SDC-LoRA: Singular-Subspace Drift Controlled LoRA to Mitigate Knowledge Forgetting

Geyuan Zhang[1,2], Xiaofei Zhou[1,2]*, Shihao Liu[1,2], Jingyuan Tian[1,2], Jizheng Ma[1,2]
[1]Institute of Information Engineering, Chinese Academy of Sciences,
[2]School of Cyber Security, University of Chinese Academy of Sciences
{zhanggeyuan, zhouxiaofei, liushihao, tianjingyuan, majizheng}@iie.ac.cn

Abstract

Knowledge forgetting is a central challenge when adapting LLMs to new tasks. Prior studies indicate that pretrained knowledge is concentrated in the principal singular subspace of pretrained weight W_0; so recent Low-Rank Adaptation (LoRA) variants initialize LoRA in the minor subspace to steer early updates away from principal directions and mitigate forgetting. However, we observe that during fine-tuning, the update direction progressively shifts from the minor to the principal subspace, which is called as *Singular-subspace Drift (SD)*, thereby allocating more energy to the directions that carry pretrained knowledge and leaving a persistent risk of forgetting. To address this issue, we propose **S**ingular-subspace **D**rift **C**ontrolled **LoRA** (SDC-LoRA), which constrains the growth of update energy in the principal singular subspace of W_0 and thus mitigate SD. SDC-LoRA proposes *Principal Subspace Energy-Controlled Learning*, using *Spectral Calibration* factor γ_{sc} to selectively downscale gradients along the principal singular subspace of W_0 while keeping minor-subspace updates unchanged. Across extensive experiments with LLaMA-3.1-8B-Instruct and Qwen2.5-7B-Chat on MetaMathQA and CodeFeedback, SDC-LoRA mitigates forgetting on MMLU, TruthfulQA, and HellaSwag while matching or improving GSM8K and HumanEval, offering a practical route to adapt LLMs without sacrificing prior knowledge.

1 Introduction

Fine-tuning LLMs translates broad pre-training into task-specific ability (Touvron et al., 2023a; Yang et al., 2024a; Touvron et al., 2023b), but full fine-tuning (FFT) is increasingly prohibitive in memory, compute, and storage (Qiu et al., 2020). Parameter-efficient fine-tuning (PEFT) mitigates this by freezing most weights and training a small subset (Houlsby et al., 2019; Zaken et al., 2022; Lester et al., 2021; Li and Liang, 2021). Among PEFT methods, LoRA (Hu et al., 2022) reparameterizes the update weight matrix with low-rank factors and is widely adopted, with many variants further reducing trainable parameters or improving effectiveness. Both FFT and PEFT still suffer from *knowledge forgetting* (McCloskey and Cohen, 1989; Kirkpatrick et al., 2017; Goodfellow et al., 2013): as the fine-tuned model improves on the target task, its performance on general-purpose evaluations often drops. This trade-off motivates methods that adapt LLMs while explicitly preserving their pre-training abilities.

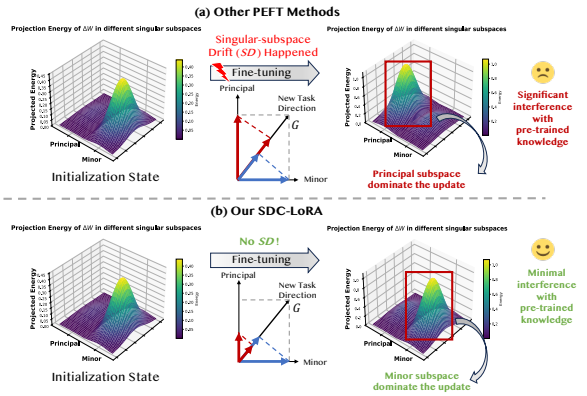

Figure 1: Illustration of *SD* and its mitigation by SDC-LoRA: in standard PEFT the update energy of ΔW moves from the minor to the principal subspace, while SDC-LoRA keeps it minor-subspace dominant.

Recent works refine LoRA initialization. PiSSA (Meng et al., 2024) initializes from the SVD of the pretrained weight W_0 using leading singular vectors/values to speed convergence and improve performance; LoRA-GA (Wang et al., 2024) and LoRA-One (Zhang et al., 2025) initialize from the SVD of a one-step full-parameter gradient so that LoRA better approximates full fine-tuning. But such principal- or gradient-aligned initializations steer the update ΔW toward the principal singular subspace of W_0. Since pretrained knowledge con-

*Corresponding author

Findings of the Association for Computational Linguistics: ACL 2026, pages 24116–24132

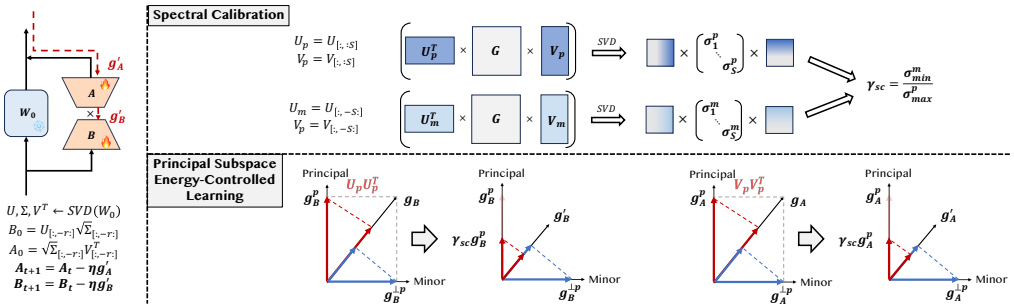

Figure 2: Illustration of SDC-LoRA. The pretrained weight W_0 is decomposed by SVD into principal (U_p, V_p) and minor (U_m, V_m) subspaces, LoRA weights (B_0, A_0) are initialized in the minor subspace, and during fine-tuning the gradients (g_A, g_B) are split into principal and minor components, with only the principal part scaled by γ_{sc} which abtained by spectral calibration using sampled full gradient G.

centrates in this subspace (Hajimolahoseini et al., 2021; Sharma et al., 2023; Li et al., 2024), sustained growth of ΔW there tends to overwrite existing representations and aggravate knowledge forgetting (Wang et al., 2025a). To mitigate this risk, CorDA (Yang et al., 2024b) uses activation covariance from world-knowledge samples to guide an SVD of W_0 and fine-tune components least tied to that knowledge, but it depends on such data and is sensitive to its sampling quality. MiLoRA (Wang et al., 2025a) initializes in the minor subspace to keep early updates away from principal directions without and reduce knowledge forgetting in early stage.

However, we find that both initialization strategy above can not keep the LoRA update away from principal directions as fine-tuning proceeds. During fine-tuning, in the SVD-aligned basis of W_0, the update ΔW allocates progressively more energy to the principal singular subspace, moving from minor- to principal-dominant directions (Fig. 1). We term this pattern *Singular-subspace Drift (SD)*. The existence of the *SD* phenomenon would lead to sustained expansion of ΔW along the principal singular directions of W_0, progressively overwriting the subspace that encodes pretrained knowledge and thereby aggravating knowledge forgetting, which empirically appears as a consistent drop on general-ability benchmarks even when the target task continues to improve. (Details for Section 3.2).

To address this issue, we propose **S**ingular-subspace **D**rift **C**ontrolled **LoRA** (SDC-LoRA), which mitigates knowledge forgetting by keeping incremental updates preferentially in the minor singular subspace of W_0 (Fig. 1). Our analysis attributes *SD* to an imbalance in gradient projections onto W_0's singular subspaces: a much stronger principal component drives update energy to drift from the minor toward the principal subspace. Building on this observation, SDC-LoRA implements *Principal Subspace Energy-Controlled Learning*. At each step, it decomposes the LoRA gradient into a component aligned with the principal singular subspace of W_0 and a complementary component. It then rescales only the principal-aligned part by a *Spectral Calibration* factor γ_{sc} derived from the singular-value spectra of the restricted gradients; Theorem 1 shows that this choice prevents growth of the principal-to-minor projection energy ratio and thus alleviates SD. Our SDC-LoRA effectively slows the accumulation of update energy in the principal subspace and reducing forgetting while preserving target-task learning. We fine-tune Llama-3.1-8B-Instruct and Qwen2.5-7B-Chat on MetaMathQA, and then evaluate on GSM8K (+0.12/+0.36) for math reasoning ability and MMLU (+2.14/+1.87), TruthfulQA (+1.29/+2.16), and HellaSwag (+0.76/+1.44) for the retention of pretrained knowledge. We also fine-tune Llama-3.1-8B-Instruct and Qwen2.5-7B-Chat on CodeFeedback, and then evaluate on HumanEval (+1.09/+0.86) for code generation ability and MMLU (+1.52/+2.69), TruthfulQA (+1.14/+1.96), and HellaSwag (+2.77/+1.02) for the retention of pretrained knowledge.

2 Related Work

Parameter Efficient Fine-Tuning. As LLMs scale, full fine-tuning (FFT) becomes prohibitively expensive (Qiu et al., 2020). Parameter-efficient fine-tuning (PEFT) reduces training and storage cost by freezing most weights. Adapter-style methods insert bottleneck modules and train only these, but change the architecture and add inference latency (Houlsby et al., 2019; Pfeiffer et al., 2021b,a). Aghajanyan et al. (2021) further observed that task-specific updates tend to lie in a low-dimensional

subspace (the intrinsic dimension). Building on this, LoRA (Hu et al., 2022) reparameterizes the weight update as the product of two low-rank matrices, which preserves the inference graph and markedly lowers training cost. Extensions refine LoRA via (i) adaptive/dynamic rank (Zhang et al., 2023; Valipour et al., 2023), (ii) improved initialization (Meng et al., 2024; Wang et al., 2025a, 2024; Zhang et al., 2025), and (iii) alternative factorizations (Liu et al., 2024a; Yuan et al., 2024; Liu et al., 2024b; Kopiczko et al., 2023; Gao et al., 2024). Despite these advances, both FFT and PEFT still suffer post-fine-tuning degradation of pretraining knowledge, i.e., *knowledge forgetting* (Biderman et al., 2024).

Knowledge Forgetting. Knowledge forgetting (McCloskey and Cohen, 1989; Kirkpatrick et al., 2017; Goodfellow et al., 2013) in LLMs denotes the degradation of general-knowledge performance when adapting a pretrained model to new data. Mitigation strategies in classical deep-learning literature (Hou et al., 2019; Li et al., 2023; Yang et al., 2023; Yan et al., 2021) often fail to scale due to model size and compute demands. PEFT-based approaches include CorDA (Yang et al., 2024b) (task-aware, context-guided adaptation), LoRA-Null (Tang et al., 2025) (null-space updates to limit interference), and MiLoRA (Wang et al., 2025a) (minor-subspace initialization to bias updates away from principal directions), yet none fully resolves forgetting in practice. We introduce a targeted method that mitigates knowledge forgetting without increasing inference cost, while maintaining (and sometimes modestly improving) target-task performance, thereby achieving a better balance between target accuracy and retention of pretraining knowledge.

3 Singular-subspace Drift in PEFT

3.1 Setup

Consider a pre-trained weight matrix $W_0 \in \mathbb{R}^{d\times k}$ and a low-rank PEFT update $\Delta W_t = B_t A_t$ with $B_t \in \mathbb{R}^{d\times r}$ and $A_t \in \mathbb{R}^{r\times k}$. The adapted weight at timestep t is then given by:

$$W_t = W_0 + \Delta W_t = W_0 + B_t A_t. \tag{1}$$

Let the SVD of W_0 be

$$W_0 = U\,\Sigma\,V^\top. \tag{2}$$

Here, $U \in \mathbb{R}^{d\times d}$ and $V \in \mathbb{R}^{k\times k}$ are orthogonal, and $\Sigma \in \mathbb{R}^{d\times k}$ is rectangular diagonal with non-negative, descending entries.

Definition 1 (Projection energy and energy ratio)**.** *For the principal singular subspace of W_0 spanned by $U_p := U_{[:,:S]} \in \mathbb{R}^{d\times S}$ and $V_p := V_{[:,:S]} \in \mathbb{R}^{k\times S}$, and the minor subspace spanned by $U_m := U_{[:,-S:]} \in \mathbb{R}^{d\times S}$ and $V_m := V_{[:,-S:]} \in \mathbb{R}^{k\times S}$, we define the double-sided projection energy of ΔW_t onto these two singular subspaces as follows:*

$$E_{p,t} := \left\|U_p U_p^\top\, \Delta W_t\, V_p V_p^\top\right\|_F^2, \tag{3}$$

$$E_{m,t} := \left\|U_m U_m^\top\, \Delta W_t\, V_m V_m^\top\right\|_F^2, \tag{4}$$

and the contrastive energy ratio:

$$\mathcal{R}_t := \frac{E_{p,t} - E_{m,t}}{E_{p,t} + E_{m,t}} \in [-1, 1]. \tag{5}$$

Specifically, $E_{p,t}$ quantifies the energy of the component of ΔW_t aligned with the principal singular subspace of W_0. A higher $E_{p,t}$ indicates greater modification of directions most strongly associated with pretraining knowledge. In contrast, $E_{m,t}$ measures how much of the update ΔW_t is injected along the minor singular directions of W_0, reflecting how much of the update is allocated to directions that are less critical to the preservation of pre-trained knowledge. Consequently, the contrastive energy ratio $\mathcal{R}_t$ serves as a quantitative indicator of whether fine-tuning updates predominantly align with the principal singular subspace (positive $\mathcal{R}_t > 0$) or remain concentrated in the safer minor singular subspace (negative $\mathcal{R}_t < 0$).

3.2 Projection Energy Analysis

To examine how the contrastive energy ratio of the LoRA update ΔW_t (measured between the principal and minor singular subspaces of W_0) evolves during fine-tuning, we conduct controlled experiments. Specifically, we fine-tune a LLaMA-3.1-8B-Instruct model on the MetaMath dataset using four parameter-efficient methods: LoRA, MiLoRA, PiSSA, and LoRA-One. At multiple training steps, we log the contrast $\mathcal{R}_t$ of ΔW_t (computed per layer and then averaged across layers; Fig. 3(a)). In addition, we evaluate general-capability performance at matched steps on the MMLU, TruthfulQA, and HellaSwag benchmarks (Fig. 3 (b)–(d)).

From Fig. 3(a), we observe the following phenomena. PiSSA starts with the contrastive ratio $\mathcal{R}_0 = 1$ and remains at a high, nearly steady

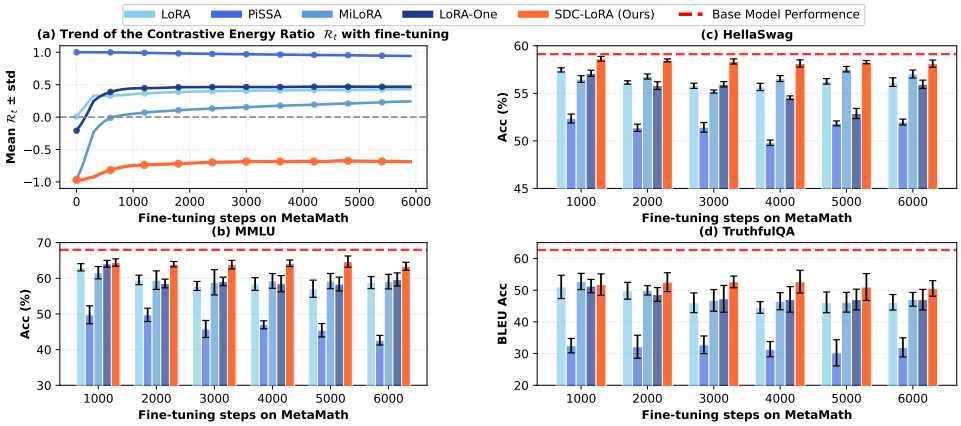

Figure 3: **Rising principal share during training.** Principal–vs.–minor energy ratio (Section 3.2) across steps for several PEFT variants. Methods initialized or aligned with principal directions show faster ratio growth; minor-initialization slows but does not stop the growth. Our SDC-LoRA suppresses this rise.

level throughout training, indicating that the energy of ΔW projected onto the principal singular subspace dominates the minor projection, i.e., updates persistently align with the principal directions of W_0. LoRA begins with the contrast ratio $\mathcal{R}_0 = 0$ and LoRA-One begins with $\mathcal{R}_0 \approx 0$ (rough parity between principal and minor projections) but then exhibits a monotonic increase as training proceeds, rising roughly from 0 to 0.5, which reflects the update direction of ΔW in W_0 singular subspace a progressive tilt toward the principal singular subspace. A similar upward trend appears for MiLoRA, and is even more pronounced: it initializes with contrast near $\mathcal{R}_0 = -1$ (The update direction in the singular subspace of W_0 is completely concentrated in the minor singular subspace) and then increases substantially, reaching about 0.2. **We refer to this stepwise increase of the principal/minor singular subspace contrast, i.e., the migration of projection energy from the minor to the principal subspace, as *Singular-subspace Drift (SD)*.**

Empirical motivation from Fig. 3. Fig. 3 reveals a consistent pattern that motivates our design. *First*, larger contrastive energy ratio $\mathcal{R}_t$ correlate with worse performance on general capability benchmarks after fine-tuning in our setting. *Second*, principal–aligned initialization (PiSSA) maintains a consistently high $\mathcal{R}_t$ with minimal variation, and correspondingly exhibits the strongest performance degradation. *Third*, non–principal or non–aligned initializations (LoRA, LoRA-One, MiLoRA) typically show a rising $\mathcal{R}_t$ over steps, indicating a singular subspace shift toward the principal directions, with MiLoRA displaying the most pronounced growth. These observations yield a simple insight: $\mathcal{R}_t$ directly tracks principal-subspace injection. When $\mathcal{R}_t$ remains high (positive), updates allocate more energy to the principal subspace, which correlates with stronger forgetting of pretraining knowledge. Conversely, when $\mathcal{R}_t$ stays near -1, principal-subspace injection is limited, allowing the model to preserve pretraining knowledge while still learning the target task. Building on this insight, we introduce SDC-LoRA in Section 4.1, which explicitly regulates $\mathcal{R}_t$ by scaling only the update component aligned with the principal singular subspace during training.

4 Method

4.1 Spectral Calibration

Minor-subspace initialization. To avoid immediate interference with pretraining knowledge encoded in the principal subspace of W_0, we initialize the LoRA factors so that the initial update lies entirely in the minor subspace (Wang et al., 2025a). Let $W_0 = U\,\Sigma\,V^\top$ and we initializes SDC-LoRA as follows:

$$B_0 = U_{[:,-r:]}\sqrt{\Sigma}_{[:,-r:]}, A_0 = \sqrt{\Sigma}_{[:,-r:]}\,V^\top_{[:,-r:]}, \tag{6}$$

where $B_0 \in \mathbb{R}^{d\times r}$ and $A_0 \in \mathbb{R}^{r\times k}$. This initialization guarantees that the update ΔW starts entirely in the minor subspace, orthogonal to the principal directions where pre-trained knowledge resides (which ensures $U_p U_p^\top\,\Delta W_0\,V_p V_p^\top = 0$). However, as mentioned before (Section 3.2), as training progresses, we observe *SD*. According to Wang et al. (2024) and Wang et al. (2025b), we can use full parameter gradient to describe the update process

of A and B:

$$A_{t+1} \approx A_t - \eta\, B_t^\top G, \qquad B_{t+1} \approx B_t - \eta\, G A_t^\top, \tag{7}$$

where $G := \nabla_W \mathcal{L}(W)\big|_{W=W_0}$ is a one-step full-parameter gradient. Then we project the LoRA parameters onto the principal subspace of W_0: $\widetilde{A}_{p,t} := A_t V_p \in \mathbb{R}^{r\times S}$ and $\widetilde{B}_{p,t} := U_p^\top B_t \in \mathbb{R}^{S\times r}$. Therefore the parameter update in principal subspace of W_0 becomes:

$$\begin{aligned} \widetilde{A}_{p,t+1} &\approx \widetilde{A}_{p,t} - \eta\, \widetilde{B}_{p,t}^\top H_p, \\ \widetilde{B}_{p,t+1} &\approx \widetilde{B}_{p,t} - \eta\, H_p\, \widetilde{A}_{p,t}^\top, \\ &\text{where } H_p = U_p^\top G V_p \in \mathbb{R}^{S\times S}. \end{aligned} \tag{8}$$

Similarly we can describe the parameter update in minor subspace of W_0 by project the LoRA parameters onto the minor subspace of W_0: $\widetilde{A}_{m,t} := A_t V_m \in \mathbb{R}^{r\times S}$ and $\widetilde{B}_{m,t} := U_m^\top B_t \in \mathbb{R}^{S\times r}$:

$$\begin{aligned} \widetilde{A}_{m,t+1} &\approx \widetilde{A}_{m,t} - \eta\, \widetilde{B}_{m,t}^\top H_m, \\ \widetilde{B}_{m,t+1} &\approx \widetilde{B}_{m,t} - \eta\, H_m\, \widetilde{A}_{m,t}^\top, \\ &\text{where } H_m = U_m^\top G V_m \in \mathbb{R}^{S\times S}. \end{aligned} \tag{9}$$

The proof of Eq. (8) and Eq. (9) see Appendix A. From Eq. (8)and (9), we can find that the updates of A and B restricted in different subspace are assosiated with the projected gradient H_p and H_m. **Therefore, the *SD* is intuitively because the gradient signal in the principle subspace is significantly stronger than that in the minor subspaces.** Building on this insight, we introduce a spectrally calibrated factor that reduces the principal-subspace component of the update, slowing drift and forgetting while preserving convergence on the new task.

Theorem 1 (Spectral Calibration Factor)**.** *We perform SVD to $H_m = P_m \Sigma_m Q_m^\top$ and $H_p = P_p \Sigma_p Q_p^\top$, where $\Sigma_m = \mathrm{diag}(\sigma_1^m \geq \cdots \geq \sigma_S^m \geq 0)$ and $\Sigma_p = \mathrm{diag}(\sigma_1^p \geq \cdots \geq \sigma_S^p \geq 0)$. Then let $\sigma_{\max}^p := \max_i \sigma_i^p$ and $\sigma_{\min}^m := \min_j \sigma_j^m$, and set*

$$\gamma_{\mathrm{sc}} := \frac{\sigma_{\min}^m}{\sigma_{\max}^p} \in (0, 1].$$

Under the linearized update dynamics, applying γ_{sc} to the principal block update constrains its aggregate energy growth rate to be no greater than that of the minor subspace, thereby theoretically preventing the SD.

Proof. See Appendix B.

Remark. While Theorem 1 guarantees drift suppression in the deterministic linearized regime, in practical training, $\mathcal{R}_t$ may exhibit local fluctuations due to higher-order dynamics.

4.2 Principal Subspace Energy-Controlled Learning

At step t, let g_{A_t} and g_{B_t} be the raw updates for A_t and B_t. Project onto the principal subspace of $W_0 = U\Sigma V^\top$ (with $U_p \in \mathbb{R}^{d\times S}$, $V_p \in \mathbb{R}^{k\times S}$):

$$g_{A_t}^p = g_{A_t} V_p V_p^\top, \qquad g_{B_t}^p = U_p U_p^\top g_{B_t}, \tag{10}$$

and define residuals $g_{A_t}^{\perp p} = g_{A_t} - g_{A_t}^p$, $g_{B_t}^{\perp p} = g_{B_t} - g_{B_t}^p$. Apply conservative scaling to the principal components:

$$g'_{A_t} = g_{A_t}^{\perp p} + \gamma_{\mathrm{sc}}\, g_{A_t}^p, \; g'_{B_t} = g_{B_t}^{\perp p} + \gamma_{\mathrm{sc}}\, g_{B_t}^p. \tag{11}$$

At each step, replace (g_{A_t}, g_{B_t}) by (g'_{A_t}, g'_{B_t}) to constrain principal-subspace injection and mitigate SD (see Algorithm 1).

This SDC scaling is equivalent to a symmetric positive-definite preconditioner $M_\gamma = U_m U_m^\top + \gamma_{\mathrm{sc}} U_p U_p^\top$ with $\gamma_{\mathrm{sc}} \in (0, 1]$: each step remains a descent step, merely shortening the principal-direction step by γ_{sc} rather than flipping direction, which at most slows the rate by a constant factor. As a result, it curbs harmful changes to knowledge-bearing directions while preserving target-task learning. Please refer to Appendix D for specific proof of convergence.

Algorithm 1 SDC-LoRA

Require: $W_0 \in \mathbb{R}^{d\times k}$, rank r, subspace suppression dimension S, learning rate η, optimizer, sample batch $\mathcal{D}_s$.
Bases: $W_0 = U\Sigma V^\top$; $U_p = U_{[:,:S]}$, $V_p = V_{[:,:S]}$; $U_m = U_{[:,-S:]}$, $V_m = V_{[:,-S:]}$.
Minor init: set $B_0 = U_{[:,-r:]}\sqrt{\Sigma}_{[:,-r:]}$, $A_0 = \sqrt{\Sigma}_{[:,-r:]} V_{[:,-r:]}^\top$, and $\Delta W_0 = B_0 A_0$.
Compute γ: compute full-grad G on $\mathcal{D}_s$;
  $H_p = U_p^\top G V_p$, $H_m = U_m^\top G V_m$; get $\sigma_{\max}^p$, $\sigma_{\min}^m$;
  $\gamma_{\mathrm{sc}} \leftarrow \min(1, \sigma_{\min}^m / \sigma_{\max}^p)$.
for each batch $\mathcal{B}$ **do**
  Obtain per-step increments g_A and g_B from optimizer.
  $g_A^p = g_A V_p V_p^\top$, $g'_A = g_A - (1-\gamma_{\mathrm{sc}})\, g_A^p$.
  $g_B^p = U_p U_p^\top g_B$, $g'_B = g_B - (1-\gamma_{\mathrm{sc}})\, g_B^p$.
  $A_{t+1} = A_t - \eta g'_A$, $B_{t+1} = B_t - \eta g'_B$.
end for

5 Experiments

In this section, we conduct experiments to compare SDC-LoRA with typical LoRA based algorithms across math reasoning and code generation

Table 1: Comparison of our SDC-LoRA with several baselines on Math Reasoning task. Results are reported as mean with standard deviations over 5 runs (higher is better).

Model	Method	# Params (%)	GSM8K	MMLU	TruthfulQA	HellaSwag
LLaMA3.1-8B	LoRA (Hu et al., 2022)	33M (0.40%)	$78.76_{\pm 0.6}$	$58.80_{\pm 1.5}$	$46.14_{\pm 2.1}$	$56.18_{\pm 0.5}$
	MiLoRA (Wang et al., 2025a)	33M (0.40%)	$78.95_{\pm 0.7}$	$59.07_{\pm 2.0}$	$47.12_{\pm 2.2}$	$57.03_{\pm 0.4}$
	PiSSA (Meng et al., 2024)	33M (0.40%)	$79.20_{\pm 0.4}$	$42.64_{\pm 1.4}$	$31.95_{\pm 3.0}$	$51.98_{\pm 0.3}$
	LoRA-One (Zhang et al., 2025)	33M (0.40%)	$79.26_{\pm 0.6}$	$59.67_{\pm 1.8}$	$47.00_{\pm 2.9}$	$55.95_{\pm 0.4}$
	CorDA (Yang et al., 2024b)	33M (0.40%)	$79.15_{\pm 0.4}$	$61.23_{\pm 1.3}$	$48.28_{\pm 1.9}$	$57.36_{\pm 0.4}$
	LoRA-Null (Tang et al., 2025)	33M (0.40%)	$79.07_{\pm 0.5}$	$60.27_{\pm 1.8}$	$47.73_{\pm 1.8}$	$57.11_{\pm 0.3}$
	SDC-LoRA (Ours)	33M (0.40%)	$\mathbf{79.38_{\pm 0.7}}$	$\mathbf{63.37}_{\pm 1.1}$	$\mathbf{49.57}_{\pm 2.5}$	$\mathbf{58.12}_{\pm 0.3}$
Qwen2.5-7B	LoRA (Hu et al., 2022)	32M (0.41%)	$82.00_{\pm 0.5}$	$66.24_{\pm 1.3}$	$42.23_{\pm 2.7}$	$54.73_{\pm 0.4}$
	MiLoRA (Wang et al., 2025a)	32M (0.41%)	$81.74_{\pm 0.6}$	$68.32_{\pm 1.7}$	$42.59_{\pm 1.6}$	$56.46_{\pm 0.3}$
	PiSSA (Meng et al., 2024)	32M (0.41%)	$80.74_{\pm 1.3}$	$46.12_{\pm 2.1}$	$33.54_{\pm 2.1}$	$48.97_{\pm 0.5}$
	LoRA-One (Zhang et al., 2025)	32M (0.41%)	$82.14_{\pm 0.5}$	$66.02_{\pm 1.8}$	$43.57_{\pm 2.0}$	$53.53_{\pm 0.1}$
	CorDA (Yang et al., 2024b)	32M (0.41%)	$81.47_{\pm 0.6}$	$68.35_{\pm 1.6}$	$44.03_{\pm 2.3}$	$56.38_{\pm 0.4}$
	LoRA-Null (Tang et al., 2025)	32M (0.41%)	$82.05_{\pm 0.6}$	$67.67_{\pm 1.8}$	$45.21_{\pm 2.7}$	$54.31_{\pm 0.3}$
	SDC-LoRA (Ours)	32M (0.41%)	$\mathbf{82.50}_{\pm 0.4}$	$\mathbf{70.13}_{\pm 1.9}$	$\mathbf{47.37}_{\pm 1.8}$	$\mathbf{57.90}_{\pm 0.3}$

Table 2: Comparison of baselines and our SDC-LoRA on Code-Feedback. Results are reported as mean with standard deviations over 5 runs (higher is better).

Model	Method	# Params (%)	HumanEval	MMLU	TruthfulQA	HellaSwag
LLaMA3.1-8B	LoRA (Hu et al., 2022)	33M (0.40%)	$48.65_{\pm 1.4}$	$58.70_{\pm 1.3}$	$47.29_{\pm 1.6}$	$55.77_{\pm 0.7}$
	MiLoRA (Wang et al., 2025a)	33M (0.40%)	$51.71_{\pm 1.7}$	$58.18_{\pm 2.1}$	$49.33_{\pm 2.2}$	$55.79_{\pm 0.7}$
	PiSSA (Meng et al., 2024)	33M (0.40%)	$49.76_{\pm 1.7}$	$52.67_{\pm 1.8}$	$41.16_{\pm 1.4}$	$52.39_{\pm 0.3}$
	LoRA-One (Zhang et al., 2025)	33M (0.40%)	$50.85_{\pm 1.8}$	$59.68_{\pm 1.8}$	$47.61_{\pm 2.7}$	$51.39_{\pm 0.4}$
	CorDA (Yang et al., 2024b)	33M (0.40%)	$51.14_{\pm 1.6}$	$60.54_{\pm 1.8}$	$49.18_{\pm 2.9}$	$56.12_{\pm 0.4}$
	LoRA-Null (Tang et al., 2025)	33M (0.40%)	$51.47_{\pm 1.3}$	$59.67_{\pm 1.4}$	$48.42_{\pm 2.4}$	$55.81_{\pm 0.5}$
	SDC-LoRA (Ours)	33M (0.40%)	$\mathbf{52.80}_{\pm 1.5}$	$\mathbf{62.06}_{\pm 1.1}$	$\mathbf{50.47}_{\pm 2.4}$	$\mathbf{58.89}_{\pm 0.4}$
Qwen2.5-7B	LoRA (Hu et al., 2022)	32M (0.41%)	$68.17_{\pm 1.5}$	$66.44_{\pm 1.1}$	$43.33_{\pm 2.4}$	$57.27_{\pm 0.4}$
	MiLoRA (Wang et al., 2025a)	32M (0.41%)	$69.51_{\pm 1.2}$	$66.15_{\pm 0.9}$	$43.33_{\pm 2.7}$	$56.33_{\pm 0.6}$
	PiSSA (Meng et al., 2024)	32M (0.41%)	$69.27_{\pm 1.1}$	$61.32_{\pm 1.1}$	$43.70_{\pm 1.8}$	$55.02_{\pm 0.2}$
	LoRA-One (Zhang et al., 2025)	32M (0.41%)	$70.12_{\pm 1.3}$	$67.46_{\pm 0.5}$	$44.55_{\pm 2.1}$	$57.61_{\pm 0.4}$
	CorDA (Yang et al., 2024b)	32M (0.41%)	$69.38_{\pm 0.6}$	$67.49_{\pm 1.1}$	$44.32_{\pm 2.3}$	$56.41_{\pm 0.5}$
	LoRA-Null (Tang et al., 2025)	32M (0.41%)	$69.86_{\pm 1.0}$	$66.73_{\pm 1.8}$	$43.81_{\pm 2.6}$	$57.15_{\pm 0.3}$
	SDC-LoRA (Ours)	32M (0.41%)	$\mathbf{70.98}_{\pm 0.8}$	$\mathbf{70.18}_{\pm 0.8}$	$\mathbf{46.51}_{\pm 1.1}$	$\mathbf{58.63}_{\pm 0.3}$

tasks. We fine-tune LLaMA3.1-8B-Instruct (Team, 2024) and Qwen-2.5-7B-Chat (Yang et al., 2025) on: (1) 100k samples from MetaMathQA (Yu et al., 2024) and evaluate out-of-domain math reasoning on GSM8K (Cobbe et al., 2021) with direct prompting (no chain-of-thought or tool use); (2) 100k samples from CodeFeedback (Zheng et al., 2024) training set and evaluate PASS@1 on HumanEval (Chen et al., 2021). We use three benchmarks, MMLU (Hendrycks et al., 2021), TruthfulQA (Lin et al., 2022), and HellaSwag (Zellers et al., 2019) to evaluate the degree of pretraining-knowledge forgetting in the fine-tuned model. We compare LoRA (Hu et al., 2022), PiSSA (Meng et al., 2024), MiLoRA (Wang et al., 2025a), LoRA-One (Zhang et al., 2025), CorDA (Yang et al., 2024b) and LoRA-Null (Tang et al., 2025) with rank 16. We perform ablation studies to examine the effectiveness of key SDC-LoRA components and analyze the results to gain deeper insights into its underlying mechanics. Additional experiments refer to Appendix E. For SDC-LoRA, we set the principal-subspace suppression dimension to $S = 256$. Full hyperparameters see Appendix F.

5.1 Math Reasoning

Table 1 reports results on the math reasoning setup. Under the same trainabl parameter budget, SDC-LoRA attains competitive GSM8K accuracy for both backbones (79.38 vs. 79.26 for LLaMA-3.1-8B and 82.50 vs. 82.14 for Qwen2.5-7B). More importantly, it consistently strengthens knowledge retention: on LLaMA-3.1-8B, SDC-LoRA surpasses the best prior baseline (CorDA) by +2.14 on MMLU (63.37 vs. 61.23), +1.29 on TruthfulQA (49.57 vs. 48.28), and +0.76 on HellaSwag (58.12 vs. 57.36); on Qwen2.5-7B, the gains are +1.78 on MMLU (70.13 vs. 68.35), +2.16 on TruthfulQA (47.37 vs. 45.21), and +1.44 on HellaSwag (57.90 vs. 56.46). These results indicate that SDC-LoRA's principal-subspace energy control effectively preserves pretrained knowledge while maintaining or

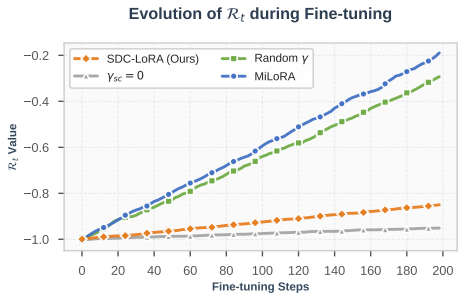

Figure 4: Evolution of $\mathcal{R}t$ during fine-tuning under different γ strategies.

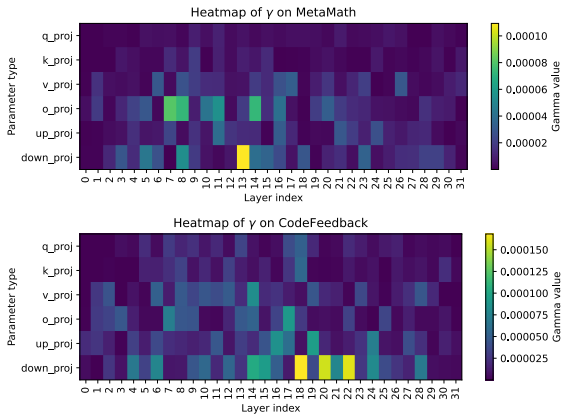

Figure 5: Spectral Calibration Factors across layers and parameter types.

slightly improving target-task performance under a fixed parameter budget.

5.2 Code Generation

Table 2 reports results on the code generation task. With the same number of trainable parameters, SDC-LoRA achieves competitive HumanEval scores on both backbones: 52.80 on LLaMA3.1-8B (+1.09 over MiLoRA) and 70.98 on Qwen2.5-7B (+0.86 over LoRA-One). SDC-LoRA also consistently improves knowledge retention. On LLaMA3.1-8B it surpasses the strongest baselines by +1.52 MMLU (62.06 vs. 60.54, CorDA), +1.14 TruthfulQA (50.47 vs. 49.33, MiLoRA), and +2.77 HellaSwag (58.89 vs. 56.12, CorDA); on Qwen2.5-7B it gains +2.69 MMLU (70.18 vs. 67.49, CorDA), +1.96 TruthfulQA (46.51 vs. 44.55, LoRA-One), and +1.02 HellaSwag (58.63 vs. 57.61, LoRA-One). Together with the math results, these findings show that across tasks and backbones, SDC-LoRA alleviates knowledge forgetting while maintaining or slightly improving downstream performance.

5.3 Ablation Study

Principal Subspace Energy-Controlled Learning & Spectral Calibration. Our experimental results show that applying Principal Subspace Energy-Controlled Learning (PSECL) to suppress updates along the principal subspace tends to improve knowledge retention metrics such as MMLU compared to the unsuppressed setting. From Table 3 and Fig. 4, our Spectral Calibration (SC) factor γ_{sc} exhibits clear advantages over two extreme choices. On the one hand, setting $\gamma = 0$ can further boost knowledge retention scores, but under the tight $r = 2$ configuration it noticeably harms target-task performance (GSM8K is clearly lower than with our spectrally calibrated factor), indicating that completely shutting off the principal subspace over-constraints the model's ability to fit the new task. On the other hand, Random γ sometimes achieves GSM8K comparable to SDC-LoRA, but yields consistently worse MMLU / TruthfulQA / HellaSwag, and its $\mathcal{R}_t$ trajectory almost coincides with MiLoRA, still drifting rapidly toward the principal subspace; this shows that randomly chosen scaling cannot effectively control drift. In contrast, SDC-LoRA combines PSECL with the spectrally calibrated γ_{sc} to precisely regulate principal-subspace update strength, and under both $r = 16$ and $r = 2$ it stably improves knowledge retention while nearly preserving (or slightly improving) GSM8K / HumanEval.

Heatmap of Spectral Calibration. From Fig. 5, we observe that the γ_{sc} values exhibit clear structural variation across layers, parameter types, and datasets, rather than collapsing to a near constant. On both MetaMath and CodeFeedback, middle layers generally have larger γ_{sc}, while shallow and deep layers tend to have smaller γ_{sc}, indicating that gradients in the principal subspace are more "aggressive" at the shallow and deep layers and therefore require stronger suppression. Larger γ_{sc} values are also concentrated in the W_{down}, W_{up}, W_o, whereas W_q, W_k, W_v usually have smaller γ_{sc}, showing that the degree of control is not uniform across parameter types. Moreover, the two tasks display distinct patterns: for example, on MetaMath we see stronger suppression (smaller γ_{sc}) in the upper 16 layers, while on CodeFeedback the stronger suppression appears in the lower 16 layers, suggesting that γ_{sc} adapts to the task distribution. Overall, these observations indicate that our spectrally calibrated γ_{sc} adjusts the strength of principal-subspace updates in a layer, module, and task-aware manner, rather than relying on a single

Table 3: Ablation Study of our SDC-LoRA fine-tuning LLaMA-3.1-8B-Instruct on MetaMath.

	Method	PSECL	SC	GSM8K	MMLU	TruthfulQA	HellaSwag
LoRA_$r = 16$	MiLoRA	✗	✗	$78.95_{\pm 0.7}$	$59.07_{\pm 2.0}$	$47.12_{\pm 2.2}$	$57.03_{\pm 0.4}$
	SDC-LoRA ($\gamma_{\text{sc}} = 0$)	✓	✗	$78.62_{\pm 1.2}$	$63.51_{\pm 0.9}$	$50.21_{\pm 2.1}$	$58.26_{\pm 0.2}$
	SDC-LoRA (Random γ_{sc})	✓	✗	$79.27_{\pm 0.4}$	$60.21_{\pm 2.9}$	$47.47_{\pm 1.9}$	$57.71_{\pm 0.5}$
	SDC-LoRA (Ours)	✓	✓	$79.38_{\pm 0.7}$	$63.37_{\pm 1.1}$	$49.57_{\pm 2.5}$	$58.12_{\pm 0.3}$
LoRA_$r = 2$	MiLoRA	✗	✗	$78.86_{\pm 1.1}$	$61.16_{\pm 1.5}$	$48.33_{\pm 2.4}$	$58.17_{\pm 0.3}$
	SDC-LoRA ($\gamma_{\text{sc}} = 0$)	✓	✗	$77.19_{\pm 0.8}$	$63.67_{\pm 1.1}$	$50.03_{\pm 2.5}$	$58.51_{\pm 0.3}$
	SDC-LoRA (Random γ_{sc})	✓	✗	$78.58_{\pm 0.6}$	$61.32_{\pm 1.9}$	$49.69_{\pm 1.7}$	$58.18_{\pm 0.5}$
	SDC-LoRA (Ours)	✓	✓	$78.65_{\pm 0.6}$	$63.41_{\pm 1.3}$	$50.23_{\pm 2.2}$	$58.36_{\pm 0.3}$

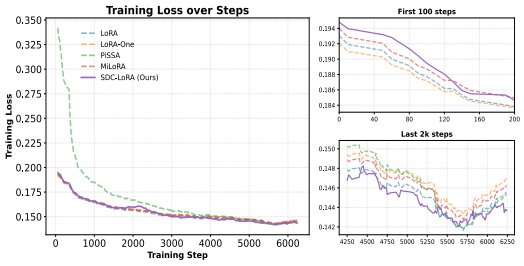

Figure 6: Training loss over steps on MetaMathQA.

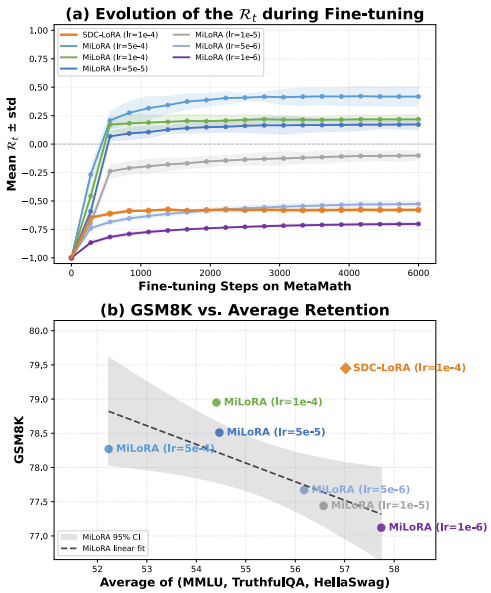

Figure 7: (a) $\mathcal{R}_t$ during MetaMath fine tuning compared MiLoRA; (b) GSM8K vs. average retention with a MiLoRA linear fit and 95% confidence band.

global or random scaling factor, highlighting the necessity and of our spectral calibration design.

5.4 In-Depth Analysis

Training Loss over Steps. From Fig. 6, all methods share almost identical training loss curves on MetaMath: they drop from ∼0.19 to ∼0.14 within a few hundred steps and then converge smoothly, with no extra loss introduced by *SD* suppression. In the zoomed view of the first 100 steps, SDC-LoRA's initial descent rate is very close to that of MiLoRA, only slightly higher at the beginning but already aligned around $step = 200$, indicating that scaling in the principal subspace does not significantly slow early optimization. In the zoomed view of the last $2k$ steps, SDC-LoRA consistently attains the lowest or near-lowest loss and exhibits smaller fluctuations than MiLoRA and PiSSA, with slightly better tail loss, showing that our method does not harm and may even improve long-horizon optimization quality. Overall, this comparison provides direct evidence that our SDC-LoRA can suppress *SD* and mitigate knowledge forgetting without sacrificing convergence speed or target-task optimization.

Comparing with MiLoRA under Different Learning Rate. Fig. 7(b) shows that MiLoRA with different learning rates lies on a clear negatively sloped regression line: smaller learning rate yield higher average retention (MMLU, TruthfulQA, HellaSwag) but lower GSM8K, forming a small learning rate frontier. SDC-LoRA, however, sits in the upper-right of this band, attaining both higher retention and higher GSM8K than MiLoRA even at its smallest learning rate, which cannot be explained by simply shrinking the step size. Consistently, Fig. 7(a) shows that, SDC-LoRA markedly suppresses the early rise of $\mathcal{R}_t$ without the performance loss seen for MiLoRA, because it selectively scales only the principal-subspace component of the update rather than uniformly shrinking all directions. Together, these results indicate that SDC-LoRA breaks the small-learning-rate trade-off and more effectively balances target-task performance with retention of pretraining knowledge.

6 Conclusion

We identify and analyze *Singular-subspace Drift* (SD), where stronger gradients in the principal singular subspace of W_0 drive LoRA updates from minor to principal dominant directions and erode pretrained knowledge. To counter this, we propose SDC-LoRA, which performs *Principal Sub-*

space Energy-Controlled Learning by decomposing each update into principal and complementary components and rescaling only the principal part with a *Spectral Calibration* factor γ_{sc} that controls their energy ratio. Across LLaMA and Qwen on math reasoning and code generation, SDC-LoRA consistently reduces forgetting on MMLU, TruthfulQA, and HellaSwag while matching or improving GSM8K and HumanEval, achieving a better trade-off between target accuracy and pretraining-knowledge retention.

Limitations

Despite the superior performance of SDC-LoRA, there are still several limitations. First, SDC-LoRA requires a one-step full-parameter gradient and an SVD on the restricted principal/minor blocks of each weight matrix to obtain γ_{sc}, which adds a moderate one-off overhead compared to vanilla LoRA; while this cost is small in our setups, it may be non-trivial for even larger models or more complex optimizer stacks. Second, our theoretical analysis is based on a local linearization around W_0 and small-step assumptions, so the guarantees do not formally cover very aggressive learning-rate schedules or extremely long fine-tuning horizons. Finally, we have evaluated SDC-LoRA on two backbones (LLaMA and Qwen) and two task families (math reasoning and code generation); its behavior on other domains (e.g., multimodal instruction tuning, RLHF or continual learning settings) remains to be systematically explored. We leave addressing these limitations to future work.

Acknowledgments

This work was supported by National Natural Science Foundation of China (No.62176252).

References

Armen Aghajanyan, Sonal Gupta, and Luke Zettlemoyer. 2021. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 7319–7328. Association for Computational Linguistics.

Dan Biderman, Jacob P. Portes, Jose Javier Gonzalez Ortiz, Mansheej Paul, Philip Greengard, Connor Jennings, Daniel King, Sam Havens, Vitaliy Chiley, Jonathan Frankle, Cody Blakeney, and John Patrick Cunningham. 2024. Lora learns less and forgets less. *Trans. Mach. Learn. Res.*, 2024.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, and 39 others. 2021. Evaluating large language models trained on code. *CoRR*, abs/2107.03374.

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. 2021. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168.

Ziqi Gao, Qichao Wang, Aochuan Chen, Zijing Liu, Bingzhe Wu, Liang Chen, and Jia Li. 2024. Parameter-efficient fine-tuning with discrete fourier transform. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Ian J Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. 2013. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*.

Habib Hajimolahoseini, Mehdi Rezagholizadeh, Vahid Partovinia, Marzieh Tahaei, Omar Mohamed Awad, and Yang Liu. 2021. Compressing pre-trained language models using progressive low rank decomposition. *Advances in Neural Information Processing Systems*, 35:6–14.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. 2019. Learning a unified classifier incrementally via rebalancing. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 831–839. Computer Vision Foundation / IEEE.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, and 1 others. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526.

Dawid Jan Kopiczko, Tijmen Blankevoort, and Yuki Markus Asano. 2023. Vera: Vector-based random matrix adaptation. *CoRR*, abs/2310.11454.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 3045–3059. Association for Computational Linguistics.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 4582–4597. Association for Computational Linguistics.

Xiaojie Li, Shaowei He, Jianlong Wu, Yue Yu, Liqiang Nie, and Min Zhang. 2023. Mask again: Masked knowledge distillation for masked video modeling. In *Proceedings of the 31st ACM International Conference on Multimedia, MM 2023, Ottawa, ON, Canada, 29 October 2023- 3 November 2023*, pages 2221–2232. ACM.

Yixia Li, Boya Xiong, Guanhua Chen, and Yun Chen. 2024. Setar: Out-of-distribution detection with selective low-rank approximation. *Advances in Neural Information Processing Systems*, 37:72840–72871.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 3214–3252. Association for Computational Linguistics.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. 2024a. Dora: Weight-decomposed low-rank adaptation. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Weiyang Liu, Zeju Qiu, Yao Feng, Yuliang Xiu, Yuxuan Xue, Longhui Yu, Haiwen Feng, Zhen Liu, Juyeon Heo, Songyou Peng, Yandong Wen, Michael J. Black, Adrian Weller, and Bernhard Schölkopf. 2024b. Parameter-efficient orthogonal finetuning via butterfly factorization. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Michael McCloskey and Neal J Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier.

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. 2024. Pissa: Principal singular values and singular vectors adaptation of large language models. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021a. Adapterfusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 487–503. Association for Computational Linguistics.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021b. Adapterfusion: Non-destructive task composition for transfer learning. *Preprint*, arXiv:2005.00247.

Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pre-trained models for natural language processing: A survey. *CoRR*, abs/2003.08271.

Pratyusha Sharma, Jordan T Ash, and Dipendra Misra. 2023. The truth is in there: Improving reasoning in language models with layer-selective rank reduction. *arXiv preprint arXiv:2312.13558*.

Pengwei Tang, Yong Liu, Dongjie Zhang, Xing Wu, and Debing Zhang. 2025. Lora-null: Low-rank adaptation via null space for large language models. *arXiv preprint arXiv:2503.02659*.

Llama Team. 2024. The llama 3 herd of models. *CoRR*, abs/2407.21783.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, and 49 others. 2023b. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Mojtaba Valipour, Mehdi Rezagholizadeh, Ivan Kobyzev, and Ali Ghodsi. 2023. Dylora: Parameter-efficient tuning of pre-trained models using dynamic

search-free low-rank adaptation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, pages 3266–3279. Association for Computational Linguistics.

Hanqing Wang, Yixia Li, Shuo Wang, Guanhua Chen, and Yun Chen. 2025a. Milora: Harnessing minor singular components for parameter-efficient llm finetuning. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4823–4836.

Shaowen Wang, Linxi Yu, and Jian Li. 2024. Lora-ga: Low-rank adaptation with gradient approximation. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Zhengbo Wang, Jian Liang, Ran He, Zilei Wang, and Tieniu Tan. 2025b. Lora-pro: Are low-rank adapters properly optimized? In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.

Shipeng Yan, Jiangwei Xie, and Xuming He. 2021. DER: dynamically expandable representation for class incremental learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 3014–3023. Computer Vision Foundation / IEEE.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 40 others. 2024a. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 23 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Yibo Yang, Xiaojie Li, Zhongzhu Zhou, Shuaiwen Song, Jianlong Wu, Liqiang Nie, and Bernard Ghanem. 2024b. Corda: Context-oriented decomposition adaptation of large language models for task-aware parameter-efficient fine-tuning. *Advances in Neural Information Processing Systems*, 37:71768–71791.

Yibo Yang, Haobo Yuan, Xiangtai Li, Zhouchen Lin, Philip H. S. Torr, and Dacheng Tao. 2023. Neural collapse inspired feature-classifier alignment for few-shot class-incremental learning. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T. Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2024. Metamath: Bootstrap your own mathematical questions for large language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Shen Yuan, Haotian Liu, and Hongteng Xu. 2024. Bridging the gap between low-rank and orthogonal adaptation via householder reflection adaptation. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Elad Ben Zaken, Yoav Goldberg, and Shauli Ravfogel. 2022. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 1–9. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4791–4800. Association for Computational Linguistics.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. 2023. Adalora: Adaptive budget allocation for parameter-efficient fine-tuning. *Preprint*, arXiv:2303.10512.

Yuanhe Zhang, Fanghui Liu, and Yudong Chen. 2025. Lora-one: One-step full gradient could suffice for fine-tuning large language models, provably and efficiently. *arXiv preprint arXiv:2502.01235*.

Tianyu Zheng, Ge Zhang, Tianhao Shen, Xueling Liu, Bill Yuchen Lin, Jie Fu, Wenhu Chen, and Xiang Yue. 2024. Opencodeinterpreter: Integrating code generation with execution and refinement. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 12834–12859. Association for Computational Linguistics.

A Gradient Projection Transform

A.1 Gradient Projection Transform for Principal Subspace

Starting from the linearized LoRA updates

$$A_{t+1} \approx A_t - \eta B_t^\top G, \qquad B_{t+1} \approx B_t - \eta G A_t^\top, \tag{7}$$

where $G := \nabla_W \mathcal{L}(W)\big|_{W=W_0}$ is the one-step full-parameter gradient.

We project the LoRA parameters onto the principal singular subspace of W_0, spanned by the top-S singular vectors U_p and V_p:

$$\widetilde{A}_{p,t} := A_t V_p \in \mathbb{R}^{r\times S}, \qquad \widetilde{B}_{p,t} := U_p^\top B_t \in \mathbb{R}^{S\times r}.$$

Update of $\widetilde{A}_{p,t}$. Right-multiplying the update for A_t in Eq. (7) by V_p gives:

$$\begin{aligned}\widetilde{A}_{p,t+1} := A_{t+1}V_p &\approx (A_t - \eta B_t^\top G)V_p \\ &= \widetilde{A}_{p,t} - \eta B_t^\top G V_p. \end{aligned} \tag{12}$$

To analyze the exact coupling, we decompose B_t using the orthogonal projector $P_p = U_p U_p^\top$ and its orthogonal complement $P_\perp = I - U_p U_p^\top$:

$$B_t = U_p(U_p^\top B_t) + P_\perp B_t = U_p \widetilde{B}_{p,t} + B_{t,\perp}.$$

Substituting this into the gradient term:

$$\begin{aligned}B_t^\top G V_p &= (U_p \widetilde{B}_{p,t} + B_{t,\perp})^\top G V_p \\ &= \widetilde{B}_{p,t}^\top (U_p^\top G V_p) + B_{t,\perp}^\top G V_p. \end{aligned} \tag{13}$$

We define the principal restricted gradient as $H_p := U_p^\top G V_p \in \mathbb{R}^{S\times S}$. The second term, $B_{t,\perp}^\top G V_p$, represents the cross-coupling from the minor/orthogonal subspace. We neglect this term based on the Spectral Alignment Assumption: in pre-trained models, the gradient G typically aligns with the weight basis, meaning the off-diagonal block $P_\perp^\top G V_p$ is structurally small compared to the diagonal block H_p. Furthermore, due to Eigenvalue Dominance, the dynamics of the principal block are governed by the large singular values in H_p, rendering the linear contribution from the cross-term negligible in determining the exponential growth rate. Hence, we obtain the decoupled update:

$$\widetilde{A}_{p,t+1} \approx \widetilde{A}_{p,t} - \eta\, \widetilde{B}_{p,t}^\top H_p. \tag{14}$$

Update of $\widetilde{B}_{p,t}$. Similarly, left-multiplying the update for B_t in Eq. (7) by $U_p^\top$ yields:

$$\begin{aligned}\widetilde{B}_{p,t+1} := U_p^\top B_{t+1} &\approx U_p^\top (B_t - \eta G A_t^\top) \\ &= \widetilde{B}_{p,t} - \eta U_p^\top G A_t^\top. \end{aligned} \tag{15}$$

Using the decomposition $A_t = \widetilde{A}_{p,t} V_p^\top + A_{t,\perp}$ with the projector $V_p V_p^\top$:

$$\begin{aligned}U_p^\top G A_t^\top &= U_p^\top G(\widetilde{A}_{p,t}V_p^\top + A_{t,\perp})^\top \\ &= (U_p^\top G V_p)\widetilde{A}_{p,t}^\top + U_p^\top G A_{t,\perp}^\top \\ &= H_p \widetilde{A}_{p,t}^\top + U_p^\top G A_{t,\perp}^\top. \end{aligned} \tag{16}$$

Again, we neglect the cross-term $U_p^\top G A_{t,\perp}^\top$ under the same Spectral Alignment and Eigenvalue Dominance assumptions. Therefore:

$$\widetilde{B}_{p,t+1} \approx \widetilde{B}_{p,t} - \eta\, H_p \widetilde{A}_{p,t}^\top. \tag{17}$$

Result. Combining the two relations above, the parameter updates in the principal subspace of W_0 are approximately decoupled as:

$$\boxed{\begin{aligned}\widetilde{A}_{p,t+1} &\approx \widetilde{A}_{p,t} - \eta\, \widetilde{B}_{p,t}^\top H_p, \\ \widetilde{B}_{p,t+1} &\approx \widetilde{B}_{p,t} - \eta\, H_p \widetilde{A}_{p,t}^\top, \end{aligned}} \tag{8}$$

which is exactly Equation (8) in the main text.

A.2 Derivation for the Minor Subspace Dynamics

The derivation for the minor subspace follows a symmetric logic to the principal subspace, focusing on the projections onto the minor singular bases U_m and V_m of W_0.

We define the projected LoRA parameters in the minor subspace:

$$\widetilde{A}_{m,t} := A_t V_m \in \mathbb{R}^{r\times S}, \qquad \widetilde{B}_{m,t} := U_m^\top B_t \in \mathbb{R}^{S\times r}.$$

Update of $\widetilde{A}_{m,t}$. Right-multiplying the linearized update for A_t (Eq. 7) by V_m:

$$\begin{aligned}\widetilde{A}_{m,t+1} := A_{t+1}V_m &\approx (A_t - \eta B_t^\top G)V_m \\ &= \widetilde{A}_{m,t} - \eta B_t^\top G V_m. \end{aligned} \tag{18}$$

We decompose B_t using the orthogonal projector $U_m U_m^\top$ onto the minor subspace and its orthogonal complement $P_{\perp_m} = I - U_m U_m^\top$:

$$B_t = U_m(U_m^\top B_t) + P_{\perp_m} B_t = U_m \widetilde{B}_{m,t} + B_{t,\perp_m}.$$

The gradient term then becomes:

$$\begin{aligned}B_t^\top G V_m &= (U_m \widetilde{B}_{m,t} + B_{t,\perp_m})^\top G V_m \\ &= \widetilde{B}_{m,t}^\top (U_m^\top G V_m) + B_{t,\perp_m}^\top G V_m. \end{aligned} \tag{19}$$

We define the minor restricted gradient as $H_m := U_m^\top G V_m$. The cross-term from the principal subspace, $B_{t,\perp_m}^\top G V_m$, is critical and normally causes interference in standard LoRA. However, in the specific context of our SDC-LoRA method, this term becomes negligible because: (1) we initialize with zero principal energy ($B_{p,0} = 0$), and (2) our compensation factor γ actively suppresses the growth of the principal component, preventing it from gaining sufficient magnitude to significantly perturb the minor subspace dynamics. Thus, we obtain:

$$\widetilde{A}_{m,t+1} \approx \widetilde{A}_{m,t} - \eta\, \widetilde{B}_{m,t}^\top H_m. \tag{20}$$

Update of $\widetilde{B}_{m,t}$. Similarly, left-multiplying the update for B_t (Eq. 7) by $U_m^\top$:

$$\begin{aligned}\widetilde{B}_{m,t+1} := U_m^\top B_{t+1} &\approx U_m^\top (B_t - \eta G A_t^\top) \\ &= \widetilde{B}_{m,t} - \eta U_m^\top G A_t^\top. \quad (21)\end{aligned}$$

Using the decomposition $A_t = \widetilde{A}_{m,t} V_m^\top + A_{t,\perp_m}$ via the projector $V_m V_m^\top$:

$$\begin{aligned}U_m^\top G A_t^\top &= U_m^\top G (\widetilde{A}_{m,t} V_m^\top + A_{t,\perp_m})^\top \\ &= (U_m^\top G V_m) \widetilde{A}_{m,t}^\top + U_m^\top G A_{t,\perp_m}^\top \\ &= H_m \widetilde{A}_{m,t}^\top + U_m^\top G A_{t,\perp_m}^\top. \quad (22)\end{aligned}$$

Neglecting the cross-term $U_m^\top G A_{t,\perp_m}^\top$ under the same weak-coupling assumption yields:

$$\widetilde{B}_{m,t+1} \approx \widetilde{B}_{m,t} - \eta\, H_m \widetilde{A}_{m,t}^\top. \quad (23)$$

Result. Combining the relations above, the decoupled parameter updates in the minor subspace are:

$$\boxed{\begin{aligned}\widetilde{A}_{m,t+1} &\approx \widetilde{A}_{m,t} - \eta\, \widetilde{B}_{m,t}^\top H_m, \\ \widetilde{B}_{m,t+1} &\approx \widetilde{B}_{m,t} - \eta\, H_m \widetilde{A}_{m,t}^\top.\end{aligned}} \quad (24)$$

These dynamics mirror those of the principal subspace but are governed by the minor restricted gradient H_m.

B Proof of Theorem 1

Sketch. We work under the linearized update around W_0 given by

$$A_{t+1} \approx A_t - \eta\, B_t^\top G, \qquad B_{t+1} \approx B_t - \eta\, G A_t^\top,$$

with $G = \nabla_W \mathcal{L}(W)\big|_{W=W_0}$ and a small stepsize $\eta > 0$. Projecting onto the principal and minor subspaces of W_0 we obtain

$$\begin{aligned}\widetilde{A}_{p,t+1} &\approx \widetilde{A}_{p,t} - \eta\, \widetilde{B}_{p,t}^\top H_p, \\ \widetilde{B}_{p,t+1} &\approx \widetilde{B}_{p,t} - \eta\, H_p\, \widetilde{A}_{p,t}^\top, \\ \widetilde{A}_{m,t+1} &\approx \widetilde{A}_{m,t} - \eta\, \widetilde{B}_{m,t}^\top H_m, \\ \widetilde{B}_{m,t+1} &\approx \widetilde{B}_{m,t} - \eta\, H_m\, \widetilde{A}_{m,t}^\top,\end{aligned} \quad (25)$$

where $H_p = U_p^\top G V_p \in \mathbb{R}^{S\times S}$ and $H_m = U_m^\top G V_m \in \mathbb{R}^{S\times S}$ are the principal and minor restricted gradients.

Take economy-size SVDs $H_p = P_p \Sigma_p Q_p^\top$, $H_m = P_m \Sigma_m Q_m^\top$, with

$$\begin{aligned}\Sigma_p &= \mathrm{diag}(\sigma_1^p, \ldots, \sigma_S^p), \\ \Sigma_m &= \mathrm{diag}(\sigma_1^m, \ldots, \sigma_S^m),\end{aligned} \quad (26)$$

and $\sigma_1^p \geq \cdots \geq \sigma_S^p \geq 0$, $\sigma_1^m \geq \cdots \geq \sigma_S^m \geq 0$. Rotate coordinates within each block:

$$\begin{aligned}\widehat{A}_{p,t} &:= \widetilde{A}_{p,t} Q_p, \quad \widehat{B}_{p,t} := P_p^\top \widetilde{B}_{p,t}, \\ \widehat{A}_{m,t} &:= \widetilde{A}_{m,t} Q_m, \quad \widehat{B}_{m,t} := P_m^\top \widetilde{B}_{m,t}.\end{aligned} \quad (27)$$

In this basis, the S principal modes decouple: for each $i = 1, \ldots, S$,

$$\begin{aligned}\widehat{a}_{i,t+1} &= \widehat{a}_{i,t} - \eta\, \sigma_i^p\, \widehat{b}_{i,t}, \\ \widehat{b}_{i,t+1} &= \widehat{b}_{i,t} - \eta\, \sigma_i^p\, \widehat{a}_{i,t},\end{aligned} \quad (28)$$

and analogously on the minor block with σ_j^m.

Define the energy of mode i as the squared Frobenius norm of its contribution rank-one matrix:

$$e_{i,t}^p := \big\|\widehat{b}_{i,t}\widehat{a}_{i,t}^\top\big\|_F^2 = \|\widehat{b}_{i,t}\|_2^2\, \|\widehat{a}_{i,t}\|_2^2. \quad (29)$$

We approximate the total principal projection energy as the sum of per-mode energies:

$$E_{p,t} \approx \sum_{i=1}^{S} e_{i,t}^p. \quad (30)$$

Note: This approximation assumes that cross-modal interference terms are negligible, which holds due to the high-dimensional quasi-orthogonality of the updated vectors.

A direct calculation on the 2-dimensional system $(\widehat{a}_{i,t}, \widehat{b}_{i,t}) \mapsto (\widehat{a}_{i,t+1}, \widehat{b}_{i,t+1})$ shows that the maximal possible growth of $e_{i,t}^p$ in one step occurs when $\widehat{a}_{i,t}$ and $\widehat{b}_{i,t}$ are aligned with the eigenvector $(1, -1)$ of the 2×2 matrix $\left(\begin{smallmatrix} 1 & -\eta\sigma_i^p \\ -\eta\sigma_i^p & 1 \end{smallmatrix}\right)$. In this "alignment" regime we obtain

$$e_{i,t+1}^p = \big(1 + \eta\, \sigma_i^p\big)^4 e_{i,t}^p,$$

The same reasoning applies to the minor block, yielding $e_{i,t+1}^m = \big(1 + \eta\, \sigma_i^m\big)^4 e_{i,t}^m$,. Please refer to Appendix C for the proof.

Assumption 1 (Asymptotic alignment regime)**.** *Motivated by the analysis in Appendix C, which shows that update dynamics drive the vectors $(\widehat{a}, \widehat{b})$ towards the principal eigenvector of the update matrix, we assume the system is in the asymptotic alignment regime. Specifically, we assume the per-mode energy growth is governed by the spectral rate:*

$$\begin{aligned}e_{i,t+1}^p &\approx (1 + \eta\sigma_i^p)^4 e_{i,t}^p, \\ e_{j,t+1}^m &\approx (1 + \eta\sigma_j^m)^4 e_{j,t}^m \quad \textit{for all } i, j.\end{aligned} \quad (31)$$

Now consider SDC-LoRA with the conservative coefficient

$$\gamma_{\mathrm{sc}} = \frac{\sigma^m_{\min}}{\sigma^p_{\max}},$$
$$\text{where} \quad \sigma^p_{\max} := \max_i \sigma^p_i, \quad \sigma^m_{\min} := \min_j \sigma^m_j. \tag{32}$$

Scaling only the principal block by γ_{sc} is equivalent to replacing σ^p_i by $\gamma_{\mathrm{sc}}\sigma^p_i$ in the principal-mode dynamics. Thus the compensated growth factor of mode i is

$$\begin{aligned} \frac{e^p_{i,t+1}}{e^p_{i,t}} &= \left(1 + \eta\, \gamma_{\mathrm{sc}} \sigma^p_i\right)^4 \\ &= \left(1 + \eta\, \frac{\sigma^m_{\min}}{\sigma^p_{\max}}\, \sigma^p_i\right)^4 \\ &\le \left(1 + \eta\, \sigma^m_{\min}\right)^4, \end{aligned} \tag{33}$$

where we used $\sigma^p_i \le \sigma^p_{\max}$ and the monotonicity of $(1+\eta x)^4$ in x.

On the minor block, Assumption 1 gives

$$\frac{e^m_{j,t+1}}{e^m_{j,t}} \approx (1 + \eta \sigma^m_j)^4, \tag{34}$$

and since $\sigma^m_j \ge \sigma^m_{\min}$ and $(1+\eta x)^4$ is increasing,

$$(1 + \eta\, \sigma^m_{\min})^4 \le (1 + \eta\, \sigma^m_j)^4, \quad \forall j. \tag{35}$$

Denote the total principal and minor energies by $E_{p,t} \approx \sum_i e^p_{i,t}$ and $E_{m,t} \approx \sum_j e^m_{j,t}$, and define the per-block growth factors

$$\begin{aligned} \alpha_p(\gamma) &:= \sum_i w^p_{i,t}\, (1 + \eta\gamma \sigma^p_i)^4, \\ \alpha_m &:= \sum_j w^m_{j,t}\, (1 + \eta \sigma^m_j)^4, \end{aligned} \tag{36}$$

where $w^p_{i,t} := e^p_{i,t}/E_{p,t}$ and $w^m_{j,t} := e^m_{j,t}/E_{m,t}$ are energy weights. These are convex combinations of the per-mode factors.

From the bound above we have, for every principal mode,

$$(1 + \eta\, \gamma_{\mathrm{sc}} \sigma^p_i)^4 \le (1 + \eta\, \sigma^m_{\min})^4,$$

hence

$$\alpha_p(\gamma_{\mathrm{sc}}) = \sum_i w^p_{i,t} (1 + \eta\, \gamma_{\mathrm{sc}} \sigma^p_i)^4 \le (1 + \eta\, \sigma^m_{\min})^4. \tag{37}$$

On the minor block, using the inequality above for each j,

$$(1 + \eta\, \sigma^m_{\min})^4 \le \sum_j w^m_{j,t} (1 + \eta\, \sigma^m_j)^4 = \alpha_m. \tag{38}$$

Therefore

$$\alpha_p(\gamma_{\mathrm{sc}}) \le \alpha_m.$$

Under the linearized dynamics, the block energies evolve approximately as $E_{p,t+1} \approx \alpha_p(\gamma_{\mathrm{sc}}) E_{p,t}$ and $E_{m,t+1} \approx \alpha_m E_{m,t}$, so the energy ratio $\rho_t := E_{p,t}/E_{m,t}$ satisfies

$$\rho_{t+1} \approx \frac{\alpha_p(\gamma_{\mathrm{sc}})}{\alpha_m}\, \rho_t \le \rho_t.$$

Since the contrast $\mathcal{R}_t = (E_{p,t} - E_{m,t})/(E_{p,t} + E_{m,t})$ is a strictly increasing function of ρ_t on $(0, \infty)$, a non-increasing ρ_t implies a non-increasing $\mathcal{R}_t$ in this linearized regime. In particular, applying γ_{sc} prevents growth of the contrast and thus suppresses singular-subspace drift at the one-step level. □

C Justification of the per-mode energy growth via a 2-D linear system

We look at one principal mode and drop the superscript/suffix for clarity. The linearized 2-dimensional system for this mode is

$$a_{t+1} = a_t - \eta\sigma\, b_t, \qquad b_{t+1} = b_t - \eta\sigma\, a_t, \tag{39}$$

which can be written as

$$\begin{pmatrix} a_{t+1} \\ b_{t+1} \end{pmatrix} = M \begin{pmatrix} a_t \\ b_t \end{pmatrix}, \qquad M := \begin{pmatrix} 1 & -\eta\sigma \\ -\eta\sigma & 1 \end{pmatrix}.$$

For this mode we define the energy

$$e_t := (a_t b_t)^2. \tag{40}$$

Exact growth in the "aligned" case. Suppose (a_t, b_t) is aligned with the eigenvector $(1, -1)$ of M, i.e. $b_t = -a_t$. Then

$$\begin{aligned} a_{t+1} &= a_t - \eta\sigma(-a_t) = (1 + \eta\sigma) a_t, \\ b_{t+1} &= -a_t - \eta\sigma a_t = -(1 + \eta\sigma) a_t. \end{aligned} \tag{41}$$

Hence

$$\begin{aligned} e_{t+1} &= (a_{t+1} b_{t+1})^2 = \left((1 + \eta\sigma)^2 a_t b_t\right)^2 \\ &= (1 + \eta\sigma)^4 e_t. \end{aligned} \tag{42}$$

This gives the claimed growth factor in the "alignment" regime.

Growth dynamics for general (a_t, b_t). For general initial states (a_t, b_t), we observe that the update matrix M has eigenvalues $\lambda_1 = 1 + \eta\sigma$ and $\lambda_2 = 1 - \eta\sigma$ corresponding to eigenvectors $v_1 = (1, -1)$ and $v_2 = (1, 1)$. Since $\lambda_1 > \lambda_2$ (assuming $\eta\sigma > 0$), the component along v_1 is amplified by a factor of $1 + \eta\sigma$ at each step, while the component along v_2 grows more slowly (or shrinks).

Consequently, for any initialization not orthogonal to v_1, the vector (a_t, b_t) rapidly aligns with the principal eigenvector direction $b_t \approx -a_t$ (which corresponds to $s \to -1$). While the transient relative energy growth e_{t+1}/e_t can be arbitrarily large when e_t is near zero (e.g., at initialization), the *asymptotic* per-step energy growth converges to the spectral rate calculated above:

$$\lim_{t\to\infty} \frac{e_{t+1}}{e_t} = (1 + \eta\sigma)^4.$$

Thus, $(1+\eta\sigma)^4$ represents the characteristic growth rate of the principal mode under typical training dynamics.

D Proof of convergence

Convergence of principal-subspace scaling. Let $f(W) = \mathcal{L}(W)$ be L-smooth and lower bounded. Let P_p and P_m be the orthogonal projectors onto the principal and minor singular subspaces of W_0 (so $P_p+P_m = I$, $P_pP_m = 0$). Define the anisotropic preconditioner

$$M_\gamma \;:=\; P_m \;+\; \gamma\, P_p, \qquad \gamma \in (0, 1].$$

Under the small-step linearization around W_0 (Section 3.1), one SDC-LoRA step is locally equivalent to the preconditioned gradient step

$$W_{t+1} \;=\; W_t \;-\; \eta\, M_\gamma\, \nabla f(W_t).$$

Lemma 1 (Monotone descent under L-smoothness)**.** *If f is L-smooth and $\eta \in (0, 1/L]$, then for any $\gamma \in (0, 1]$,*

$$f(W_{t+1}) \;\le f(W_t) \;-\; \eta\Big(1 - \tfrac{L\eta}{2}\Big)\, \nabla f(W_t)^\top M_\gamma \nabla f(W_t). \tag{43}$$

In particular, since $M_\gamma \succeq \gamma I$ and $\|M_\gamma g\| \le \|g\|$,

$$f(W_{t+1}) \;\le\; f(W_t) \;-\; \eta\Big(1 - \tfrac{L\eta}{2}\Big)\, \gamma\, \|\nabla f(W_t)\|^2.$$

Proof sketch. By L-smoothness (Descent Lemma),

$$f(W_{t+1}) \le f(W_t) - \eta\langle \nabla f(W_t), M_\gamma \nabla f(W_t)\rangle + \tfrac{L\eta^2}{2}\|M_\gamma \nabla f(W_t)\|^2. \tag{44}$$

Since the eigenvalues of M_γ are in $\{\gamma, 1\} \subseteq (0, 1]$, we have $M_\gamma^2 \preceq M_\gamma$, which implies $\|M_\gamma g\|^2 \le g^\top M_\gamma g$. Substituting this into the inequality yields:

$$f(W_{t+1}) \;\le f(W_t) \;-\; \eta\Big(1 - \tfrac{L\eta}{2}\Big)\, \nabla f(W_t)^\top M_\gamma \nabla f(W_t). \tag{45}$$

Finally, use $M_\gamma \succeq \gamma I$ to obtain the stated bound in terms of $\|\nabla f\|^2$. □

Lemma 1 shows that SDC-LoRA remains a descent method whenever vanilla gradient descent would descend (same stepsize regime), and the decrease per step is lower bounded by a factor proportional to γ. Hence the method *does not* break convergence; it at most scales the per-step decrease by a constant factor.

Theorem 2 (Convergence to stationary points)**.** *Let f be L-smooth and lower bounded, and choose $\eta \in (0, 1/L]$. Then the sequence $\{W_t\}$ generated by $W_{t+1} = W_t - \eta M_\gamma \nabla f(W_t)$ satisfies*

$$\sum_{t=0}^{T-1} \|\nabla f(W_t)\|^2 \;\le\; \frac{f(W_0) - f^\star}{\eta\gamma\,(1 - \frac{L\eta}{2})},$$

so $\min_{0\le t<T} \|\nabla f(W_t)\|^2 \to 0$ as $T \to \infty$. Moreover, M_γ is invertible with eigenvalues in $\{\gamma, 1\}$, so $M_\gamma \nabla f(W) = 0 \iff \nabla f(W) = 0$: the stationary points are unchanged by the scaling.

Thus, in the general nonconvex setting, SDC-LoRA converges to the same set of stationary points as vanilla gradient descent, with the usual sublinear rate up to the constant factor γ.

Corollary 1 (Linear rate under PL/strong convexity)**.** *Suppose f satisfies the Polyak–Łojasiewicz (PL) inequality with constant $\mu > 0$ in the region visited, i.e., $\frac{1}{2}\|\nabla f(W)\|^2 \ge \mu\,(f(W) - f^\star)$. If $\eta \in (0, 1/L]$, then*

$$f(W_{t+1}) - f^\star \;\le\; \big(1 - \eta\,\mu\,\gamma\big)\,\big(f(W_t) - f^\star\big),$$

so SDC-LoRA enjoys linear *convergence with rate factor $1 - \eta\mu\gamma$.*

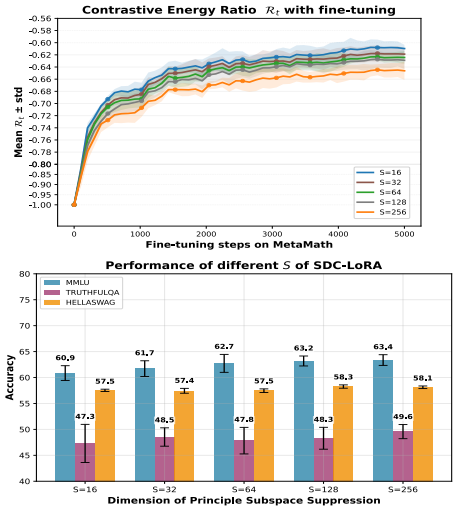

Figure 8: Effect of the principal subspace suppression dimension S in SDC-LoRA: (a) evolution of R_t during fine-tuning on MetaMath; (b) MMLU, TruthfulQA, and HellaSwag accuracy over different S.

Error to vanilla GD and acceptability. Let $g_t = \nabla f(W_t)$. The SDC-LoRA step uses $\tilde{g}_t := M_\gamma g_t = g_t - (1-\gamma)P_p g_t$. Hence the *update distortion* relative to vanilla GD is

$$\|\tilde{g}_t - g_t\| \;=\; (1-\gamma)\,\|P_p g_t\| \;\le\; (1-\gamma)\,\|g_t\|,$$

i.e., a bounded, *directional* shrinkage only along the principal subspace. Because $M_\gamma \succeq \gamma I$, descent is preserved and the convergence rate degrades by at most a constant factor γ. Intuitively, SDC-LoRA behaves like a benign preconditioner: it attenuates steps along high-curvature, knowledge-bearing principal directions while leaving the minor directions unchanged, which stabilizes training without eliminating useful learning signals.

Link to drift control. The results above show that such a choice *does not* compromise convergence: it preserves descent (Lemma 1), converges to the same stationary points (Theorem 2), and retains linear rates under PL (Corollary 1), with only a constant-factor slowdown governed by γ. In practice, since SDC-LoRA keeps full step size in the minor subspace and only scales the principal component, the slowdown is limited while forgetting risk is substantially reduced.

E Additional Experiments

E.1 Principal Subspace Suppression Dimension.

We ablate the principal subspace suppression size S in SDC-LoRA by sweeping $S \in \{16, 32, 64, 128, 256\}$. For each S, we precompute from W_0 the top-S left/right singular vectors and, during fine-tuning, attenuate only the gradient components projected onto this S-dimensional principal subspace, keeping all other settings (learning rate, batch size, steps, LoRA rank, and trainable parameter count) identical. We track the contrastive energy ratio $\mathcal{R}_t$ at every step, averaging across layers and seeds, with a fixed evaluation window R for the top-R versus bottom-R singular directions independent of S, and evaluate knowledge retention on MMLU, TruthfulQA, and HellaSwag (Fig. 8). As S increases, the $\mathcal{R}_t$ trajectory shifts upward and becomes smoother, and retention consistently improves, with $S = 128$ and $S = 256$ yielding the most stable anti-forgetting behavior. In contrast, $S = 16$ underperforms $S = 256$, indicating that too small an S leaves principal modes under-regulated. In practice, we recommend $S \in \{128, 256\}$ by default, and $S = 64$ as a more memory-efficient alternative.

Table 4: Training time and GPU memory usage on MetaMathQA for standard LoRA and our SDC-LoRA

	LoRA	SDC-LoRA
Training time	4h 37min	4h 45min
Init. Memory Usage	15.08 GB	16.93 GB
Fine-tuning Memory Usage	23.27 GB	23.17 GB

E.2 Memory Footprint and Training Time

From Table 4, SDC-LoRA introduces only a marginal training overhead compared with standard LoRA: the total training time increases from 4h37min to 4h45min (about 3% slower). The initial memory usage of SDC-LoRA is slightly higher (16.93GB vs. 15.08GB), reflecting the extra bookkeeping for computing one-step gradient G, but the peak memory during fine-tuning is essentially unchanged (23.17GB vs. 23.27GB). Overall, SDC-LoRA provides its forgetting mitigation with negligible additional compute and memory cost.

F Hyperparameters

We have detailed the hyperparameters required for fine-tuning LLaMA3-8B and Qwen2.5-7B using SDC-LoRA on the math reasoning tasks in Table 5.

We have detailed the hyperparameters required for fine-tuning LLaMA3-8B and Qwen2.5-7B using SDC-LoRA on the code generation tasks in Table 6.

Table 5: Hyperparameters for the SDC-LoRA run on MetaMath with LLaMA-3.1-8B-Instruct and Qwen-2.5-7B-Chat.

Hyperparameter	**Value**
Epochs	1
Max sequence length	256
Global batch size	16
Learning rate	1×10^{-4}
Optimizer	AdamW
Rank r	16
Target modules	Q, K, V, O, Up, Down
LoRA α	32
LoRA dropout	0.05
RS-LoRA	True
Principle surpressing dimension	256

Table 6: Hyperparameters for the SDC-LoRA run on Code-Feedback with LLaMA-3.1-8B-Instruct and Qwen-2.5-7B-Chat.

Hyperparameter	**Value**
Epochs	1
Max sequence length	512
Global batch size	16
Learning rate	5×10^{-4}
Optimizer	AdamW
Rank r	16
Target modules	Q, K, V, O, Up, Down
LoRA α	32
LoRA dropout	0.05
RS-LoRA	True
Principle surpressing dimension	256

G Usage of LLMs

In this paper, LLMs were used for coding assistance and writing support. Specifically, they were employed to optimize our runing code. For writing support, they were primarily utilized for proofreading the text and formatting LaTeX code.