DECOUPLE BEFORE INTEGRATION: POST-HOC SYNTHESIS OF SFT AND RLVR TASK VECTORS

Anonymous authors
Paper under double-blind review

ABSTRACT

SFT and RLVR improve large language models through complementary learning signals, but existing integration methods rely on training-time choices such as stage lengths, loss weights, or data-mixing ratios. Revisiting this balance after training generally requires another costly optimization run. We therefore ask whether the contributions of SFT and RLVR can instead be calibrated post hoc. Through a task-vector analysis, we find that their updates differ substantially in scale, sign overlap, and module-wise distribution, making direct composition difficult. Motivated by these differences, we propose **D**ec**o**upled **S**yn**t**hesis (DoST), which independently sparsifies the two task vectors, restores their original norms, and calibrates their relative contributions by searching only two scalar coefficients with Bayesian optimization. We further analyze when such composition can outperform either individually rescaled source. Across seven mathematical reasoning benchmarks, DoST matches or exceeds training-based SFT–RLVR integration methods, while requiring only about $2 \sim 3\%$ of other SFT–RLVR integration baselines' computation when source checkpoints are available. The learned coefficients also transfer to out-of-domain QA and coding benchmarks without re-tuning.

1 INTRODUCTION

Supervised fine-tuning (SFT) (Ouyang et al., 2022; Zhang et al., 2024) and reinforcement learning with verifiable rewards (RLVR) (Cobbe et al., 2021; Guo et al., 2025) have become two important paradigms for post-training large language models (LLMs). They improve models through different learning signals. SFT learns directly from demonstrations, allowing a model to acquire knowledge and solution patterns that it may not yet produce reliably (Gudibande et al., 2024; Allen-Zhu & Li, 2025). RLVR optimizes model-generated responses using verifiable feedback (Lightman et al., 2024) and has proved particularly effective for strengthening mathematical reasoning. Recent studies further suggest that SFT and RLVR exhibit different learning behaviors. SFT tends to expand the set of behaviors available to the model, while RLVR often strengthens capabilities already accessible to it (Rajani et al., 2025; Matsutani et al., 2025). Combining these complementary strengths is therefore a natural goal for LLM post-training.

Existing methods integrate SFT and RLVR during post-training. Sequential pipelines (Wen et al., 2025; Ma et al., 2025) first apply SFT and then continue with RLVR, whereas unified or mixed-policy approaches (Yan et al., 2025; Fu et al., 2025) interleave supervised and reinforcement updates or introduce demonstrations into the RLVR objective. Although these methods combine the two paradigms differently, the relative influence of SFT and RLVR is determined through training-time choices, such as stage lengths, loss weights, or data-mixing ratios. These choices can substantially affect the resulting model. A subsequent RLVR stage may weaken capabilities acquired through SFT (Rajani et al., 2025; Chu et al., 2025), while jointly optimized objectives can provide persistently conflicting gradients, as we observe in App C.3. More importantly, these choices are expensive to revisit: after post-training is completed, exploring a different balance between SFT and RLVR generally requires another training run. This motivates a different question: *can the contribution of SFT and RLVR be calibrated after their source models have been trained, without further parameter optimization?*

We study this possibility through task vectors (Ilharco et al., 2023), which represent the parameter changes learned independently by SFT and RLVR from a shared base model. Once two source

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

checkpoints are available, their relative contributions can in principle be adjusted directly in parameter space without revisiting either training procedure. The difficulty is that the two task vectors are far from comparable. On Qwen2.5-Math-7B, the SFT task vector has an L2 norm approximately $30\times$ larger than the RLVR task vector across layers. Moreover, 44.91% of their coordinates have opposite signs before processing, creating substantial cancellation under direct addition. Their large-magnitude updates are also distributed differently across Transformer modules: after global top-10% selection, SFT retains relatively more parameters in LayerNorm and the LM head, whereas RLVR retains relatively more in attention. Together, these observations show that post-hoc composition requires more than directly adding two independently learned updates. It must account for their different scales, reduce unnecessary coordinate overlap, and calibrate how much each source contributes to the final model.

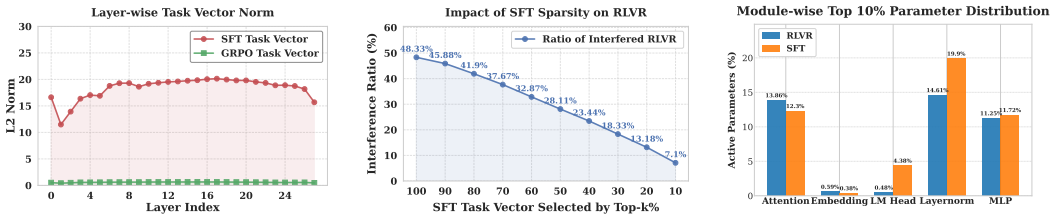
Motivated by these observations, we propose Decoupled Synthesis (DoST), a framework for composing independently trained SFT and RLVR checkpoints after source training. DoST first processes each task vector independently through magnitude-based sparsification, followed by rescaling the retained vector to its original norm. This processing preserves the dominant entries of each source update while substantially reducing their opposite-sign overlap. DoST then calibrates the relative contribution of the two processed task vectors through two scalar coefficients, λ_{SFT} and λ_{RLVR}. We select coefficients with Bayesian optimization on a small set of training problems, using average verifier accuracy across multiple sampled responses as the objective. The source task vectors remain fixed throughout the search. As a result, exploring alternative compositions requires only forward evaluation over two coefficients, rather than another round of gradient-based post-training.

This formulation also raises a basic question: when can adjusting two independently learned directions produce a model that improves over either source alone? We study this question theoretically in a stylized reward-plateau setting. In this setting, RLVR receives no useful reward gradient along a direction needed for a subset of hard questions, while supervised demonstrations allow SFT to learn that direction. RLVR, in turn, improves a different group of questions. Their composition can therefore outperform either source even after each source is given its individually optimal scaling. We further characterize when this composition gain survives sparsification and subsequent norm restoration. The analysis complements the parameter-space observations above by connecting source complementarity to the functional gain obtained after composition.

Empirically, DoST shows that post-hoc calibration can provide strong SFT–RLVR integration. On Qwen2.5-Math-7B, combining the SFT and On-Policy RL checkpoints reaches an average score of 46.8 across seven mathematical reasoning benchmarks, improving over both source models at 44.3 and 40.9, as well as the training-based LUFFY (Yan et al., 2025) baseline at 45.6. DoST can also directly compose stronger post-trained checkpoints: combining ExGRPO (Zhan et al., 2026) and ReLIFT (Ma et al., 2025) reaches an average score of 47.1. Given existing source checkpoints, coefficient search requires only 20 GPU hours, compared with 616 GPU hours for LUFFY. Moreover, the coefficients selected on mathematical reasoning problems transfer without re-tuning to ARC-C, GPQA, MMLU-Pro, and LiveCodeBench, where the composed model also improves the average performance over its source checkpoints.

Our main contributions are summarized as follows:

- We characterize the task vectors learned by SFT and RLVR and identify a $\sim 30\times$ magnitude disparity, $\sim 45\%$ opposite-sign coordinates, and different module-wise distributions of large updates. These properties reveal why directly composing the two independently trained sources is challenging.
- We propose DoST, which moves the balance between SFT and RLVR from training-time integration to post-hoc task-vector composition. DoST reduces coordinate overlap through selective sparsification and calibrates the two source contributions by searching only two scalar coefficients, without further gradient-based model training. We further analyze when such composition can outperform either individually rescaled source and when the gain survives task-vector processing.
- Across seven mathematical reasoning benchmarks, DoST matches or exceeds training-based SFT–RLVR integration methods. Given existing source checkpoints, coefficient search requires only 20 GPU hours, and the learned coefficients transfer to out-of-domain QA and coding benchmarks without re-tuning.



(a) Layer-wise L2 norm.

(b) Opposite-sign overlap under sparsification.

(c) Module-wise top-10% retention rates.

Figure 1: Task-vector analysis on Qwen2.5-Math-7B. **(a)** SFT has an approximately $30\times$ larger task-vector norm than RLVR. **(b)** Opposite-sign overlap with the top-10% RLVR entries decreases as SFT is sparsified; the two dense vectors disagree in sign on 44.91% of coordinates. **(c)** SFT and RLVR show different module-wise distributions after global top-10% magnitude selection.

2 Understanding SFT and RLVR Task Vectors

Preliminary. Following the standard formulation (Ilharco et al., 2023), given the parameters θ of a pre-trained base model, a *task vector* is defined as the element-wise difference between post-trained parameters θ_{ft} and the base model: $\tau = \theta_{\text{ft}} - \theta$. In our setting, SFT and RLVR are applied independently to the same base model θ and training dataset, yielding two task vectors: $\tau_{\text{SFT}} = \theta_{\text{SFT}} - \theta$, $\tau_{\text{RLVR}} = \theta_{\text{RLVR}} - \theta$. A merged model can then be constructed as $\theta_{\text{merged}} = \theta + \lambda_{\text{SFT}} \cdot \tau_{\text{SFT}} + \lambda_{\text{RLVR}} \cdot \tau_{\text{RLVR}}$, where λ_{SFT} and λ_{RLVR} are scalar coefficients. We analyze these task vectors on Qwen2.5-Math-7B (Yang et al., 2024a) and identify three structural properties. Together, they help explain why direct task-vector merging is difficult and motivate the design of DoST.

2.1 Finding 1: Large Magnitude Disparity

We first examine the layer-wise magnitudes of the SFT and RLVR task vectors. As shown in Fig. 1a, the L2 norm of τ_{SFT} is consistently much larger than that of τ_{RLVR} across layers. The SFT task vector has norms around 15 to 20 in most layers, whereas the RLVR task vector remains below 1. Overall, we observe $\|\tau_{\text{SFT}}\|_2 \approx 30 \times \|\tau_{\text{RLVR}}\|_2$. This scale gap is much larger than what is usually assumed in model merging, where task vectors are expected to have more comparable magnitudes.

At equal coefficients, this imbalance makes the SFT contribution much larger in Euclidean norm. It motivates independent coefficient calibration, but does not establish that the smaller update has a smaller effect on accuracy or identify the cause of the baseline scores in Tab. 1. Importantly, the smaller norm of the RLVR task vector should not be interpreted as lower importance. Prior work suggests that RLVR updates can be nearly orthogonal to pretrained weights (Zhu et al., 2025), indicating a different update geometry that direct merging must account for.

Implication for DoST. We retain high-magnitude entries and restore each retained vector to its own original norm before coefficient search. Restoration does not equalize the two source norms; independent coefficients control their relative contribution. The functional effect of changing each direction is analyzed in Sec. 4.3.

2.2 Finding 2: Opposite-sign Overlap

Magnitude alone does not capture how the two updates interact coordinate by coordinate. We therefore measure sign disagreement as the fraction of coordinates on which τ_{SFT} and τ_{RLVR} have opposite signs. Before processing, 44.91% of coordinates disagree in sign. A similar pattern appears during joint optimization: the gradient sign-conflict ratio between the SFT and GRPO objectives remains close to 0.5 throughout training (Tab. 6).

We next examine whether this overlap can be reduced by retaining only large-magnitude updates. Motivated by magnitude-based task-vector sparsification (Li et al., 2025), we keep the top-10% entries of τ_{RLVR} fixed and progressively sparsify τ_{SFT}. Fig. 1b reports the fraction of retained RLVR entries whose corresponding SFT entries have the opposite sign. With the dense SFT vector, this fraction is 48.33%; it decreases steadily to 7.1% when only the top-10% SFT entries are retained. The reduction

mainly comes from shrinking the overlap between the two supports: after low-magnitude SFT entries are removed, fewer retained RLVR coordinates receive an update from SFT in the opposite direction.

Implication for DoST. This observation motivates sparsifying the two task vectors before composition. The goal is not to treat every opposite-sign coordinate as harmful, but to reduce unnecessary overlap between the two independently learned updates while retaining their dominant components.

2.3 Finding 3: Heterogeneous Module-wise Retention Rates

After observing strong conflicts in magnitude and sign, we next ask *whether SFT and RLVR affect the same parts of the model*. We retain the global top 10% parameters by magnitude in each task vector and measure the retention rate within each Transformer module, using that module's total parameter count as the denominator.

As shown in Fig. 1c, SFT and RLVR show clearly different module-wise patterns. SFT has higher within-module retention rates in LayerNorm (19.9% vs. 14.61%) and the LM head (4.38% vs. 0.48%), whereas RLVR has a higher rate in attention (13.86% vs. 12.30%). We observe a similar pattern on LLaMA3.1-8B in Appendix C.4, especially for LayerNorm and the LM head, suggesting that this difference is not specific to Qwen2.5-Math-7B.

This finding complements Findings 1 and 2: the two sources differ in their module-wise retention patterns as well as in scale and coordinate signs. These heterogeneous update patterns motivate testing whether the sources provide complementary directions in verifier accuracy.

Implication for DoST. Finding 3 motivates evaluating post-hoc composition of independently trained sources. DoST constructs candidate directions through magnitude-based processing and selects their combination by verifier accuracy.

Summary. Together, these findings motivate treating the two source updates as distinct directions and calibrating their composition. Scale, coordinate-sign and module-retention statistics are descriptive; they do not prove preserved capabilities. The method in Sec. 4 therefore selects by verifier accuracy, while the theory identifies conditions under which useful composition directions survive processing.

3 Related Works

SFT–RLVR: Divergence and Integration. Prior work has studied the differences between SFT and RLVR from several perspectives. Chu et al. (2025) show that SFT tends to memorize training distributions, while RL generalizes better. Rajani et al. (2025) and Matsutani et al. (2025) find that SFT can overwrite pretrained knowledge, whereas RLVR tends to amplify existing capabilities. Zhu et al. (2025) further show that RLVR updates are often close to orthogonal to pretrained weights, while SFT updates are more aligned with them. Our analysis in Sec. 2 complements these studies by quantifying the SFT–RLVR difference at the task-vector level, including magnitude disparity, sign interference, and module-wise update distributions.

Recent methods attempt to combine SFT and RLVR by interleaving the two procedures or jointly optimizing their objectives (Ma et al., 2025; Yan et al., 2025; Zhang et al., 2025; Wu et al., 2025a; Fu et al., 2025; Qin & Springenberg, 2025; Wu et al., 2025b; Chen et al., 2025). These methods can be effective, but they require additional training and careful tuning to balance supervised and RL signals. In contrast, DoST decouples the two training processes and performs capability synthesis after training. This avoids directly optimizing conflicting objectives during post-training.

Model Merging. Task arithmetic (Ilharco et al., 2023) shows that task vectors from homogeneous fine-tuned models can be linearly composed. Subsequent methods such as TIES-Merging (Yadav et al., 2023) and DARE (Yu et al., 2024) improve merging robustness through sparsification and rescaling. Test-time merging methods, including AdaMerging (Yang et al., 2024b) and WUDI-Merging (Cheng et al., 2025), further adapt merging coefficients at inference time. However, these methods are mainly designed for task vectors obtained from similar training procedures, where the task vectors often have more comparable scales. As shown in Sec. 2.1, SFT and RLVR task vectors violate this assumption because their magnitudes differ by about $30\times$. This scale mismatch, together with sign interference, makes standard merging methods unreliable in our setting, as confirmed by the TIES-Merging and DARE results in Sec. 5.

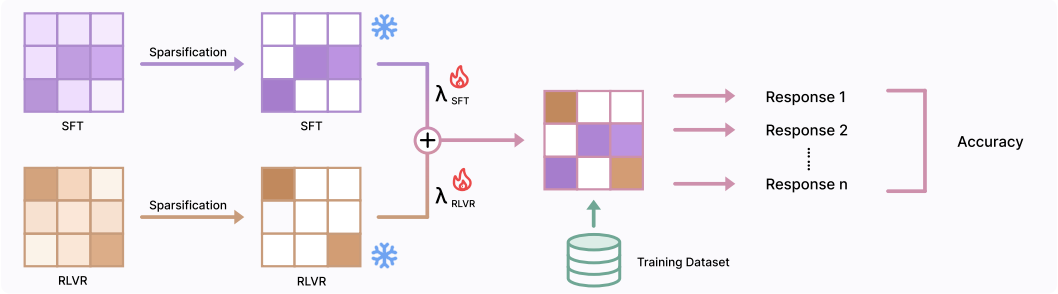


Figure 2: Overview of the DoST framework. First, each task vector is pruned by magnitude and rescaled according to its original L2 norm. After sparsification, the Bayesian optimizer searches for two scalar coefficients using average verifier accuracy across multiple sampled responses per problem. The processed task vectors remain fixed, and the highest-scoring trial determines the final merged model.

Model Merging in LLM Post-training. Model merging has been explored for LLM post-training. HMA (Lin et al., 2024) interpolates models to reduce the alignment tax in RLHF. WARP (Ramé et al., 2024) and WARM (Rame et al., 2024) average policies or reward models to improve robustness along the reward-KL trade-off. These methods merge models within the same training paradigm, where the models differ by checkpoint, reward, or training stage. By contrast, DoST aims to combine two structurally different post-training paradigms, SFT and RLVR. This setting requires explicit conflict control before task-vector synthesis, rather than relying on direct interpolation alone.

4 Methodology

The findings in Sec. 2 motivate **D**ec**o**upled **S**yn**t**hesis (**DoST**), a two-stage framework for combining independently trained SFT and RLVR checkpoints through task-vector arithmetic. Both source models are trained from the same base model θ_{base}, yielding task vectors τ_{SFT} and τ_{RLVR}. Let $\mathcal{D}_{\text{train}}$ denote their mathematical training corpus. As shown in Fig. 2 and Algo. 1, DoST first sparsifies these task vectors, then optimizes two scalar combination coefficients using answer accuracy on a small search set sampled from $\mathcal{D}_{\text{train}}$. The source task vectors remain fixed during coefficient search.

4.1 Selective Sparsification of Task Vectors

Findings 1 and 2 identify scale differences and opposite-sign overlap between the source updates. Keeping high-magnitude entries reduces this overlap, motivating the selective sparsification.

Magnitude-based pruning. Following prior work on task-vector sparsity (Li et al., 2025; Yadav et al., 2023), we retain the top fraction p of entries by absolute magnitude separately within each parameter tensor and set all other entries to zero. For tensor $\tau^{(\ell)}$ with d_ℓ entries, let t_ℓ be its $\lfloor pd_\ell \rfloor$-th largest absolute entry. We define

$$\tilde{\tau}^{(\ell)}_{\text{sparse}} = \tau^{(\ell)} \odot M^{(\ell)}, \quad M^{(\ell)}_i = \mathbb{1}\big[|\tau^{(\ell)}_i| \geq t_\ell\big]. \tag{1}$$

We retain ties at the threshold, and set a tensor to zero when $\lfloor pd_\ell \rfloor = 0$. We use $p = 0.3$ by default. This operator changes the candidate update directions.

Norm rescaling. Pruning reduces the L2 norm of a task vector, which may weaken the effect of the retained update. We therefore rescale the pruned vector using the norm of the original task vector:

$$\tilde{\tau} = \gamma \cdot \tilde{\tau}_{\text{sparse}}, \quad \gamma = \frac{\|\tau\|_2}{\|\tilde{\tau}_{\text{sparse}}\|_2}, \tag{2}$$

where both norms are global across all tensors of the same source update. A zero retained vector remains zero. This step restores each retained update to its own original norm; it does not equalize the two source norms. The remaining scale difference is handled by the coefficients in Sec. 4.2.

4.2 Bayesian Coefficient Optimization

Search set. We randomly sample n mathematical problems from $\mathcal{D}_{\text{train}}$ to form the search set $\mathcal{D}_{\text{sel}} = \{(x_i, a_i)\}_{i=1}^{n}$, where x_i is a problem and a_i is its reference answer. The calibration questions

are drawn from the source-training corpus, with exact and near lexical benchmark overlaps removed before sampling. We use $n = 256$ questions and keep the same search set across coefficient trials.

Given the sparsified task vectors $\tilde{\tau}_{\text{SFT}}$ and $\tilde{\tau}_{\text{RLVR}}$, we define the merged model as

$$\theta(\boldsymbol{\lambda}) = \theta_{\text{base}} + \lambda_{\text{SFT}}\tilde{\tau}_{\text{SFT}} + \lambda_{\text{RLVR}}\tilde{\tau}_{\text{RLVR}}, \tag{3}$$

where $\boldsymbol{\lambda} = (\lambda_{\text{SFT}}, \lambda_{\text{RLVR}}) \in [0, 2]^2$ contains the coefficients to be optimized. The two coefficients control the contribution of each task vector to the merged model.

Accuracy objective. For each candidate pair $\boldsymbol{\lambda}$, we construct the merged model using Eq. (3) and sample R responses $y_{i,1}, \ldots, y_{i,R}$ for each problem–answer pair $(x_i, a_i) \in \mathcal{D}_{\text{sel}}$. Let $V(x_i, y_{i,j}, a_i) \in \{0, 1\}$ indicate whether the verifier accepts the response as correct. We maximize the average accuracy across problems and sampled responses:

$$\widehat{A}(\boldsymbol{\lambda}) = \frac{1}{nR} \sum_{i=1}^{n} \sum_{j=1}^{R} V(x_i, y_{i,j}, a_i). \tag{4}$$

All trials use the same decoding settings and $R = 4$ responses per problem. The response protocol allows 8192 initial output tokens and up to 512 additional tokens to state a final answer after truncation or a missing final answer. This continuation is triggered without access to the reference answer and is applied consistently during search and evaluation.

Search algorithm and model selection. We use the Tree-structured Parzen Estimator (TPE) (Bergstra et al., 2011) for black-box Bayesian optimization. We set the search budget empirically to $T = 100$ trials. Each subsequent trial uses the observed coefficient–accuracy pairs to propose a new candidate. After evaluating all candidates, we select the trial with the highest observed accuracy:

$$t^* = \arg\max_{1 \le t \le T} \widehat{A}^{(t)}, \qquad \theta^* = \theta(\boldsymbol{\lambda}^{(t^*)}), \tag{5}$$

where $\widehat{A}^{(t)}$ is the score obtained in trial t. The selected coefficients are fixed when evaluating θ^* on unseen test problems.

4.3 Theoretical Analysis

We study how SFT can supply a direction missed by RLVR, and when sparsification preserves the resulting composition gain. Full definitions and proofs are in Appendix D.

We consider a stylized setting in which hard-question accuracy is locally flat near the base model, while easier-question accuracy improves along another coordinate. Demonstration likelihood guides SFT across the plateau; RLVR dynamics learn the second direction.

Combining the two learned directions. Let $A(\theta)$ be expected verifier accuracy under the fixed question distribution. For raw source updates s, r from the common base θ_0, write

$$F(a, b) = A(\theta_0 + as + br), \qquad B = \max\left\{ \max_{a \in [0,2]} F(a, 0), \max_{b \in [0,2]} F(0, b) \right\}. \tag{6}$$

Thus the comparison already gives each source its best individual coefficient.

Theorem 4.1 (Composition across a reward plateau)**.** *In the shared-policy construction, the derived SFT and RLVR sources are stationary for their respective training objectives. Let $\Delta_H > 0$ and $\Delta_E > 0$ be the hard- and easier-question accuracy gains supplied by their respective directions, weighted by the question-group probabilities. Their composition satisfies*

$$A(\theta_0 + s + r) - B = \min\{\Delta_H, \Delta_E\} > 0. \tag{7}$$

The best individual source coefficients are both 1*, within the allowed range.*

The reason is simple: either source alone leaves one group's improvement unrealized, whereas the composition obtains both. The RLVR endpoint has zero expected reward gradient and can be the stronger source; its stationarity does not imply that no better composition exists. This case requires comparing the composed model directly, since an infinitesimal SFT step at the RLVR endpoint can have zero accuracy derivative. Appendix D.3 proves the exact gap.

Table 1: Overall performance on **Qwen2.5-Math-7B**. We compare base models, SFT and RLVR methods, training-based SFT–RLVR integration methods, model merging baselines, and DoST.

Model	AIME 24	AIME 25	AIME 26	AMC	MATH500	Minerva	Olympiad	Average
Qwen2.5-Math-7B-base (Yang et al., 2024a)	14.7	6.4	8.4	49.4	70.6	27.2	35.9	30.4
Qwen2.5-Math-7B-Instruct	13.0	7.0	8.1	44.7	78.0	37.5	38.2	32.4
SFT, RLVR and Their Combined Models								
SFT	27.5	23.6	24.0	61.8	86.8	38.6	47.7	44.3
On-Policy RL	24.0	15.4	15.9	63.2	84.2	36.0	47.9	40.9
SFT+RL	26.3	22.0	20.3	62.4	84.6	38.2	48.6	43.2
LUFFY (Yan et al., 2025)	28.2	22.4	21.6	65.2	87.4	35.7	58.5	45.6
ExGRPO (Zhan et al., 2026)	30.9	17.6	18.1	65.4	86.4	36.4	52.1	43.9
ReLIFT (Ma et al., 2025)	27.6	20.5	21.0	65.5	87.2	40.8	53.8	45.2
Model Merging Baselines								
TIES-Merging (SFT + On-Policy RL)	20.9	17.7	14.8	56.7	82.4	39.7	49.2	40.2
DARE (SFT + On-Policy RL)	4.0	2.4	1.6	19.8	44.0	13.6	19.6	15.0
Our Methods								
DoST (SFT + On-Policy RL)	33.1	22.2	25.8	64.2	86.8	44.9	50.8	46.8
DoST (ExGRPO + ReLIFT)	33.8	23.2	23.1	67.0	88.6	37.1	56.7	**47.1**

When sparsification preserves the gain. For any source pair, take a dense candidate $\theta_w = \theta_0 + as + br$, with $(a, b) \in [0, 2]^2$ and gain $\Delta = A(\theta_w) - B > 0$. If the masks remove total update u, the masked candidate is $\theta_w - u$. Own-norm restoration keeps this candidate accessible by reducing its coefficients to compensate for the norm increase.

Proposition 4.2 (Gain remaining after processing)**.** *Suppose A is differentiable near the deletion segment and its gradient changes by at most L_A times the parameter distance along that segment. With own-norm restoration, the processed directions d_S, d_R satisfy*

$$\max_{\lambda \in [0,2]^2} A(\theta_0 + \lambda_S d_S + \lambda_R d_R) - B \geq \Delta - C_{\mathrm{del}}, C_{\mathrm{del}} = \nabla A(\theta_w)^\top u + \tfrac{1}{2} L_A \|u\|_2^2. \tag{8}$$

Thus a processed composition retains a positive gain whenever deletion costs less than the original margin: $C_{\mathrm{del}} < \Delta$.

Appendix D.4 proves this general bound and verifies it for the constructed sources. There, top-k% pruning removes nonzero coordinates that encode demonstration preferences without changing correctness, so the composition gain is preserved. Small changes in the source checkpoints and accuracy function also preserve the comparison under explicit error bounds (Appendix D.5).

5 Experiment

We evaluate DoST on mathematical reasoning benchmarks across different model backbones. Our main goal is to test if post-hoc task-vector synthesis can match or exceed training-based SFT–RLVR integration while using substantially less computation. The experimental settings are as follows.

Source checkpoints and task vectors. We use Qwen2.5-Math-7B (Yang et al., 2024a) as the primary backbone. The base model, SFT checkpoint, and On-Policy RL checkpoint are taken from prior work (Yan et al., 2025)[1]. The SFT and On-Policy RL checkpoints are post-trained on a subset of OpenR1-Math-220k (Hugging Face, 2025). DoST directly extracts task vectors from these public checkpoints and does not require additional training.

Evaluation benchmarks. We evaluate on seven mathematical reasoning benchmarks: AIME 2024, AIME 2025, AIME 2026, AMC (Li et al., 2024), MATH500 (Hendrycks et al., 2021), Minerva, and OlympiadBench (He et al., 2024). For benchmarks with limited test samples, including AIME 2024, AIME 2025, AIME 2026, and AMC, we report average@32. For MATH500, Minerva, and OlympiadBench, we report pass@1. Following prior work (Yan et al., 2025), we use temperature 0.6 and set the maximum generation length to 8192 tokens for all evaluations. However, we observe some final answers will be truncated due to the long response. For this situation, we provide 512 additional tokens for models to give the final answers.

Baselines. We compare with four groups of baselines. *Foundation models* include Qwen2.5-Math-7B-base and Qwen2.5-Math-7B-Instruct (Yang et al., 2024a). *Training-based SFT–RLVR integration*

[1] https://huggingface.co/collections/Elliott/luffy-rl

methods include SFT+RL, LUFFY (Yan et al., 2025) and ReLIFT (Ma et al., 2025). SFT and On-Policy RL are the source models used by DoST for task-vector extraction. *RLVR-only methods trained from the base model* include On-Policy RL and ExGRPO (Zhan et al., 2026). *Model merging baselines* include TIES-Merging (Yadav et al., 2023) and DARE (Yu et al., 2024), using the same source models as DoST.

5.1 Main Results on Qwen2.5-Math-7B

Tab. 1 reports the main results on Qwen2.5-Math-7B. We discuss the results from three perspectives.

Why do standard merging methods fail? The model merging baselines support the analysis in Sec. 2. DARE obtains an average score of 15.0, the lowest result in the table. One likely reason is that DARE assumes task-vector updates are sparse and low-magnitude, an assumption that does not hold for the SFT task vector in this setting. TIES-Merging also performs poorly, with an average score of 40.2. Its sign-election mechanism is for task vectors with comparable scales, but the large SFT–RLVR magnitude gap can cause the SFT direction to dominate the merged update and weaken the RLVR contribution. These two merging methods both fail to outperform the source models, suggesting that sparsification is useful but not sufficient. Adaptive coefficient search is also needed.

How does DoST compare to training-based integration? DoST (SFT + On-Policy RL) averages a score of 46.8, outperforming both source models, SFT at 44.3 and On-Policy RL at 40.9. It also matches or exceeds training-based integration methods, including SFT+RL at 43.2, and LUFFY at 45.6. The comparison with SFT+RL is especially relevant, as sequential training is a common way to combine SFT and RLVR. DoST improves over this baseline without additional post-training, supporting the benefit of decoupling source training from task-vector synthesis.

Does DoST work with stronger checkpoints? To test whether DoST is limited to vanilla SFT and RLVR, we apply it to ExGRPO (Zhan et al., 2026), a stronger RLVR model, and ReLIFT (Ma et al., 2025), a stronger post-trained checkpoint that already encodes complementary knowledge and reasoning behaviors on hard questions. As shown in Tab. 1, DoST (ExGRPO + ReLIFT) achieves an average of 47.1. It improves over both ExGRPO and ReLIFT by 3.2 and 1.9 points, respectively, and outperforms LUFFY by 1.5 points. This indicates that DoST can serve as a general post-hoc composition method, rather than only a remedy for weak source checkpoints.

Table 2: Comparison of computational costs.

Model	**LUFFY**	**SFT**	**On-policy RL**	**SFT+RL**	**DoST**	**DoST** w/o re-training
GPU Hours	616	192	320	1040	532	**20**

5.2 Training Efficiency

Tab. 2 compares the computational cost of DoST with training-based baselines. Because DoST operates on existing public checkpoints, its cost comes only from Bayesian coefficient search. It uses 20 GPU hours in total, compared with 616 GPU hours for LUFFY and more than 1,000 GPU hours for SFT+RL. This corresponds to approximately 3% of LUFFY's training cost and 2% of the cost of SFT+RL. Moreover, even when source models are not publicly available, DoST still possesses better training efficiency compared with other training-based baselines. As shown in Sec. 5.1, this reduction in computation does not come at the cost of performance.

5.3 OOD Generalization

A key question is whether the coefficients found on mathematical reasoning queries overfit to that domain. To test this, we evaluate the same merged model from Sec. 5.1 on four out-of-domain QA and coding benchmarks: ARC-C, GPQA, MMLU-Pro, and LiveCodeBench(LCB). The coefficients are learned on mathematical reasoning samples and directly transferred without re-tuning.

Table 3: OOD performance with strong source checkpoints. Coefficients are learned on mathematical reasoning samples and directly transferred without re-tuning.

Model	**ARC-C**	**GPQA**	**MMLU-Pro**	**LCB**	**Average**
LUFFY	80.5	39.9	53.0	18.8	48.1
ExGRPO (Zhan et al., 2026)	**84.7**	37.4	52.9	19.2	48.6
ReLIFT (Ma et al., 2025)	82.8	36.9	53.9	17.8	47.9
DoST (ExGRPO + ReLIFT)	84.6	**42.9**	**53.9**	**19.2**	**50.2**

Table 4: Ablation Study on Qwen2.5-Math-7B.

Model	AIME 24	AIME 25	AIME 26	AMC	MATH500	Minerva	Olympiad	Average
DoST (ExGRPO + ReLIFT)	33.8	23.2	23.1	67.0	88.6	37.1	56.7	**47.1**
DoST w/o Task Vector Sparsification	34.5	24.3	24.2	65.2	85.8	37.5	54.7	46.6
Dense Task Vectors with Fixed Coefficients $(1.0, 1.0)$	30.4	21.4	24.4	57.4	77.4	27.9	50.2	41.3
Dense Task Vectors with Fixed Coefficients $(0.5, 0.5)$	29.5	16.6	15.7	63.5	85.8	39.7	50.4	43.0
Dense Task Vectors with Fixed Coefficients $(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$	32.2	19.9	20.9	65.9	88.2	38.6	55.4	45.9

ficients are learned only from mathematical reasoning queries and are not re-tuned for these OOD QA and coding benchmarks.

As shown in Tab. 3, DoST (ExGRPO + ReLIFT) achieves an average score of 50.2. It outperforms LUFFY by 2.1 points, ExGRPO by 1.6 points, and ReLIFT by 2.3 points. The largest improvement appears on GPQA, where DoST improves over ExGRPO by 5.5 points. These results suggest that the learned coefficients are not merely a math-specific heuristic. They can transfer to broader OOD tasks without additional tuning.

5.4 Ablation Study

We conduct ablation studies on Qwen2.5-Math-7B, with checkpoints from ExGRPO and ReLIFT, to evaluate the contribution of each component in DoST. The results are shown in Tab. 4.

Bayesian coefficient search. On dense task vectors, Bayesian coefficient search achieves an average score of 46.6, compared with 41.3, 43.0, and 45.9 for fixed coefficient pairs of $(1.0, 1.0)$, $(0.5, 0.5)$, and $(1/\sqrt{2}, 1/\sqrt{2})$, respectively. These results support calibrating the relative contributions of the two source updates through accuracy-based search. **Task-vector sparsification**. With Bayesian coefficient search applied in both settings, sparsification further improves the average score from 46.6 to 47.1. The improvement is particularly pronounced on OlympiadBench, where accuracy increases from 54.7 to 56.7. Together, these comparisons show that coefficient calibration and task-vector sparsification both contribute to DoST's performance.

5.5 Extension to Alternative backbones

To test generalization across model scales and capabilities, we extend DoST to Qwen2.5-Math-1.5B (Yang et al., 2024a) and LLaMA3.1-8B (Grattafiori et al., 2024). Due to the length limit, the setup and results are available in Appendix C.1.

6 Conclusion

We study why integrating SFT and RLVR is difficult in LLM post-training. Through a task-vector analysis, we identify three structural properties of their task vectors: a $\sim 30\times$ magnitude disparity, $\sim 45\%$ sign interference, and heterogeneous module-wise update distributions. These findings identify challenges for direct merging and motivate testing the functional complementarity of SFT and RLVR updates. Motivated by this analysis, we propose DoST, a decoupled task-vector synthesis framework that combines independently trained SFT and RLVR checkpoints through task-vector composition. DoST uses selective sparsification with norm rescaling to construct candidate directions, and Bayesian optimization over a small subset of the shared training data to select combination coefficients by average verifier accuracy across sampled responses. In a shared-policy reward-plateau model, we derive SFT and RLVR endpoints whose composition outperforms either optimally rescaled source. A general deletion bound gives sufficient conditions for this gain to survive sparsification and norm restoration. It distinguishes direction loss from the coefficient-domain expansion induced by own-norm rescaling. The conditions concern verifier accuracy and do not imply preservation of every source behavior. Across mathematical reasoning and OOD benchmarks, DoST matches or exceeds training-based integration methods. Given existing source checkpoints, coefficient search requires 20 GPU hours, approximately 3% of LUFFY's training cost. These results suggest that post-hoc synthesis is an efficient alternative to training-time integration for heterogeneous post-training paradigms.

AI ASSISTANCE

During the preparation of this manuscript, AI-assisted tools were used to check grammar, improve language clarity, and assess the consistency and coherence of the text. The authors reviewed and verified all AI-assisted suggestions and take full responsibility for the accuracy and integrity of the final manuscript.

REFERENCES

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *The 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2623–2631, 2019.

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.3, knowledge capacity scaling laws. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=FxNNiUgtfa.

James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. *Advances in neural information processing systems*, 24, 2011.

Huayu Chen, Kaiwen Zheng, Qinsheng Zhang, Ganqu Cui, Yin Cui, Haotian Ye, Tsung-Yi Lin, Ming-Yu Liu, Jun Zhu, and Haoxiang Wang. Bridging supervised learning and reinforcement learning in math reasoning. *arXiv preprint arXiv:2505.18116*, 2025.

Runxi Cheng, Feng Xiong, Yongxian Wei, Wanyun Zhu, and Chun Yuan. Whoever started the interference should end it: Guiding data-free model merging via task vectors. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=xR9msNaREW.

Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V Le, Sergey Levine, and Yi Ma. SFT memorizes, RL generalizes: A comparative study of foundation model post-training. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=dYur3yabMj.

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.

Yuqian Fu, Tinghong Chen, Jiajun Chai, Xihuai Wang, Songjun Tu, Guojun Yin, Wei Lin, Qichao Zhang, Yuanheng Zhu, and Dongbin Zhao. Srft: A single-stage method with supervised and reinforcement fine-tuning for reasoning. *arXiv preprint arXiv:2506.19767*, 2025.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Arnav Gudibande, Eric Wallace, Charlie Victor Snell, Xinyang Geng, Hao Liu, Pieter Abbeel, Sergey Levine, and Dawn Song. The false promise of imitating proprietary language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Kz3yckpCN5.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3828–3850, 2024.

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. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. URL https://openreview.net/forum?id=7Bywt2mQsCe.

Hugging Face. Open r1: A fully open reproduction of deepseek-r1, 2025.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6t0Kwf8-jrj.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Hongkang Li, Yihua Zhang, Shuai Zhang, Pin-Yu Chen, Sijia Liu, and Meng Wang. When is task vector provably effective for model editing? a generalization analysis of nonlinear transformers. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=vRvVVb0NAz.

Jia Li, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Huang, Kashif Rasul, Longhui Yu, Albert Q Jiang, Ziju Shen, et al. Numinamath: The largest public dataset in ai4maths with 860k pairs of competition math problems and solutions. *Hugging Face repository*, 13(9):9, 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*, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

Yong Lin, Hangyu Lin, Wei Xiong, Shizhe Diao, Jianmeng Liu, Jipeng Zhang, Rui Pan, Haoxiang Wang, Wenbin Hu, Hanning Zhang, et al. Mitigating the alignment tax of rlhf. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 580–606, 2024.

Lu Ma, Hao Liang, Meiyi Qiang, Lexiang Tang, Xiaochen Ma, Zhen Hao Wong, Junbo Niu, Chengyu Shen, Runming He, Yanhao Li, et al. Learning what reinforcement learning can't: Interleaved online fine-tuning for hardest questions. *arXiv preprint arXiv:2506.07527*, 2025.

Kohsei Matsutani, Shota Takashiro, Gouki Minegishi, Takeshi Kojima, Yusuke Iwasawa, and Yutaka Matsuo. Rl squeezes, sft expands: A comparative study of reasoning llms. *arXiv preprint arXiv:2509.21128*, 2025.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Chongli Qin and Jost Tobias Springenberg. Supervised fine tuning on curated data is reinforcement learning (and can be improved). *arXiv preprint arXiv:2507.12856*, 2025.

Neel Rajani, Aryo Pradipta Gema, Seraphina Goldfarb-Tarrant, and Ivan Titov. Scalpel vs. hammer: Grpo amplifies existing capabilities, sft replaces them. *arXiv preprint arXiv:2507.10616*, 2025.

Alexandre Ramé, Johan Ferret, Nino Vieillard, Robert Dadashi, Léonard Hussenot, Pierre-Louis Cedoz, Pier Giuseppe Sessa, Sertan Girgin, Arthur Douillard, and Olivier Bachem. Warp: On the benefits of weight averaged rewarded policies. *arXiv preprint arXiv:2406.16768*, 2024.

Alexandre Rame, Nino Vieillard, Leonard Hussenot, Robert Dadashi, Geoffrey Cideron, Olivier Bachem, and Johan Ferret. WARM: On the benefits of weight averaged reward models. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=s7RDnNUJy6.

Liang Wen, Yunke Cai, Fenrui Xiao, Xin He, Qi An, Zhenyu Duan, Yimin Du, Junchen Liu, Tanglifu Tanglifu, Xiaowei Lv, et al. Light-r1: Curriculum sft, dpo and rl for long cot from scratch and beyond. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, pp. 318–327, 2025.

Jinyang Wu, Chonghua Liao, Mingkuan Feng, Shuai Zhang, Zhengqi Wen, Pengpeng Shao, Huazhe Xu, and Jianhua Tao. Thought-augmented policy optimization: Bridging external guidance and internal capabilities. *arXiv preprint arXiv:2505.15692*, 1(8):10, 2025a.

Yongliang Wu, Yizhou Zhou, Zhou Ziheng, Yingzhe Peng, Xinyu Ye, Xinting Hu, Wenbo Zhu, Lu Qi, Ming-Hsuan Yang, and Xu Yang. On the generalization of sft: A reinforcement learning perspective with reward rectification. *arXiv preprint arXiv:2508.05629*, 2025b.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. TIES-merging: Resolving interference when merging models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=xtaX3WyCj1.

Jianhao Yan, Yafu Li, Zican Hu, Zhi Wang, Ganqu Cui, Xiaoye Qu, Yu Cheng, and Yue Zhang. Learning to reason under off-policy guidance. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=vO8LLoNWWk.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, et al. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*, 2024a.

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 International Conference on Learning Representations*, 2024b. URL https://openreview.net/forum?id=nZP6NgD3QY.

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 International Conference on Machine Learning*, 2024.

Runzhe Zhan, Yafu Li, Zhi Wang, Xiaoye Qu, Dongrui Liu, Jing Shao, Derek F. Wong, and Yu Cheng. ExGRPO: Learning to reason from prior successes. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=701tjQXWVk.

Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. When scaling meets LLM finetuning: The effect of data, model and finetuning method. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=5HCnKDeTws.

Xuechen Zhang, Zijian Huang, Yingcong Li, Chenshun Ni, Jiasi Chen, and Samet Oymak. BREAD: Branched rollouts from expert anchors bridge SFT & RL for reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=NUDaln2vCe.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand, 2024. Association for Computational Linguistics. URL http://arxiv.org/abs/2403.13372.

Hanqing Zhu, Zhenyu Zhang, Hanxian Huang, DiJia Su, Zechun Liu, Jiawei Zhao, Igor Fedorov, Hamed Pirsiavash, Zhizhou Sha, Jinwon Lee, et al. The path not taken: Rlvr provably learns off the principals. *arXiv preprint arXiv:2511.08567*, 2025.

A Experimental Details

A.1 Coefficient Optimization Configuration for DoST

We use Optuna (Akiba et al., 2019) with the Tree-structured Parzen Estimator (TPE) sampler to maximize the accuracy objective in Eq. (4). Both coefficients are searched over $[0, 2]$. The empirical search budget is 100 total trials: five initial random trials followed by 95 TPE trials. We use vLLM (Kwon et al., 2023) to generate R responses per problem on the fixed search set $\mathcal{D}_{\text{sel}}$. The trial with the highest observed accuracy provides the final coefficients.

For the filtered-SFT LLaMA setting, we construct the calibration pool by retaining demonstrations of at most 2048 tokens under the source LLaMA tokenizer, followed by benchmark deduplication and random sampling. The SFT++ uses the full corpus and is searched and reported separately.

A.2 Implementation of Source Models

For Qwen2.5-Math-7B, we use the public checkpoints released by LUFFY[2]. For the remaining settings, including Qwen2.5-Math-1.5B under the LUFFY configuration and LLaMA3.1-8B, the corresponding source weights are not publicly released. We therefore train the SFT and RLVR source models ourselves.

RLVR implementation. Our RLVR implementation is based on VeRL[3]. Following LUFFY (Yan et al., 2025), we use GRPO without the KL divergence term. We set the rollout batch size to 128, update batch size to 64, number of rollouts to 8, learning rate to 1×10^{-6}, and rollout temperature to 1.0. Training runs for 500 steps on one node with $8\times$ A800-80GB GPUs.

SFT implementation. We use LLaMA-Factory (Zheng et al., 2024) for full fine-tuning. The global batch size is 64. We use learning rate 5×10^{-5}, cosine scheduler with warmup ratio 0.1, DeepSpeed ZeRO-3, bfloat16 precision, and 3 training epochs.

A.3 System prompts

We specify the prompts used for source training and evaluation below. Within each source pair, the source and merged models use the same evaluation prompt template. Template-generated tokens are encoded without adding a second set of special tokens. The scores are reported with their corresponding evaluation configurations.

For Qwen-series models, the setup uses the same system prompt for both training and inference. The prompt asks the model to produce a structured reasoning process followed by a final answer.

> Your task is to follow a systematic, thorough reasoning process before providing the final solution. This involves analyzing, summarizing, exploring, reassessing, and refining your thought process through multiple iterations. Structure your response into two sections: Thought and Solution. In the Thought section, present your reasoning using the format: "`<think>\n` thoughts `</think>\n`". Each thought should include detailed analysis, brainstorming, verification, and refinement of ideas. After "`</think>\n`" in the Solution section, provide the final, logical, and accurate answer, clearly derived from the exploration in the Thought section. If applicable, include the answer in `\boxed{}` for closed-form results like multiple choices or mathematical solutions.
> **User:** This is the problem: `{QUESTION}`
> **Assistant:** `<think>`

For LLaMA3.1-8B, we use different prompts for SFT and RLVR training. For SFT training, we use the same prompt as the Qwen-series models. For RLVR training, we follow prior work (Yan et al., 2025), which finds that LLaMA models are less stable under complex system prompts. We therefore use a simpler Chain-of-Thought prompt that omits the `</think>` delimiter.

[2] https://huggingface.co/collections/Elliott/luffy-rl
[3] https://github.com/verl-project/verl

User: `{QUESTION}`
Answer: Let's think step by step.

During inference with LLaMA3.1-8B, we use a hybrid prompt to better align with both the SFT and RLVR source models. This prompt keeps the structured `<think>` format used by SFT while also including the simpler CoT prefix used during RLVR training.

Your task is to follow a systematic, thorough reasoning process before providing the final solution. This involves analyzing, summarizing, exploring, reassessing, and refining your thought process through multiple iterations. Structure your response into two sections: Thought and Solution. In the Thought section, present your reasoning using the format: "`<think>\n` thoughts `</think>\n`". Each thought should include detailed analysis, brainstorming, verification, and refinement of ideas. After "`</think>\n`" in the Solution section, provide the final, logical, and accurate answer, clearly derived from the exploration in the Thought section. If applicable, include the answer in `\boxed{}` for closed-form results like multiple choices or mathematical solutions.
User: This is the problem: `{QUESTION}`
Assistant: `<think>`
Answer: Let's think step by step.

A.4 Baseline Implementations

For baselines with public weights, including SFT, On-Policy RL, SFT+RL, and LUFFY, we download the released checkpoints and re-evaluate them under our evaluation setting. For model merging baselines, including TIES-Merging and DARE, we use the same source models as DoST and follow their official hyperparameter settings, with drop rate 10%.

B Algorithm

Algo. 1 summarizes the full procedure of DoST.

C Additional Experiments

This section provides additional experimental results on extension to alternative backbones for Sec. 5.5 in C.1 and C.2, and additional analyses for Sec. 2 in C.3 and C.4. Unless otherwise stated, all results are obtained with the default backbone Qwen2.5-Math-7B.

C.1 Extension to Alternative backbones

Setup. To test generalization across model scales and capabilities, we extend DoST to Qwen2.5-Math-1.5B (Yang et al., 2024a) and LLaMA3.1-8B (Grattafiori et al., 2024). For Qwen2.5-Math-1.5B, we employ training configurations from LUFFY (Yan et al., 2025) using SFT and On-Policy RL checkpoints trained on OpenR1-Math-220k. For LLaMA3.1-8B, we also follow the LUFFY training configuration. Since LLaMA3.1-8B is a weaker backbone, prior work filters out hard samples with generation length greater than 2048 tokens to stabilize training. We additionally train an SFT++ checkpoint that retains these harder samples, providing a stronger and more complementary SFT source model for DoST.

Results on Qwen2.5-Math-1.5B. Tab. 5 shows that DoST consistently improves over its source models across Qwen2.5-Math-1.5B settings. DoST (SFT + On-Policy RL) reaches 37.8, outperforming both source models and LUFFY. These results show that DoST remains effective at a smaller model scale.

Results on LLaMA3.1-8B. The LLaMA3.1-8B results show that the quality and complementarity of source models are important for DoST. When using the standard SFT checkpoint, DoST (SFT +

Algorithm 1 Procedure of Decoupled Synthesis (DoST)

Input: Base model θ_{base}, independently trained source models θ_{SFT} and θ_{RLVR}, source-training corpus $\mathcal{D}_{\text{train}}$, verifier V.
Hyperparameters: Retention fraction $p = 0.3$, search set size $n = 256$, responses per problem $R = 4$, initial random trials $T_0 = 5$, total trials $T = 100$.
Output: Final merged model θ^*.

Stage 1: Selective Task-Vector Sparsification
Extract $\tau_{\text{SFT}} \leftarrow \theta_{\text{SFT}} - \theta_{\text{base}}$ and $\tau_{\text{RLVR}} \leftarrow \theta_{\text{RLVR}} - \theta_{\text{base}}$.
for each vector $\tau \in \{\tau_{\text{SFT}}, \tau_{\text{RLVR}}\}$ **do**
  Compute mask M using Eq. (1).
  Rescale $\tau \odot M$ to its source's original global norm using Eq. (2).
end for

Stage 2: Bayesian Coefficient Optimization
Randomly sample $\mathcal{D}_{\text{sel}} = \{(x_i, a_i)\}_{i=1}^{n} \subseteq \mathcal{D}_{\text{train}}$.
Initialize TPE with search space $\boldsymbol{\lambda} \in [0, 2]^2$.
for $t = 1$ to T **do**
  if $t \leq T_0$ **then**
    Sample a random coefficient pair $\boldsymbol{\lambda}^{(t)} \in [0, 2]^2$.
  else
    Propose $\boldsymbol{\lambda}^{(t)}$ using TPE.
  end if
  Construct $\theta^{(t)} = \theta_{\text{base}} + \lambda_{\text{SFT}}^{(t)} \tilde{\tau}_{\text{SFT}} + \lambda_{\text{RLVR}}^{(t)} \tilde{\tau}_{\text{RLVR}}$.
  Sample R responses $y_{i,1}^{(t)}, \ldots, y_{i,R}^{(t)}$ per search problem x_i.
  Compute $\widehat{A}^{(t)} = \frac{1}{nR} \sum_{i=1}^{n} \sum_{j=1}^{R} V(x_i, y_{i,j}^{(t)}, a_i)$.
  Update the optimizer with $(\boldsymbol{\lambda}^{(t)}, \widehat{A}^{(t)})$.
end for
Select $t^* \leftarrow \arg\max_{1 \leq t \leq T} \widehat{A}^{(t)}$.
return $\theta^* \leftarrow \theta^{(t^*)}$.

Table 5: Overall performance based on alternative backbones, including **Qwen2.5-Math-1.5B** and **LLaMA3.1-8B**. All models are evaluated under a unified setting.

Model	AIME 24	AIME 25	AIME 26	AMC	MATH500	Minerva	Olympiad	Average
Qwen2.5-Math-1.5B-base (Yang et al., 2024a)	7.0	4.1	4.8	28.0	36.4	8.5	20.7	15.6
Qwen2.5-Math-1.5B-Instruct (Yang et al., 2024a)	11.6	8.2	8.3	47.4	76.2	30.9	39.7	31.8
Qwen2.5-Math-1.5B: Training Configuration from LUFFY (Yan et al., 2025)								
SFT	17.6	20.1	15.1	51.5	79.4	32.4	43.3	37.1
On-Policy RL	11.9	6.9	9.2	44.4	74.6	28.3	39.3	30.6
LUFFY (Yan et al., 2025)	16.7	11.8	14.2	45.9	76.8	32.0	43.6	34.4
DoST (SFT + On-Policy RL)	19.9	18.9	14.0	54.5	80.6	29.8	47.1	**37.8**
LLaMA-3.1-8B: Training Configuration from LUFFY (Yan et al., 2025)								
LLaMA-3.1-8B-base (Grattafiori et al., 2024)	0.2	0.4	0.0	4.0	9.8	4.8	1.6	3.0
LLaMA-3.1-8B-Instruct (Grattafiori et al., 2024)	5.4	0.1	0.5	21.1	49.8	22.8	16.1	16.6
SFT	0.9	0.2	0.9	8.2	26.6	7.4	10.2	7.8
SFT++	9.9	15.1	12.0	44.1	65.8	12.5	36.4	28.0
On-Policy RL	0.1	0.2	0.0	7.6	21.8	14.7	6.5	7.3
LUFFY	1.9	0.1	0.0	13.5	39.0	15.1	9.6	11.3
DoST (SFT + On-Policy RL)	0.3	0.3	0.8	11.1	30.2	9.2	9.2	8.7
DoST (SFT++ + On-Policy RL)	10.6	16.3	11.1	44.2	64.0	15.4	39.0	**28.7**

On-Policy RL) reaches 8.7, outperforming both SFT and On-policy RL variants. Moreover, DoST (SFT++ + On-Policy RL) reaches 28.7, outperforming LUFFY by 17.4 points. This suggests that DoST benefits more when the source checkpoints encode genuinely complementary capabilities. When SFT contains knowledge from harder samples, DoST can synthesize this knowledge with RLVR reasoning even on a weaker backbone. This also supports the motivation for decoupled synthesis, since directly injecting harder samples into RLVR training can make training less stable.

C.2 Harder Samples on Weaker Backbones

For LLaMA3.1-8B, we further test whether training-based integration can directly benefit from harder samples, or whether decoupled synthesis is needed.

From the result presented in Tab. 8, we can observe that directly adding harder samples to LUFFY leads to near-zero performance in this setting. In contrast, DoST can use SFT++ as a stronger knowledge source and synthesize it with the RLVR checkpoint after training. This supports the advantage of decoupled synthesis on weaker backbones, where direct training-time integration can be unstable.

Table 6: Gradient conflict ratio between SFT and GRPO, averaged over 10-step windows.

Steps	**1–10**	**11–20**	**21–30**	**31–40**	**41–50**	**51–60**	**61–70**	**71–80**	**81–90**	**91–100**	**Avg.**
Mean Conflict Ratio	0.499	0.505	0.506	0.505	0.501	0.505	0.513	0.505	0.502	0.512	0.505

Table 7: Within-module parameter retention rates after global top-10% selection across Qwen2.5-Math-7B and LLaMA3.1-8B. Each rate uses the corresponding module's total parameter count as the denominator.

Module	**Qwen-7B SFT**	**Qwen-7B RLVR**	**LLaMA-8B SFT**	**LLaMA-8B RLVR**
Attention	12.30%	13.86%	10.07%	11.82%
Embedding	0.38%	0.59%	1.70%	1.02%
LM Head	4.38%	0.48%	11.69%	0.37%
LayerNorm	19.90%	14.61%	10.17%	0.07%
MLP	11.72%	11.25%	10.78%	11.83%

Table 8: LLaMA3.1-8B performance when harder long-generation samples with length > 2048 tokens are included. LUFFY and LUFFY* indicate the training with filtered and unfiltered data, respectively.

Model	**AIME 24**	**AIME 25**	**AIME 26**	**AMC**	**MATH500**	**Minerva**	**Olympiad**	**Average**
LUFFY	1.9	0.1	0.0	13.5	39.0	15.1	9.6	11.3
LUFFY*	0.0	0.0	0.0	0.0	0.2	0.1	0.0	0.0
DoST (SFT++ + On-Policy RL)	10.6	16.3	11.1	44.2	64.0	15.4	39.0	**28.7**

C.3 Gradient Conflict in Joint Training

To complement the task-vector sign interference in Finding 2, we measure the gradient conflict ratio between SFT and GRPO objectives during joint training.

The result is reported in Tab. 6. The conflict ratio remains close to 0.5 throughout training. This indicates that the directional conflict between SFT and GRPO objectives is persistent under this joint-training setup, rather than only a short-lived optimization artifact.

C.4 Cross-Architecture Module Analysis

To test whether the module-wise differences in Finding 3 generalize beyond Qwen2.5-Math-7B, we repeat the within-module retention analysis on LLaMA3.1-8B. The result is reported in Tab. 7.

Across both backbones, SFT and RLVR show different module-wise retention patterns. The exact modules with the largest differences vary by architecture, but the broader pattern remains consistent: the two post-training paradigms retain different fractions of parameters across modules. This supports the presence of heterogeneous module-wise update patterns across both backbones.

D Proofs and Extensions of the Theoretical Analysis

The proof follows the mechanism in the main text. We first define a single response policy used by both SFT and RLVR. Its demonstration likelihood varies along a direction in which correctness reward is locally flat. We derive the two source checkpoints from their training objectives, calculate their best individual scalings, and prove a strict composition gain. We then verify that the actual top-30% processor can preserve this gain while removing nonzero coordinates. The remaining results address perturbations, general deletion bounds, and coefficient selection.

D.1 A Shared Policy with a Locally Uninformative Reward

There are two groups of questions, H and E, with probabilities π and $1-\pi$, where $0 < \pi < 1$. Both training objectives use this same question distribution. Each question has three full outputs: a demonstrated correct answer y_1, another correct answer y_2, and an incorrect answer y_3. The verifier accepts exactly the first two. The output texts may depend on the question; their probabilities depend on its group.

Translate the base parameters to zero and write $\theta = (x, y, z) \in \mathbb{R}^{10}$, with $z \in \mathbb{R}^8$. The coordinates x, y control correctness on the two groups. The coordinates z control preferences among answers that receive the same verifier reward. Choose $0 < \varepsilon < 1/2$ and $0 < \tau < 1$, and put

$$v = \frac{\tau}{8}(8, 7, 6, 5, 4, 3, 2, 1).$$

Define a smooth bump and two success probabilities by

$$\rho(x) = \begin{cases} \exp\left(1 - \dfrac{1}{1 - 4(x-1)^2}\right), & |x-1| < 1/2, \\ 0, & |x-1| \geq 1/2, \end{cases} \qquad (9)$$
$$f(x) = \varepsilon + (1/2 - \varepsilon)\rho(x), \qquad g(y) = \tfrac{1}{2} + \tfrac{1}{4}\exp[-(y-1)^2].$$

The bump is C^∞, is zero on a neighborhood of $x = 0$, and has its unique maximum $\rho(1) = 1$. It increases toward 1 and decreases thereafter on its support. Thus $f(0) = \varepsilon$, $f(1) = 1/2$, $g(0) = 1/2 + 1/(4e)$, and $g(1) = 3/4$.

For a question from either group, assign the demonstrated answer probability

$$d(\theta) = \frac{f(x)}{2}\exp\left[-(x-1)^2 - y^2 - \|z - v\|^2\right]. \qquad (10)$$

The complete policy is

	$P_\theta(y_1)$	$P_\theta(y_2)$	$P_\theta(y_3)$
H	$d(\theta)$	$f(x) - d(\theta)$	$1 - f(x)$
E	$d(\theta)$	$g(y) - d(\theta)$	$1 - g(y)$.

(11)

All entries are strictly positive and each row sums to one: $0 < d \leq f/2 \leq 1/4$, $f \geq \varepsilon > 0$, and $1/2 < g \leq 3/4$. In particular, supervised likelihood is well-defined even when ε is very small. The expected correctness reward is

$$A(\theta) = \pi f(x) + (1-\pi)g(y). \qquad (12)$$

These definitions separate assumptions from consequences. The plateau, the two correctness coordinates, and the demonstration preferences are structural choices of the model. We next derive the source checkpoints and the accuracy comparison; neither the endpoints nor a favorable merge are imposed as additional premises.

D.2 Deriving the Source Checkpoints

The SFT target is y_1 for every question. Its expected negative log-likelihood is exactly

$$\mathcal{L}_S(\theta) = \log 2 - \log f(x) + (x-1)^2 + y^2 + \|z - v\|^2. \qquad (13)$$

Since f has its unique maximum at 1, the unique minimizer of this loss is $s = (1, 0, v)$. This point is also the endpoint of SFT gradient flow from the common base. Indeed,

$$\dot{x} = \frac{f'(x)}{f(x)} + 2(1-x), \qquad \dot{y} = -2y, \qquad \dot{z} = -2(z - v).$$

The first right-hand side is strictly positive for $x < 1$, strictly negative for $x > 1$, and zero at 1. From zero, x increases toward 1, while y stays zero and z approaches v. Thus the objective and its dynamics give the same source.

For RLVR, use expected-reward gradient flow on this same policy:

$$\dot{x} = \pi f'(x), \qquad \dot{y} = (1-\pi)g'(y), \qquad \dot{z} = 0.$$

Because f is constant near zero, uniqueness of the smooth differential equation keeps $x = 0$. The style coordinates remain zero. Since $g'(y) > 0$ for $y < 1$ and vanishes at 1, the remaining coordinate approaches 1 from below. The source updates from the zero base are therefore

$$s = (1, 0, v), \qquad r = (0, 1, 0). \tag{14}$$

The SFT source minimizes its training loss, and the RLVR source lies on a local maximum plateau of expected reward. In particular,

$$\nabla A(r) = 0, \qquad \partial_x \mathcal{L}_S(r) = -2. \tag{15}$$

The demonstration gradient can move the model across the plateau, where f subsequently increases. The reward gradient from the base never moves in that coordinate. The hard-group success probability ε can be arbitrarily small, without making the demonstrated-answer probability zero.

Relation to a group-relative update. The reward-gradient model also describes the mean direction of a restricted group-relative update. For one question, sample two independent responses, use binary correctness rewards and group advantages normalized by the population standard deviation without an additive stabilizer, and set the update to zero when rewards match. A mixed group then has advantages $+1$ and -1. Use one on-policy score-gradient update per group, hold advantages fixed when differentiating, and omit additional regularizers. If the success probability is p, a mixed group occurs with probability $2p(1-p)$. Conditional on this event, its mean update is

$$\tfrac{1}{2}\left(\mathbb{E}[\nabla \log P(Y) \mid Y \text{ correct}] - \mathbb{E}[\nabla \log P(Y) \mid Y \text{ incorrect}]\right) = \frac{\nabla p}{2p(1-p)}.$$

Multiplication by the event probability gives ∇p. Averaging questions gives ∇A. The result above concerns this expected direction, not every sampled trajectory: zero expected gradient does not imply that every realized update is zero, and stochastic exploration can leave a plateau.

D.3 Proof of the Composition Gain

Proof of Theorem 4.1. The source derivation gives Eq. (14). For coefficients a, b, their composition has accuracy

$$F(a, b) = A(as + br) = \pi f(a) + (1 - \pi) g(b),$$

because the style coordinates do not affect correctness. Each function has its global maximum at 1. Hence both optimal single-source coefficients are 1, with values

$$\begin{aligned} B_S &= \frac{\pi}{2} + (1-\pi)\left(\frac{1}{2} + \frac{1}{4e}\right), \\ B_R &= \pi\varepsilon + \frac{3(1-\pi)}{4}, \\ A_* := A(s+r) &= \frac{\pi}{2} + \frac{3(1-\pi)}{4}. \end{aligned} \tag{16}$$

Define the weighted group gains

$$\Delta_H = \pi(1/2 - \varepsilon), \qquad \Delta_E = \frac{1-\pi}{4}(1 - e^{-1}). \tag{17}$$

The RLVR source leaves gain Δ_H unrealized, and the SFT source leaves gain Δ_E unrealized. Therefore

$$A_* - B = A_* - \max\{B_S, B_R\} = \min\{\Delta_H, \Delta_E\} > 0.$$

The composition coefficients $(1, 1)$ belong to $[0, 2]^2$, completing the proof. □

The stronger source can have zero entering derivative. Take $\pi = 1/5$. Then

$$B_S = \frac{1}{2} + \frac{1}{5e}, \qquad B_R = \frac{3}{5} + \frac{\varepsilon}{5}, \qquad A_* = \frac{7}{10}.$$

For every allowed ε, $B_R > B_S$, yet $\partial_a F(0, 1) = \pi f'(0) = 0$. The gain $A_* - B = 1/10 - \varepsilon/5$ is strictly positive. This is why the main theorem compares the composed point directly instead of requiring a positive infinitesimal derivative at the stronger source. For other weights the SFT source may be stronger; then the RLVR entering derivative at $(1, 0)$ is $(1-\pi)/(2e) > 0$. The exact composition comparison covers both cases.

D.4 Processing and the Cost of Removing Coordinates

For general raw sources, let M_S, M_R be coordinate masks. Set

$$d_S = \gamma_S M_S s, \quad d_R = \gamma_R M_R r, \qquad \gamma_i = \frac{\|v_i\|}{\|M_i v_i\|} \geq 1 \quad (v_S = s,\ v_R = r),$$

when the retained vector is nonzero; otherwise set $\gamma_i = 1$ and $d_i = 0$. For a dense witness $\theta_w = \theta_0 + as + br$, the discarded update is

$$u = a(I - M_S)s + b(I - M_R)r.$$

The coefficients $(a/\gamma_S, b/\gamma_R)$ remain in $[0, 2]^2$ and realize $\theta_w - u$.

Proof of Proposition 4.2. The stated gradient condition is $\|\nabla A(x) - \nabla A(y)\| \leq L_A\|x - y\|$ for points on the deletion segment, with differentiability on a neighborhood of that segment. Writing $g_w = \nabla A(\theta_w)$ and integrating the first derivative gives

$$\begin{aligned} A(\theta_w - u) - A(\theta_w) &= -g_w^\top u - \int_0^1 [\nabla A(\theta_w - tu) - g_w]^\top u \, dt \\ &\geq -g_w^\top u - \tfrac{1}{2} L_A \|u\|^2. \end{aligned}$$

Use the feasible processed coefficients above, maximize over the processed family, and subtract the unchanged raw-source comparator B. □

D.4.1 Verifying the Condition for the Constructed Sources

The construction uses one ten-dimensional tensor, so the method's top-30% rule retains its three largest magnitudes, including threshold ties. The SFT source has largest entries $1, \tau, 7\tau/8$. Its other six nonzero style entries are strictly smaller and are deleted. Write $v_{\text{keep}} = (\tau, 7\tau/8, 0, \ldots, 0)$. The RLVR source's unit y coordinate is retained. The norm factors are

$$\gamma_S^2 = \frac{64 + 204\tau^2}{64 + 113\tau^2} > 1, \qquad \gamma_R = 1.$$

The processed coefficient pair $(1/\gamma_S, 1)$ realizes $(1, 1, v_{\text{keep}})$, with accuracy A_*. For the dense witness $s + r$, the deleted update is $(0, 0, v - v_{\text{keep}})$. Throughout its deletion segment, the accuracy gradient is zero; thus $L_A = 0$ is valid on that segment and $C_{\text{del}} = 0$. Proposition 4.2 therefore preserves the exact composition margin while the mask deletes nonzero entries.

The processed single-source axes still attain B_S, B_R: each can reach its correctness coordinate 1, and the source-specific function has no larger value elsewhere. The same gap therefore also exceeds the best individually scaled processed sources. This example establishes compatibility of the processor with complementary updates; it does not show that pruning improves on the best dense composition.

D.4.2 Directional Curvature and Coefficient Adjustment

When A is C^2, a sharper bound may use a symmetric Q satisfying

$$u^\top[-\nabla^2 A(\theta_w - tu)]u \leq u^\top Q u, \qquad 0 \leq t \leq 1.$$

Taylor's integral identity then gives

$$\max_{\lambda \in [0,2]^2} A(\theta_0 + \lambda_S d_S + \lambda_R d_R) - B \geq \Delta - g_w^\top u - \tfrac{1}{2} u^\top Q u. \tag{18}$$

Only curvature in the deletion direction is bounded. Taking $Q = L_A I$ recovers the main bound when the smoothness condition holds. With $D = [d_S, d_R]$, $\lambda^0 = (a/\gamma_S, b/\gamma_R)$, and a feasible coefficient adjustment $h \in [0, 2]^2 - \lambda^0$, the total displacement is $z(h) = -u + Dh$. If a common Q bounds curvature along all such displacements in a chosen compact set $\mathcal{H}$ containing zero, the same argument gives

$$\max_{\lambda \in [0,2]^2} A(\theta_0 + D\lambda) - B \geq \Delta + \max_{h \in \mathcal{H}} \left[g_w^\top z(h) - \tfrac{1}{2} z(h)^\top Q z(h) \right].$$

Likewise, applying the deletion bound at a dense coefficient optimum θ_D^* yields

$$V_{\text{processed}} - V_{\text{dense}} \geq -\nabla A(\theta_D^*)^\top u_D - \tfrac{1}{2} u_D^\top Q u_D.$$

A positive right-hand side certifies improvement over dense merging; the reference construction above establishes preservation rather than this stronger comparison.

D.5 Stability of the Composition Comparison

Exact coordinate separation is not required for the resulting accuracy gap to survive small errors. Let A_0 denote the reference accuracy in Eq. (12), and choose a global Lipschitz constant K for it. Such a constant exists because the derivatives of the smooth bump and the Gaussian are bounded. Suppose perturbed checkpoints satisfy

$$\|\widehat{s} - s\| \le \eta, \qquad \|\widehat{r} - r\| \le \eta, \qquad 0 \le \eta < 1/8.$$

Let $\widehat{A}$ be a continuous accuracy function with $|\widehat{A} - A_0| \le \zeta$ on the candidate sets considered below. This permits small functional interactions between coordinates. Let $\widehat{B}_{\mathrm{raw}}$ and $\widehat{B}_{\mathrm{processed}}$ be the best respective single-source values over $[0, 2]$, and put $\Delta = \min\{\Delta_H, \Delta_E\}$.

Proposition D.1 (A margin under checkpoint and accuracy perturbations)**.** *After the same top-30% rule and own-norm restoration, the processed family satisfies*

$$\widehat{V}_{\mathrm{processed}} - \widehat{B}_{\mathrm{raw}} \ge \Delta - 4K\eta - 2\zeta, \tag{19}$$

$$\widehat{V}_{\mathrm{processed}} - \widehat{B}_{\mathrm{processed}} \ge \Delta - 8K\eta - 2\zeta. \tag{20}$$

Proof. The SFT source's x coordinate is retained after perturbation. Its magnitude is at least $1 - \eta$, while all but two competing coordinates have magnitude at most $3\tau/4 + \eta < 1 - \eta$. The RLVR source's y coordinate is also retained, since its magnitude is at least $1 - \eta$ and all competitors have magnitude at most η. Mask ties do not affect these conclusions.

Undoing the two norm factors by feasible coefficients produces the witness

$$\widehat{\theta}_w = M_S\widehat{s} + M_R\widehat{r}.$$

Its two correctness coordinates are within Euclidean distance 2η of $(1, 1)$: each masked source retains its unit reference coordinate, and its remaining correctness-coordinate error has norm at most η. Since A_0 ignores style coordinates, $\widehat{A}(\widehat{\theta}_w) \ge A_* - 2K\eta - \zeta$. An actual raw-axis point with coefficient in $[0, 2]$ is within 2η of its corresponding reference-axis point, so $\widehat{B}_{\mathrm{raw}} \le B + 2K\eta + \zeta$. Subtraction proves the first inequality.

For processed axes, both retained norms are at least $1 - \eta$. Since $\|s\| < \sqrt{5}$ and $\|r\| = 1$, the norm factors obey

$$\widehat{\gamma}_i \le \frac{\sqrt{5} + \eta}{1 - \eta} < 3.$$

A processed single-source candidate has its two correctness coordinates within $2\widehat{\gamma}_i\eta \le 6\eta$ of a point on the corresponding reference axis. Although its effective reference coefficient can exceed 2, the global maximum along that axis remains B_S or B_R, respectively. Hence $\widehat{B}_{\mathrm{processed}} \le B + 6K\eta + \zeta$. Subtracting from the witness bound proves the second inequality. □

These are stability statements about checkpoints and accuracy values. They do not assert that arbitrary changes to the training objective preserve the exact plateau dynamics or automatically yield the assumed checkpoint error. They also do not infer derivative bounds from a value-error bound.

D.6 Local Gain as a Special Case

For arbitrary source pairs, a positive entering derivative remains a useful sufficient condition. If F is differentiable near a best single-source point $(0, b_*)$, with $b_* \in (0, 2)$ and $q = \partial_a F(0, b_*) > 0$, then

$$F(t, b_*) - B = qt + o(t) > 0$$

for sufficiently small positive t. At an interior anchor $\theta_* = \theta_0 + b_* r$, differentiability of A gives $\nabla A(\theta_*)^\top r = 0$. Thus

$$q = \nabla A(\theta_*)^\top \left(s - \frac{\langle s, r \rangle}{\|r\|^2} r \right)$$

when $r \neq 0$. Parameter norms and signs alone do not establish this alignment. The interior condition excludes a local gain obtained merely by extending a bounded scaling range. The reward-plateau example with a stronger RLVR source has $q = 0$, so its improvement is certified by the full composition calculation instead.

D.7 Top-k% Retention Can Lose Functional Complementarity

This example uses exactly one ten-coordinate tensor and the method's top-30% mask. Fix $K > 1/2$, set $\theta_0 = 0$, and let

$$s = (K, 2K, 3K, \tfrac{1}{2}, 0, \ldots, 0), \qquad r = (-K, -2K, -3K, \tfrac{1}{2}, 0, \ldots, 0).$$

Define the globally smooth bounded accuracy

$$A(\theta) = \tfrac{1}{4} + \tfrac{1}{2} \exp\left[-\frac{\theta_1^2 + \theta_2^2 + \theta_3^2}{28K^2} - (\theta_4 - 1)^2 \right]. \tag{21}$$

It is realizable by a two-output policy assigning probability $A(\theta)$ to a correct output and $1 - A(\theta)$ to an incorrect one. Both raw sources improve over the base: $A(s) = A(r) = 1/4 + e^{-3/4}/2 > A(0) = 1/4 + e^{-1}/2$. Along either source ray with scale t, the negative exponent is

$$t^2/2 + (t/2 - 1)^2 = \tfrac{3}{4}t^2 - t + 1.$$

It is minimized at $t = 2/3$, yielding $B = 1/4 + e^{-2/3}/2$. The dense merge $s + r$ instead attains the global maximum $3/4$.

The three largest absolute entries of each update are $K, 2K, 3K$. Top-30% pruning removes the fourth coordinate, and every masked or rescaled candidate has $\theta_4 = 0$. Its best accuracy is exactly $A(0) = 1/4 + e^{-1}/2 < B$, attained by cancelling the retained coordinates or setting both coefficients to zero. Yet each mask retains the energy fraction

$$\frac{14K^2}{14K^2 + 1/4} \longrightarrow 1 \quad \text{as } K \longrightarrow \infty.$$

The three opposite-sign large coordinates cancel beneficially in the dense merge. Thus neither a high retained-energy fraction nor treating opposite signs as necessarily harmful establishes an accuracy-retention guarantee. This does not show that sparsification harms the experimental checkpoints. The removed absolute contribution remains $1/2$ per source, consistent with the directional perturbation bounds.

D.8 Coefficient Search and Accuracy-Based Selection

The preceding results establish that a favorable coefficient pair exists. Let A now denote target expected accuracy, let A_U be expected accuracy on the fixed search questions, and let B_{target} be the chosen target comparator. Suppose a witness satisfies $A(\theta_*) - B_{\text{target}} \geq G > 0$. Over the candidate family, assume

$$\sup_\theta |A_U(\theta) - A(\theta)| \leq g, \qquad |\widehat{A}_t - A_U(\theta_t)| \leq e_s \quad \text{for every evaluated candidate.}$$

The first bound must be justified for the selected questions even when they reuse training data; it does not follow from a high training score. The second concerns score estimation, including adaptive candidate choices.

If one evaluated candidate has target accuracy at least $A(\theta_*) - \eta_{\text{search}}$, the highest-scoring candidate $\widehat{\theta}$ obeys

$$A(\widehat{\theta}) - B_{\text{target}} \geq G - \eta_{\text{search}} - 2g - 2e_s. \tag{22}$$

Indeed, highest observed score loses at most $2e_s$ in expected search accuracy relative to that candidate. Transferring the two candidate values between search and target distributions costs at most $2g$, proving the inequality.

Continuity supplies a coefficient neighborhood V around the reference witness $(1/\gamma_S, 1)$ whose candidates have target accuracy within any specified positive tolerance. To state sufficient coverage, let $\mathcal{H}_0$ contain all information available before search and $\mathcal{H}_{t-1}$ the full history before evaluation t. Assume

$$\Pr(\lambda_t \in V \mid \mathcal{H}_{t-1}) \geq \rho_t \quad \text{almost surely}, \qquad \sum_t \rho_t = \infty,$$

where $\rho_t \in [0, 1]$ is fixed conditional on $\mathcal{H}_0$. Iterating conditional miss probabilities gives

$$\Pr(\text{no visit to } V \text{ by } N \mid \mathcal{H}_0) \leq \prod_{t=1}^{N}(1 - \rho_t) \leq \exp\left(-\sum_{t=1}^{N} \rho_t\right) \longrightarrow 0.$$

An initial sequence with $\rho_t = 0$ is allowed. Thus a sufficiently large number of evaluations provides coverage under persistent exploration. Score fidelity and transfer to the target distribution remain separate requirements. The argument does not establish these assumptions for every coefficient-search implementation.