a reasoning model thinks with reinforcement learning, 2025.

IJCAI-ECAI 2026 PREPRINT

[Aytes *et al.*, 2025] Simon A. Aytes, Jinheon Baek, and Sung Ju Hwang. Sketch-of-thought: Efficient llm reasoning with adaptive cognitive-inspired sketching, 2025.

[Chen *et al.*, 2025] Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Do not think that much for 2+3=? on the overthinking of o1-like llms, 2025.

[Cobbe *et al.*, 2021] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021.

[DeepSeek-AI *et al.*, 2025] DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025.

[Han *et al.*, 2025] Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyu Zhao, Shiqing Ma, and Zhenyu Chen. Token-budget-aware llm reasoning, 2025.

[He *et al.*, 2024] Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. Olympiadbench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In *ACL*, pages 3828–3850. ACL, 2024.

[Ilharco *et al.*, 2023] Gabriel Ilharco, Marco Túlio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *The Eleventh ICLR, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

[Kang *et al.*, 2025] Yu Kang, Xianghui Sun, Liangyu Chen, and Wei Zou. C3ot: Generating shorter chain-of-thought without compromising effectiveness. In Toby Walsh, Julie Shah, and Zico Kolter, editors, *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, pages 24312–24320. AAAI Press, 2025.

[Lewkowycz *et al.*, 2022] 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 *NeurIPS*, volume 35, pages 3843–3857. Curran Associates, Inc., 2022.

[Li *et al.*, 2024a] Lujun Li, Yufan Bao, Peijie Dong, Chuanguang Yang, Anggeng Li, Wenhan Luo, Qifeng Liu, Wei Xue, and Yike Guo. Detkds: Knowledge distillation search for object detectors. In *ICML*, 2024.

[Li *et al.*, 2024b] Lujun Li, Peijie Dong, Zhenheng Tang, Xiang Liu, Qiang Wang, Wenhan Luo, Wei Xue, Qifeng Liu, Xiaowen Chu, and Yike Guo. Discovering sparsity allocation for layer-wise pruning of large language models. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

[Li *et al.*, 2024c] Lujun Li, Haosen Sun, Shiwen Li, Peijie Dong, Wenhan Luo, Wei Xue, Qifeng Liu, and Yike Guo. Auto-gas: Automated proxy discovery for training-free generative architecture search. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

[Li *et al.*, 2024d] Lujun Li, Zimian Wei, Peijie Dong, Wenhan Luo, Wei Xue, Qifeng Liu, and Yike Guo. Attnzero: efficient attention discovery for vision transformers. In *ECCV*, 2024.

[Li *et al.*, 2025a] Lujun Li, Dezhi Li, Cheng Lin, Wei Li, Wei Xue, Sirui Han, and Yike Guo. Aira: Activation-informed low-rank adaptation for large models. In *ICCV*, 2025.

[Li *et al.*, 2025b] Lujun Li, Cheng Lin, Dezhi Li, You-Liang Huang, Wei Li, Tianyu Wu, Jie Zou, Wei Xue, Sirui Han, and Yike Guo. Efficient fine-tuning of large models via nested low-rank adaptation. In *ICCV*, 2025.

[Lightman *et al.*, 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 *ICLR*. OpenReview.net, 2024.

[Lin *et al.*, 2024] Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. AWQ: activation-aware weight quantization for on-device LLM compression and acceleration. In *Proceedings of the Seventh Annual Conference on Machine Learning and Systems, MLSys 2024, Santa Clara, CA, USA, May 13-16, 2024*. mlsys.org, 2024.

[Liu *et al.*, 2018] Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. *arXiv preprint arXiv:1806.09055*, 2018.

[Lu *et al.*, 2024] Zhenyi Lu, Chenghao Fan, Wei Wei, Xiaoye Qu, Dangyang Chen, and Yu Cheng. Twin-merging: Dynamic integration of modular expertise in model merging, 2024.

[Luo *et al.*, 2025] 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, 2025.

[Munkhbat *et al.*, 2025] Tergel Munkhbat, Namgyu Ho, Seo Hyun Kim, Yongjin Yang, Yujin Kim, and Se-Young Yun. Self-training elicits concise reasoning in large language models, 2025.

[Nobari *et al.*, 2025] Amin Heyrani Nobari, Kaveh Alimohammadi, Ali ArjomandBigdeli, Akash Srivastava, Faez Ahmed, and Navid Azizan. Activation-informed merging of large language models, 2025.

[Ong *et al.*, 2025] 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, 2025.

IJCAI-ECAI 2026 PREPRINT

[OpenAI, 2025] OpenAI. Openai o3-mini: Pushing the frontier of cost-effective reasoning. https://openai.com/index/openai-o3-mini/, 2025. Accessed: January 31, 2025.

[Renze and Guven, 2024] Matthew Renze and Erhan Guven. The benefits of a concise chain of thought on problem-solving in large language models. In *2nd International Conference on Foundation and Large Language Models, FLLM 2024, Dubai, United Arab Emirates, November 26-29, 2024*, pages 476–483. IEEE, 2024.

[Tang *et al.*, 2024] Zhengyang Tang, Xingxing Zhang, Benyou Wang, and Furu Wei. Mathscale: Scaling instruction tuning for mathematical reasoning. In *ICML*. OpenReview.net, 2024.

[Wei *et al.*, 2022] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*, pages 24824–24837. Curran Associates, Inc., 2022.

[Wortsman *et al.*, 2022] Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak Gadre, Rebecca Roelofs, Raphael Gontijo Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *ICML*, volume 162, pages 23965–23998. PMLR, 2022.

[Wu *et al.*, 2025] Han Wu, Yuxuan Yao, Shuqi Liu, Zehua Liu, Xiaojin Fu, Xiongwei Han, Xing Li, Hui-Ling Zhen, Tao Zhong, and Mingxuan Yuan. Unlocking efficient long-to-short llm reasoning with model merging, 2025.

[Yadav *et al.*, 2023] Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models, 2023.

[Yang *et al.*, 2024] Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. Adamerging: Adaptive model merging for multi-task learning. In *The Twelfth ICLR, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

[Yao *et al.*, 2025] Yuxuan Yao, Shuqi Liu, Zehua Liu, Qintong Li, Mingyang Liu, Xiongwei Han, Zhijiang Guo, Han Wu, and Linqi Song. Activation-guided consensus merging for large language models, 2025.

[Yu *et al.*, 2024] Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Forty-first ICML, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024.

[Zhu *et al.*, 2025] Qiyuan Zhu, Lujun Li, Dezhi Li, Jiacheng Liu, Pengyu Cheng, Yucheng Xu, Sirui Han, and Yike Guo. Outlier-aware model merging for efficient multitask inference. In *Proceedings of the 33rd ACM International Conference on Multimedia*, 2025.

IJCAI-ECAI 2026 PREPRINT