RCA: FISHER-RESIDUAL LOW-RANK ADAPTATION FOR RLVR REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement Learning with Verifiable Rewards (RLVR) has become an effective post-training paradigm for improving language-model reasoning. Existing RLVR work has primarily studied reward sources, rollout sampling, KL regularization, etc. In this paper, we study a complementary question: how does RLVR act on the local parameter space of an adapted model, and which adapter update directions provide higher reward improvement per unit policy-change cost? We approach this question through the local Fisher geometry induced by KL-aware RLVR. Our results show that, under a reward–curvature separation condition, updates in the bilateral Fisher-residual channel space can achieve higher reward-per-curvature efficiency than equal-rank selectors that include high-curvature modes. Inspired by this conclusion, we introduce **RCA (Residual-Channel Low-Rank Adaptation)**, a Fisher-guided Low Rank Adapter (LoRA) method for RLVR reasoning. RCA first estimates layerwise Fisher input and output channels with a calibration-time Kronecker-Factored Approximate Curvature (KFAC) approximation. The resulting projectors define a residual-channel low-rank update constraint. Empirically, RCA improves reasoning reliability across pass@1–pass@128 compared with full-finetuning RL and representative LoRA variants, while maintaining a LoRA-sized trainable parameter footprint.

1 INTRODUCTION

Large language model post-training is increasingly organized around the question of how to convert latent capability into reliable reasoning behavior. Reinforcement Learning with Verifiable Rewards (RLVR) (Wen et al., 2025) is particularly attractive because the reward signal can often be obtained without a learned preference model. Most existing RLVR reasoning work focuses on the ingredients of the RL procedure: how to obtain verifiable rewards, sample rollouts, estimate advantages, design the policy-optimization objective, regularize against a reference policy, etc (Cobbe et al., 2021; Shao et al., 2024; Havrilla et al., 2024). These choices determine the scalar objective optimized during RLVR. They leave open a complementary parameter-space question: *which adapter update directions are efficient carriers of reward improvement under the local policy-change geometry?*

This question is especially important for parameter-efficient fine-tuning (PEFT). Low Rank Adapters (LoRA) freezes the backbone and learns a low-rank additive update (Hu et al., 2021; Schulman & Lab, 2025). This makes RLVR cheaper, but low rank alone does not determine how strongly an update changes the policy distribution. Meanwhile, directions with large Fisher curvature are expensive: a small parameter displacement can cause a large distributional change. Directions with smaller curvature can move farther under the same local KL cost. Thus, for RLVR, parameter efficiency is not only a question of *how many* parameters are trained, but also *which Fisher-sensitive directions* those parameters are allowed to modify.

Existing Fisher-based continual learning protects parameters and subspaces important to previously learned behavior (Kirkpatrick et al., 2017; Saha et al., 2021), while recent RLVR work analyzes an implicit off-principal update bias (Zhu et al., 2025). These perspectives motivate a reward-sensitive adapter-design question: when does excluding dominant Fisher channels improve the reward attainable within a fixed local KL budget? Curvature alone cannot settle this comparison, because the excluded directions may also carry reward signal. Therefore, in this work, we establish explicit sufficient conditions under which bilateral Fisher-residual support outperforms equal-dimensional

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.

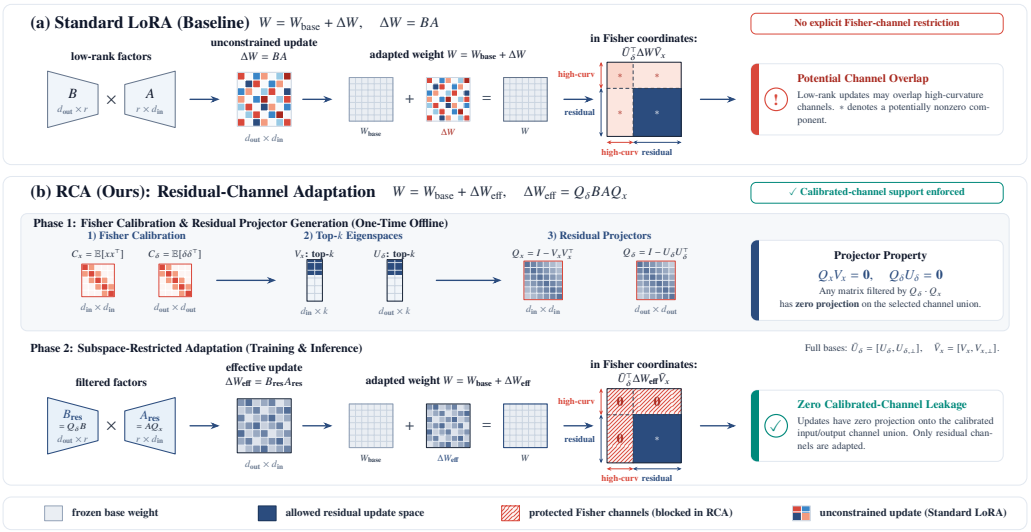


Figure 1: **RCA for PEFT-RLVR.** KL-aware RLVR induces a local Fisher geometry over parameter updates. Standard LoRA is parameter-efficient, but its low-rank update can still modify high-curvature policy-sensitive channels. RCA first identifies high-curvature input and output channels with a Fisher estimate, then restricts the adapter path to their bilateral residual complement.

spectral selectors admitting blocked modes, and quantify how much error in the factorized Fisher approximation this advantage tolerates.

While this residual channel provides a principled residual-support constraint, it does not inherently guarantee parameter efficiency. Therefore, we introduce **Residual-Channel Adaptation (RCA)**, illustrated in Figure 1, which synthesizes Fisher-guided residual-channel selection with LoRA factorization. In RCA, Kronecker-Factored Approximate Curvature (KFAC) serves as a calibration-time geometry sensor to estimate the high-curvature input and output channels. LoRA is then used to ensure that the resulting residual update remains low-rank and parameter-efficient.

To instantiate this idea, RCA separates geometry estimation from policy optimization. Before RLVR training, it runs a calibration pass on representative prompts and responses. For each adapted layer, it collects input activations and output score signals and estimates the dominant Fisher input and output channels using KFAC-style factors. These channels define the high-curvature directions to avoid. During RLVR, RCA trains ordinary LoRA factors, but the effective adapter update is projected into the bilateral Fisher-residual channel space. This gives a simple implementation: the RLVR trainer remains unchanged, while the adapter update is constrained to a low-rank residual-channel family. We empirically evaluate RCA across reasoning reliability curves and geometry audits, comparing against full-model RLVR, standard LoRA, and representative LoRA variants.

2 Preliminaries

We formalize KL-aware RLVR, the local Fisher geometry of small parameter updates, the exact layerwise Fisher block for a linear layer, and LoRA as a low-rank adapter family.

KL-aware RLVR. We consider a stochastic policy π_θ parameterized by θ, which generates a response $y = (y_1, \ldots, y_T)$ given a prompt q. In RLVR, the objective is to maximize the expected return based on a verifiable reward function $R(q, y) \in \{0, 1\}$ (e.g., unit test passage), subject to a regularization constraint that prevents deviation from a reference policy π_{ref} (usually the supervised fine-tuned model). The reward objective is $J_R(\theta) := \mathbb{E}_{q\sim\mathcal{D}}\mathbb{E}_{y\sim\pi_\theta(\cdot|q)}\big[R(q, y)\big]$, where $\mathcal{D}$ denotes the prompt distribution. Practical policy optimization also controls deviation from the reference model. We write the average reference KL as $K(\theta; \theta_{\text{ref}}) := \mathbb{E}_{q\sim\mathcal{D}} D_{\text{KL}}(\pi_\theta(\cdot \mid q) \,\|\, \pi_{\text{ref}}(\cdot \mid q))$. The KL-aware form of RLVR is therefore

$$\max_\theta \; J_R(\theta) \qquad \text{s.t.} \qquad K(\theta; \theta_{\text{ref}}) \le \tau, \tag{1}$$

where $\tau > 0$ is the allowed policy-change budget. PPO-style and GRPO-style implementations typically optimize a Lagrangian or clipped surrogate approximation to this trust-region problem (Schulman et al., 2017; Shao et al., 2024).

Local Fisher geometry. Let $u = \theta - \theta_{\mathrm{ref}}$ be a small parameter update, and let $j := \nabla_\theta J_R(\theta_{\mathrm{ref}})$ be the reward-gradient vector at the reference policy. Let F_θ be the Fisher information matrix of the reference policy,

$$F_\theta := \mathbb{E}_{q\sim\mathcal{D}}\mathbb{E}_{y\sim\pi_{\mathrm{ref}}(\cdot|q)}\left[\nabla_\theta \log \pi_{\mathrm{ref}}(y \mid q)\nabla_\theta \log \pi_{\mathrm{ref}}(y \mid q)^\top\right]. \tag{2}$$

Under the usual differentiability assumptions used in natural-gradient and trust-region analyses (Amari, 1998; Schulman et al., 2015), the reward and KL admit the local expansions

$$J_R(\theta_{\mathrm{ref}} + u) - J_R(\theta_{\mathrm{ref}}) = j^\top u + o(\|u\|), \qquad K(\theta_{\mathrm{ref}} + u; \theta_{\mathrm{ref}}) = \frac{1}{2}u^\top F_\theta u + o(\|u\|^2). \tag{3}$$

Thus the small-step RLVR problem is a linear reward-gain objective constrained by a quadratic Fisher cost.

LoRA as a restricted update family. LoRA freezes W_0 and learns a low-rank additive update (Hu et al., 2021)

$$\Delta W = BA, \qquad B \in \mathbb{R}^{d_{\mathrm{out}}\times r}, \qquad A \in \mathbb{R}^{r\times d_{\mathrm{in}}}, \tag{4}$$

where $r \ll \min(d_{\mathrm{in}}, d_{\mathrm{out}})$. This parameterization controls the number of trainable parameters and bounds the update rank. But it does not specify which Fisher-sensitive input or output channels the update uses.

3 Fisher-Efficient Residual Update Spaces

We now analyze which adapter update spaces are efficient under the local Fisher geometry induced by KL-aware RLVR. We first derive a reward-per-curvature criterion for restricted updates.

3.1 Small-KL One-Step Improvement and the KL-Efficiency Functional

Fix one adapted linear layer. Let the layerwise objective contribution be

$$\mathcal{J}(W) := \mathbb{E}_{q,y\sim\pi_W}[R(q, y)],$$

and define the gradient at W_0 by

$$G := \nabla_W \mathcal{J}(W_0) \in \mathbb{R}^{d_{\mathrm{out}}\times d_{\mathrm{in}}}, \qquad g := \mathrm{vec}(G) \in \mathbb{R}^d, \tag{5}$$

where $d = d_{\mathrm{out}}d_{\mathrm{in}}$. Let $F \in \mathbb{R}^{d\times d}$ denote the exact layerwise Fisher information matrix at the reference policy.

Assume F is symmetric positive definite[1]. We focus on the local regime where the trust-region radius is sufficiently small for the second-order Taylor expansion of the KL divergence and the first-order expansion of the objective to be valid. Specifically,

$$\mathrm{KL}\big(\pi_{W_0+\Delta W} \,\|\, \pi_{W_0}\big) = \frac{1}{2}\,\mathrm{vec}(\Delta W)^\top F\,\mathrm{vec}(\Delta W) + o\big(\|\Delta W\|_F^2\big), \tag{6}$$

$$\mathcal{J}(W_0 + \Delta W) - \mathcal{J}(W_0) = \langle G, \Delta W\rangle + o(\|\Delta W\|_F). \tag{7}$$

For any linear update restriction defined by a Euclidean projector P in $\mathbb{R}^d$, define the *KL-efficiency functional* $\Gamma_F(P) := g^\top(PFP)^\dagger g$. This quantity measures the leading-order reward improvement achievable per unit local Fisher/KL cost inside the restricted update space.

Lemma 3.1 (KL-limited one-step gain under a linear restriction)**.** *Fix any Euclidean projector P and a KL budget $\delta > 0$. Then, under* (6)–(7)*,*

$$\max_{\substack{v\in\mathrm{range}(P)\\ v^\top Fv\le 2\delta}} g^\top v = \sqrt{2\delta}\,\sqrt{\Gamma_F(P)}, \tag{8}$$

where range *denotes the range of a matrix.*

[1]This assumption is made for analytic convenience (Fatkhullin et al., 2023); the arguments below extend in a standard way to the degenerate case via pseudoinverses (Li & Yeh, 2012).

Lemma 3.1 shows that, in the small-KL regime, restricted adapter spaces should be compared by $\Gamma_F(P)$, not by Euclidean dimension, rank, or gradient norm alone.

3.2 Channel-Pair Decomposition under a Factorized Fisher Metric

To expose the channel structure of the Fisher cost, consider a factorized layerwise Fisher proxy

$$F_0 := C_x \otimes C_\delta, \tag{9}$$

where $C_x \in \mathbb{R}^{d_{\text{in}} \times d_{\text{in}}}$ is an input-side second-moment factor and $C_\delta \in \mathbb{R}^{d_{\text{out}} \times d_{\text{out}}}$ is an output-side score factor. Define eigendecompositions of these factors as

$$C_\delta = U_\delta \Lambda^\delta U_\delta^\top, \qquad C_x = V_x \Lambda^x V_x^\top, \tag{10}$$

with eigenvalues $\lambda_1^\delta \geq \lambda_2^\delta \geq \cdots \geq 0$ and $\lambda_1^x \geq \lambda_2^x \geq \cdots \geq 0$. The columns u_i of U_δ are output Fisher channels and the columns v_j of V_x are input Fisher channels. Their outer products and vectorizations $M_{ij} := u_i v_j^\top$, $f_{ij} := \text{vec}(M_{ij})$ form an orthonormal basis for the layer-update space. The factorized Fisher proxy $F_0 = C_x \otimes C_\delta$ is diagonal in this basis:

$$F_0 f_{ij} = \lambda_i^\delta \lambda_j^x f_{ij}, \qquad \lambda_{ij} := \lambda_i^\delta \lambda_j^x. \tag{11}$$

Proposition 3.2 (Channel-pair decomposition of local Fisher cost)**.** *For any layer update $D \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, write its factorized-Fisher-basis coordinates as $Z := U_\delta^\top D V_x$, $Z_{ij} = \langle D, u_i v_j^\top \rangle_F$. Then the factorized Fisher local KL cost is*

$$\frac{1}{2}\text{vec}(D)^\top (C_x \otimes C_\delta)\text{vec}(D) = \frac{1}{2}\sum_{i=1}^{d_{\text{out}}}\sum_{j=1}^{d_{\text{in}}} \lambda_i^\delta \lambda_j^x \, Z_{ij}^2. \tag{12}$$

Thus the Fisher cost of channel-pair mode $u_i v_j^\top$ is the product $\lambda_i^\delta \lambda_j^x$.

A large λ_i^δ means that changing output channel u_i strongly changes the policy distribution. A large λ_j^x means that the input channel v_j is frequently and strongly active in policy-sensitive contexts. The factorized Fisher metric therefore assigns a local KL price $\lambda_i^\delta \lambda_j^x$ to each output–input channel pair.

3.3 Removing Leading Fisher Channels Maximizes Blocked Curvature

We next identify which bilateral channel projectors remove the largest amount of Fisher curvature. Fix the avoided high-curvature ranks r_k for both input and output sides. The output-side projector P_δ is built from the leading eigenvectors of C_δ, and the input-side projector P_x is built from the leading eigenvectors of C_x:

$$P_\delta := U_{\delta,1:r_k} U_{\delta,1:r_k}^\top, \quad P_x := V_{x,1:r_k} V_{x,1:r_k}^\top, \quad Q_\delta := I - P_\delta, \quad Q_x := I - P_x. \tag{13}$$

Consider the bilateral residual-channel constraint

$$\Delta W = Q_\delta \Delta W Q_x. \tag{14}$$

This constraint removes update components that touch the leading output or input Fisher channels.

Lemma 3.3 (Residual-channel update constraint)**.** *If a layer update satisfies $\Delta W = Q_\delta \Delta W Q_x$, then upon vectorization,*

$$\text{vec}(\Delta W) = P_{\text{O}} \, \text{vec}(\Delta W), \qquad P_{\text{O}} := Q_x \otimes Q_\delta. \tag{15}$$

Moreover, P_{O} is a Euclidean projector onto the bilateral residual-channel update space.

The complementary blocked-union projector is $P_{\text{blk}} := I - P_{\text{O}}$.

Theorem 3.4 (Leading Fisher channels maximize blocked curvature trace)**.** *Among all output projectors $\widetilde{P}_\delta$ and input projectors $\widetilde{P}_x$ of rank r_k, the choice P_δ and P_x formed by the leading Fisher eigenvectors maximizes the factorized-Fisher curvature trace captured by the blocked-union subspace:*

$$\text{tr}\left[\widetilde{P}_{\text{blk}}(C_x \otimes C_\delta)\right], \qquad \widetilde{P}_{\text{blk}} = \widetilde{P}_x \otimes I + I \otimes \widetilde{P}_\delta - \widetilde{P}_x \otimes \widetilde{P}_\delta. \tag{16}$$

Equivalently, leading Fisher channels are the equal-rank bilateral channel projectors that remove the largest amount of local Fisher curvature.

Theorem 3.4 shows that the leading input and output eigenspaces of the factorized Fisher metric are the equal-rank bilateral channel projectors that remove the largest amount of local curvature. Removing them forces the update to search in directions where parameter displacement is less costly in policy space.

3.4 Reward–Curvature Separation in the Fisher Channel Basis

Removing high-curvature modes is useful only if the reward-improving gradient signal is not exclusively locked inside them. We therefore impose a reward–curvature separation condition in the factorized Fisher channel basis. Let $g_{ij} := f_{ij}^\top g$. The bilateral projection blocks the union of high-curvature left or right channels,

$$\mathcal{I}_{\mathrm{blk}} = \{(i,j) : i \le r_k \text{ or } j \le r_k\}, \tag{17}$$

and allows the residual-residual set $\mathcal{I}_{\mathrm{res}} = \{(i,j) : i > r_k,\ j > r_k\}$. For each mode, define its *mode reward–curvature efficiency*

$$\rho_{ij} := g_{ij}^2/\lambda_{ij}.$$

This metric measures reward-gradient strength relative to local Fisher price.

Assumption 3.5 (Curvature–gradient separation)**.** There exist constants $\kappa > 1$ and $C \ge 1$ such that, for any blocked mode $b \in \mathcal{I}_{\mathrm{blk}}$ and any residual mode $r \in \mathcal{I}_{\mathrm{res}}$, the following hold:

$$\lambda_b \ge \kappa\, \lambda_r, \tag{18}$$

i.e., blocked modes incur higher Fisher curvature, and

$$g_b^2 \le C\, g_r^2, \tag{19}$$

meaning that the reward-improving gradient signal is not overly concentrated on those high-curvature directions.

Define the ideal margin as the difference in reward–curvature efficiency between the worst residual mode and the best blocked mode:

$$\alpha := \inf_{r \in \mathcal{I}_{\mathrm{res}},\, b \in \mathcal{I}_{\mathrm{blk}}} (\rho_r - \rho_b) = \inf_{r \in \mathcal{I}_{\mathrm{res}},\, b \in \mathcal{I}_{\mathrm{blk}}} \left(\frac{g_r^2}{\lambda_r} - \frac{g_b^2}{\lambda_b} \right). \tag{20}$$

Lemma 3.6 (Positivity of the ideal margin)**.** *Under Assumption 3.5, if $C < \kappa$ and $\inf_{r \in \mathcal{I}_{\mathrm{res}}} \rho_r > 0$, then $\alpha > 0$.*

This assumption is intentionally weaker than requiring the reward gradient to vanish in blocked directions. It simply states that after dividing by Fisher cost, residual modes are more attractive uses of a small KL budget. To account for factorized-Fisher approximation error, we assume the true Fisher is a relative perturbation of the proxy:

$$F = F_0 + E, \qquad \varepsilon := \|F_0^{-1/2} E F_0^{-1/2}\|_{\mathrm{op}} < 1. \tag{21}$$

Theorem 3.7 (Residual-space efficiency under true-Fisher perturbation)**.** *Under Assumption 3.5, if $\alpha > 0$, let P be any spectral-selector projector in the factorized Fisher basis $\{f_{ij}\}$ with $\mathrm{rank}(P) = \mathrm{rank}(P_{\mathrm{O}})$ and $t = |\mathcal{I}(P) \cap \mathcal{I}_{\mathrm{blk}}|$ be the number of blocked modes selected by P. Then if $t \ge 1$,*

$$\Gamma_F(P_{\mathrm{O}}) - \Gamma_F(P) \ge \frac{t\alpha}{1+\varepsilon} - \frac{2\varepsilon}{(1-\varepsilon)(1+\varepsilon)} \Gamma_{F_0}(P), \tag{22}$$

where $\Gamma_{F_0}(P) := g^\top (P F_0 P)^\dagger g$. In particular, $\Gamma_F(P_{\mathrm{O}}) > \Gamma_F(P)$ whenever

$$t\alpha > \frac{2\varepsilon}{1-\varepsilon} \Gamma_{F_0}(P). \tag{23}$$

Theorem 3.7 establishes the main geometric conclusion of the theory: when reward signal is not overly concentrated in high-curvature modes and the factorized Fisher proxy is sufficiently accurate, the bilateral residual-channel projector $P_{\mathrm{O}} = Q_x \otimes Q_\delta$ can achieve strictly larger reward-per-curvature efficiency than equal-rank selectors that include blocked high-curvature modes. Thus, the theory identifies $\Delta W = Q_\delta \Delta W Q_x$ as the preferred residual update space.

4 RCA: Residual-Channel Adaptation

The theory identifies a Fisher-residual update space of the form $\Delta W = Q_\delta \Delta W Q_x$, but it does not yet specify how to estimate Q_δ, Q_x in a large language model, nor how to parameterize the residual update efficiently. In this section, we propose **Residual-Channel Adaptation (RCA)**, a parameter-efficient method that combines Fisher-based channel selection with low-rank parameterization.

4.1 KFAC Calibration of Fisher Channels

For each adapted layer ℓ, let $W_{0,\ell} \in \mathbb{R}^{d_{\text{out},\ell} \times d_{\text{in},\ell}}$ be the frozen weight matrix. For every selected token, we record $x_{\ell,t} \in \mathbb{R}^{d_{\text{in},\ell}}$, $\delta_{\ell,t} \in \mathbb{R}^{d_{\text{out},\ell}}$, where $x_{\ell,t}$ is the input activation to layer ℓ, and $\delta_{\ell,t}$ is the derivative of the calibration log-probability objective with respect to the output of that layer. If the implementation backpropagates a negative log-likelihood loss, the sign of $\delta_{\ell,t}$ is reversed, but the Fisher factor $\delta_{\ell,t}\delta_{\ell,t}^\top$ is unchanged.

Given N_ℓ selected token instances for layer ℓ, the empirical KFAC factors are

$$\widehat{C}_{x,\ell} = \frac{1}{N_\ell} \sum_{t=1}^{N_\ell} x_{\ell,t} x_{\ell,t}^\top, \qquad \widehat{C}_{\delta,\ell} = \frac{1}{N_\ell} \sum_{t=1}^{N_\ell} \delta_{\ell,t} \delta_{\ell,t}^\top. \tag{24}$$

This is precisely the KFAC approximation used by RCA: $F_{0,\ell} := \widehat{C}_{x,\ell} \otimes \widehat{C}_{\delta,\ell}$, in which $\widehat{C}_{x,\ell}$ measures which input channels are active in contexts where the policy is being scored. $\widehat{C}_{\delta,\ell}$ measures which output channels the token log-probability is sensitive to. Their leading eigenspaces are therefore the layerwise channels that dominate the empirical KL geometry. On the other hand, RCA only needs the top k_x and k_δ eigenspaces. We estimate them with randomized range finding (Halko et al., 2011). The detailed estimation algorithm and streaming update formulas are deferred to Appendix C.

4.2 Residual-Channel LoRA Parameterization

After calibration, each layer has high-curvature input and output bases $V_{x,\ell}$ and $U_{\delta,\ell}$. These bases define the high-curvature projectors

$$P_{x,\ell} := V_{x,\ell} V_{x,\ell}^\top, \qquad P_{\delta,\ell} := U_{\delta,\ell} U_{\delta,\ell}^\top, \tag{25}$$

and residual projectors

$$Q_{x,\ell} := I - P_{x,\ell}, \qquad Q_{\delta,\ell} := I - P_{\delta,\ell}. \tag{26}$$

A standard LoRA layer learns $\Delta W_\ell = B_\ell A_\ell$ as defined in (4). RCA instead parameterizes the Fisher-residual update by placing the LoRA factors inside the calibrated residual channel projectors:

$$\Delta W_{\ell,\text{eff}} = Q_{\delta,\ell} B_\ell A_\ell Q_{x,\ell}. \tag{27}$$

Thus the adapter path first removes high-curvature input-channel components, applies a low-rank update in the remaining residual coordinates, and then removes high-curvature output-channel components before adding the adapter output to the frozen backbone output.

4.3 Projection After Optimizer Steps

The forward pass can explicitly apply $Q_{\delta,\ell}$ and $Q_{x,\ell}$ as in (27). In implementation, we additionally enforce the same residual constraint on the saved LoRA factors by post-step projection: $B_\ell \leftarrow Q_{\delta,\ell} B_\ell,\ A_\ell \leftarrow A_\ell Q_{x,\ell}$, so that $B_\ell A_\ell$ can always lie in the residual-residual space. The overall procedure is summarized in Appendix C.2. And Appendices F.1 and F.3 examine the corresponding proxy efficiencies and compare RCA with curvature-aware alternatives.

5 Experiments

We evaluate whether actively selecting low-curvature but high KL-efficient Fisher/KFAC bases improves pass@k reasoning reliability and whether the gain is attributable to meaningful geometry rather than the shared hard-projection executor. Our code is available at an anonymous link `https://anonymous.4open.science/r/RCA-RLVR-4FB3/README.md`.

5.1 Experimental Setup

We train with the *VERL* framework (Sheng et al., 2024) on the `DeepScaleR` (Luo et al., 2025) corpus using a GRPO-style on-policy recipe with verifiable math rewards. Rewards are produced by a rule-based checker that extracts the final `\boxed{\cdot}` answer and tests numeric/rational equivalence. For all LoRA and its variants, including RCA, the LoRA rank is set to $r = 32$. Unless

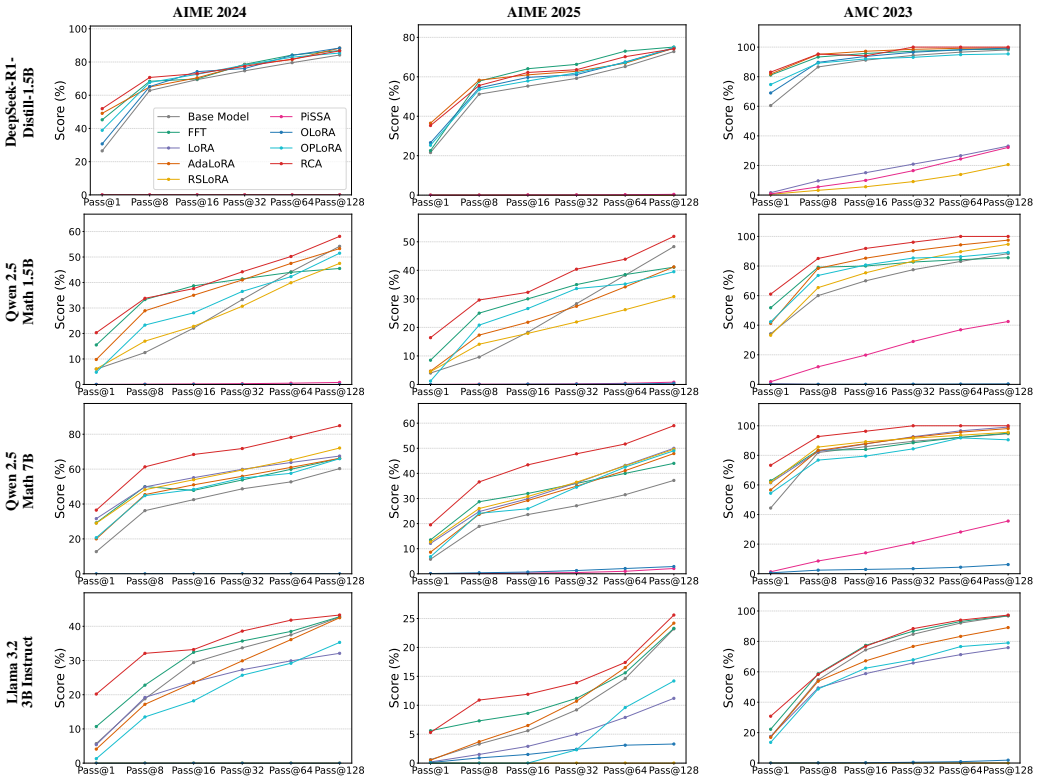


Figure 2: **Main pass@k results.** Curves compare the base model, FFT, LoRA variants, and RCA.

otherwise specified, all PEFT methods use the same LoRA configuration, RLVR trainer, optimizer, projection frequency, checkpoint format, and evaluation protocol. Training uses 8×A100 40GB GPU workers for the reported RLVR runs. Pass@k summaries are computed from the local evaluation result trees using the same programmatic checker and prompt format for each local comparison. More details appear in Appendix E.

Baselines. We compare RCA against the PEFT and full finetuning baselines:

- **Full Finetuning (FFT)**: A strong baseline that fully trains the model using GRPO.
- **LoRA** (Hu et al., 2021): The standard low-rank adaptation baseline.
- **RSLoRA** (Kalajdzievski, 2023): A variant that scales adapters by $1/\sqrt{r}$ to stabilize learning dynamics.
- **AdaLoRA** (Zhang et al., 2023): Adapts the rank r across layers based on sensitivity-based importance scores.
- **PiSSA** (Meng et al., 2024): Initializes adapters with the principal components of W_0, encouraging updates within high-energy static directions.
- **OLoRA** (Büyükakyüz, 2024): Utilizes QR decomposition to initialize adapters as orthonormal matrices for faster convergence.
- **OPLoRA:** (Xiong & Xie, 2025) An off-principal LoRA baseline for preventing catastrophic forgetting in continual learning.

5.2 Main Results

Figure 2 emphasizes both reliability and stability across a diverse set of backbones. Several unconstrained or initialization-only PEFT baselines collapse, most visibly for standard LoRA which flatlines at near-zero pass rates. This phenomenon is consistent with recent work showing that LoRA performs

poorly in RLVR scenarios (Yin et al., 2025). On the DeepSeek-R1-Distill-Qwen-1.5B backbone, FFT struggles on AIME 2025, falling below the base model and settling between RSLoRA and RCA. Even when methods initially exceed the base model, they often suffer from severe decay at higher k; yet RCA successfully preserves and scales performance well above the base model. Similarly, on Qwen2.5-Math-7B, FFT exhibits an anomalous arithmetic scaling behavior from pass@16 to pass@128, whereas RCA follows a robust and natural scaling trajectory. For Llama-3.2-3B-Instruct, FFT initially outperforms the base model at low k, but this advantage progressively shrinks, converging to near-zero difference by pass@128. RCA, by contrast, provides sustained improvements across the entire k-scaling spectrum.

To further verify the ability of RCA on general knowledge domain, we evaluate the Llama-3.2-3B-Instruct models on a suite of non-math benchmarks, including MMLU, ARC-Challenge, HellaSwag, and GPQA. As shown in Table 1, standard LoRA collapses even compared with base model as the same of math benchmarks. RCA, on the other hand, fully preserves and even slightly improves general reasoning and knowledge capabilities. This demonstrates RCA's ability to seamlessly align models to new domains without sacrificing their foundational competencies.

Table 1: **Llama-3.2-3B-Instruct on general knowledge domain pass@k audit.** The generated-answer retention table over MMLU, ARC-Challenge, HellaSwag, and GPQA.

Method	**MMLU**						**ARC-C**					
	p@1	p@8	p@16	p@32	p@64	p@128	p@1	p@8	p@16	p@32	p@64	p@128
Base	60.2	88.4	95.3	96.0	98.0	99.3	73.3	88.8	94.9	96.0	97.2	98.9
FFT	**62.1**	87.8	93.8	96.5	98.1	98.5	74.6	89.3	92.6	96.6	97.3	98.8
LoRA	49.5	69.4	80.1	82.6	86.4	88.1	44.0	56.7	80.6	82.1	85.4	86.8
RCA	**62.1**	**90.1**	**96.4**	**98.5**	**100.0**	**100.0**	**75.3**	**89.9**	**95.3**	**98.5**	**98.5**	**100.0**
Method	**HellaSwag**						**GPQA**					
	p@1	p@8	p@16	p@32	p@64	p@128	p@1	p@8	p@16	p@32	p@64	p@128
Base	54.5	91.2	97.8	98.8	99.0	99.0	23.5	83.0	93.6	97.3	98.6	99.0
FFT	56.1	**92.1**	98.1	98.2	98.8	99.0	28.3	80.6	93.0	98.0	98.9	99.5
LoRA	32.9	39.9	46.2	48.1	48.4	50.7	19.3	27.5	37.3	45.4	47.0	48.1
RCA	**59.1**	91.6	**98.3**	**99.1**	**99.1**	**100.0**	**29.7**	**83.6**	**95.3**	**99.5**	**100.0**	**100.0**

5.3 Ablation Study: Principal Rank r_k

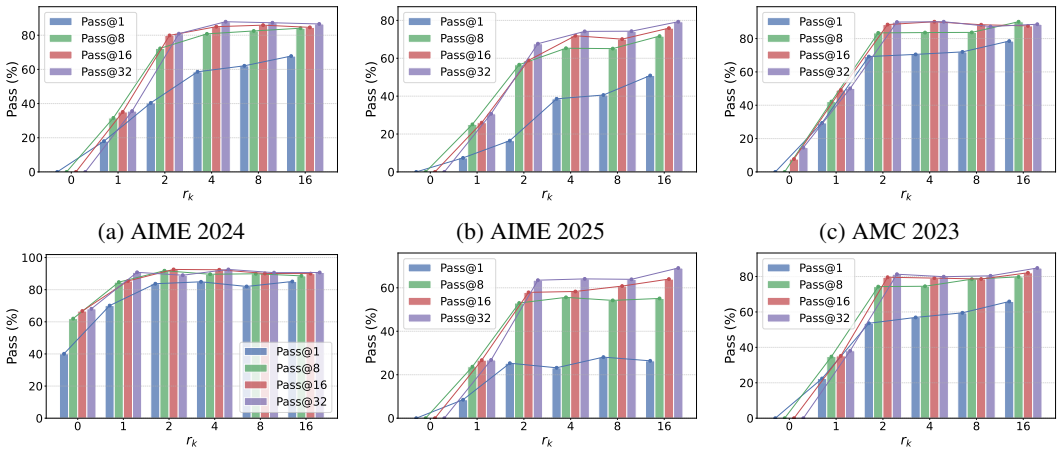
We next vary the protected rank r_k, which dictates the number of high-curvature channels blocked from the trainable residual update on each side of the adapted layer, as shown in Figure 3. Rank $r_k = 0$ corresponds to unconstrained LoRA geometry, which frequently collapses. Protecting even a single principal direction ($r_k = 1$) recovers substantial performance, placing it midway to a fully robust model. Expanding the protected subspace further ($r_k = 2$ and beyond) yields consistent and monotonic gains across datasets. This monotonic scaling validates our core hypothesis: severe interference localizes in the highest-curvature directions, and aggressively shielding them is necessary for reliable high-k scaling. We use $r_k = 16$ in the default RCA configuration because it provides a stable interior setting without fully exhausting the LoRA rank $r = 32$.

5.4 Ablation and Leakage Study

Recall in Section 3, for each adapted linear layer, the default RCA calibration collects input activations x and output-side backpropagated signals δ on a held-out calibration stream and forms the KFAC factors $C_x = \mathbb{E}[xx^\top]$ and $C_\delta = \mathbb{E}[\delta\delta^\top]$. The protected input and output subspaces are the leading r_k eigenspaces of C_x and C_δ. The executor then projects the LoRA factors so that the realized update $\Delta W = BA$ satisfies the bilateral residual constraint $\Delta W \leftarrow (I - P_\delta)\Delta W(I - P_x)$, where P_x and P_δ are the input- and output-side Fisher projectors. We audit realized leakage with

$$\eta_{\Delta W} = \frac{\|P_\delta \Delta W + \Delta W P_x - P_\delta \Delta W P_x\|_F^2}{\|\Delta W\|_F^2}, \tag{28}$$

which measures how much adapter energy re-enters the protected union after training. More results and analysis on this metric can be found in Appendix B.



(a) AIME 2024 (b) AIME 2025 (c) AMC 2023

(d) MATH500 (e) Minerva Math (f) OlympiadBench

Figure 3: **Effect of blocked rank r_k on DeepSeek-R1-Distill-Qwen-1.5B.** The training recipe is the same as Section 5.1. Rank $r_k = 0$ corresponds to unconstrained LoRA geometry.

Table 2: **Controls isolating the role of Fisher-aligned bases.** All variants use the same Qwen2.5-Math-1.5B RLVR recipe as Section 5.1; only the protected basis or projection side is changed.

Control	$\eta_{\Delta W} \downarrow$	**Reward Mean**	**Response Length**
RCA bilateral Fisher	1.06×10^{-15}	0.4512	833.2
RCA left-only	0.084435	0.0000	3072.0
RCA right-only	0.022062	0.2383	975.6
Random bilateral basis	-	0.0000	3072.0

Table 2 isolates the training-side effect of the basis and the bilateral projection. The random-basis row uses the same rank and hard projection code, but collapses to zero reward with max-length generations. Thus a strict rank-r_k projector is insufficient unless the protected subspace is correctly aligned with bilateral Fisher curvature. RCA's bilateral Fisher projection achieves near-machine-precision leakage (1.06×10^{-15}) and high reward. The one-sided controls show exactly why RCA requires a bilateral KFAC factorization rather than a single projector: output-side-only (left-only) protection leaks heavily and collapses, while input-side-only (right-only) protection remains trainable but still leaks significantly more than the bilateral default and achieves much lower reward. Thus, protecting both subspaces simultaneously is required to stabilize RLVR.

6 Conclusion

We studied parameter-efficient Reinforcement Learning with Veriable Rewards (RLVR) through the local Fisher geometry induced by KL-aware policy optimization. Our theoretical analysis shows that, under a reward–curvature separation condition and bounded factorized-Fisher approximation error, the bilateral residual-channel space can achieve higher reward-per-curvature efficiency than equal-rank selectors that include high-curvature Fisher modes. We introduced RCA, a residual-channel LoRA method that uses calibration-time KFAC estimates to identify dominant input and output Fisher channels, and then constrains the effective adapter update to their bilateral residual complement. Empirically, RCA improves reasoning reliability across pass@1–pass@128 compared with full-finetuning RL and representative LoRA variants, while preserving a LoRA-sized trainable parameter footprint. These results suggest that PEFT for RLVR should not be viewed only as a parameter-count problem, but also as an update-direction selection problem under the local KL geometry of the policy. Looking forward, while RCA demonstrates that appropriate Fisher approximations like KFAC can yield substantial gains for RLVR, exploring more advanced approximation techniques and integrating true natural gradient dynamics into the learning process remain highly promising directions for advancing reasoning capabilities in LLMs.

AI USE DISCLOSURE

We used AI tools to polish the language of the manuscript.

REFERENCES

Shun-Ichi Amari. Natural gradient works efficiently in learning. *Neural computation*, 10(2):251–276, 1998.

Kerim Büyükakyüz. Olora: Orthonormal low-rank adaptation of large language models. *arXiv preprint arXiv:2406.01775*, 2024.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. 2018.

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

Ilyas Fatkhullin, Anas Barakat, Anastasia Kireeva, and Niao He. Stochastic policy gradient methods: Improved sample complexity for fisher-non-degenerate policies. In *International Conference on Machine Learning*, pp. 9827–9869. PMLR, 2023.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645 (8081):633–638, September 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL http://dx.doi.org/10.1038/s41586-025-09422-z.

Nathan Halko, Per-Gunnar Martinsson, and Joel A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2): 217–288, 2011. doi: 10.1137/090771806. URL https://doi.org/10.1137/090771806.

Alex Havrilla, Yuqing Du, Sharath Chandra Raparthy, Christoforos Nalmpantis, Jane Dwivedi-Yu, Maksym Zhuravinskyi, Eric Hambro, Sainbayar Sukhbaatar, and Roberta Raileanu. Teaching large language models to reason with reinforcement learning. *arXiv preprint arXiv:2403.04642*, 2024.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. OlympiadBench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, August 2024.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

HuggingFaceH4. Aime 2024 (dataset card). Hugging Face, 2024. URL https://huggingface.co/datasets/HuggingFaceH4/aime_2024.

Damjan Kalajdzievski. A rank stabilization scaling factor for fine-tuning with lora. *arXiv preprint arXiv:2312.03732*, 2023.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, March 2017. ISSN 1091-6490. doi: 10.1073/pnas.1611835114. URL http://dx.doi.org/10.1073/pnas.1611835114.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay V. Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

Yen-Huan Li and Ping-Cheng Yeh. An interpretation of the moore-penrose generalized inverse of a singular fisher information matrix. *IEEE Transactions on Signal Processing*, 60(10):5532–5536, 2012.

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.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. In *Second Conference on Language Modeling*, 2025.

Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y. Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Li Erran Li, Raluca Ada Popa, and Ion Stoica. Deepscaler: Surpassing o1-preview with a 1.5b model by scaling rl. https://pretty-radio-b75.notion.site/DeepScaleR-Surpassing-O1-Preview-with-a-1-5B-Model-by-Scaling-RL-19681902c1468005bed8ca303013a4e2, 2025. Notion Blog.

James Martens and Roger Grosse. Optimizing neural networks with kronecker-factored approximate curvature. In *International conference on machine learning*, pp. 2408–2417, 2015.

math-ai. Amc 2023 (dataset card). Hugging Face, 2025. URL https://huggingface.co/datasets/math-ai/amc23.

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. Pissa: Principal singular values and singular vectors adaptation of large language models. *Advances in Neural Information Processing Systems*, 37: 121038–121072, 2024.

OpenCompass. Aime 2025 (dataset card). Hugging Face, 2025. URL https://huggingface.co/datasets/opencompass/AIME2025.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024.

Gobinda Saha, Isha Garg, and Kaushik Roy. Gradient projection memory for continual learning. In *International Conference on Learning Representations*, 2021.

John Schulman and Thinking Machines Lab. Lora without regret. *Thinking Machines Lab: Connectionism*, 2025. doi: 10.64434/tml.20250929. https://thinkingmachines.ai/blog/lora/.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pp. 1889–1897, 2015.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

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

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*, 2024.

Xumeng Wen, Zihan Liu, Shun Zheng, Zhijian Xu, Shengyu Ye, Zhirong Wu, Xiao Liang, Yang Wang, Junjie Li, Ziming Miao, Jiang Bian, and Mao Yang. Reinforcement learning with verifiable rewards implicitly incentivizes correct reasoning in base llms. *arXiv preprint arXiv:2506.14245*, 2025.

Yifeng Xiong and Xiaohui Xie. Oplora: Orthogonal projection lora prevents catastrophic forgetting during parameter-efficient fine-tuning. *arXiv preprint arXiv:2510.13003*, 2025.

Qingyu Yin, Yulun Wu, Zhennan Shen, Sunbowen Li, Zhilin Wang, Yanshu Li, Chak Tou Leong, Jiale Kang, and Jinjin Gu. Evaluating parameter efficient methods for rlvr. *arXiv preprint arXiv:2512.23165*, 2025.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, YuYue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, LingJun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 4791–4800, 2019.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adaptive budget allocation for parameter-efficient fine-tuning. In *The Eleventh International Conference on Learning Representations*, 2023.

Yaoyue Zheng, Yin Zhang, Joost van de Weijer, Gido M van de Ven, Shaoyi Du, Xuetao Zhang, and Zhiqiang Tian. Revisiting weight regularization for low-rank continual learning. In *The Fourteenth International Conference on Learning Representations*, 2026.

Hanqing Zhu, Zhenyu Zhang, Hanxian Huang, DiJia Su, Zechun Liu, Jiawei Zhao, Igor Fedorov, Hamed Pirsiavash, Jinwon Lee, David Z. Pan, Zhangyang Wang, Yuandong Tian, and Kai Sheng Tai. The path not taken: RLVR provably learns off the principals. In *NeurIPS 2025 Workshop on Efficient Reasoning*, 2025.

A Related Work

RL for LLM Reasoning. RLVR has recently become a particularly effective post-training paradigm because many reasoning domains admit programmatic or rule-based rewards. DeepSeekMath introduced GRPO as a critic-free alternative to PPO-style RL for mathematical reasoning (Shao et al., 2024), and DeepSeek-R1 further demonstrated that large-scale RL can elicit strong reasoning behavior with verifiable rewards (Guo et al., 2025). Follow-up systems and analyses such as DAPO and Dr. GRPO highlight that RLVR performance is sensitive to sampling, clipping, length normalization, rollout filtering, and other optimization details (Yu et al., 2025; Liu et al., 2025). These works primarily study the scalar RL objective, reward source, rollout distribution, and optimization recipe. RCA is complementary: it keeps the RLVR objective and trainer essentially unchanged, but asks which local adapter directions carry reward improvement efficiently under the KL/Fisher geometry induced by RLVR.

LoRA for LLM Post-Training. Full-model RLVR can be expensive and can also induce broad parameter movement, motivating parameter-efficient fine-tuning (PEFT) for post-training. LoRA freezes the pretrained weight and learns a low-rank additive update, reducing trainable parameters while preserving the original inference structure after merging (Hu et al., 2021). Many LoRA variants improve different aspects of this basic parameterization. RSLoRA modifies rank scaling for stability (Kalajdzievski, 2023); AdaLoRA reallocates rank budget across modules (Zhang et al., 2023). Other variants modify the initialization or basis of the adapter: OLoRA uses orthonormal initialization based on QR decomposition (Büyükakyüz, 2024), while PiSSA initializes or adapts principal singular components of pretrained weights (Meng et al., 2024). These methods improve the trainable parameterization, initialization, rank allocation, or memory footprint of LoRA. However, low rank by itself does not determine the KL cost of an update during RLVR. A rank-r adapter may still move high-curvature policy-sensitive channels. RCA addresses this missing geometric degree of freedom by combining LoRA with a Fisher-residual channel constraint.

Fisher Geometry and KFAC. Natural-gradient and trust-region methods motivate measuring parameter updates by the local KL/Fisher metric rather than by Euclidean distance (Amari, 1998; Schulman et al., 2015; 2017). KFAC makes this geometry tractable for neural networks by approximating layerwise Fisher blocks as Kronecker products of activation and output-gradient factors (Martens & Grosse, 2015). Fisher estimates have also been used as parameter-importance measures, for example in elastic weight consolidation, where important parameters are penalized to reduce catastrophic forgetting (Kirkpatrick et al., 2017). RCA uses Fisher information in a different way from both KFAC optimization and Fisher-penalty regularization, since it does not invert the KFAC matrix, precondition every gradient step, or add a diagonal quadratic penalty. Instead, KFAC-style factors are used once as calibration-time geometry sensors. Their leading eigenspaces identify high-curvature input and output channels, and RCA hard-projects the LoRA update into the corresponding residual-residual channel space. Thus the RLVR optimizer remains unchanged, while the adapter family is restricted to directions that are predicted to be cheaper under the local KL geometry.

B Fisher/KFAC Basis Geometry Controls

Table 3 gives the basis-geometry audit corresponding to the controls in Table 2. Each row changes only how the protected input and output bases are chosen; the downstream RCA executor remains the same bilateral residual projector whenever the basis is used for training.

Variant construction. *OPLoRA SVD basis* is the static spectral baseline: for each frozen linear weight W_0, it takes the leading left and right singular-vector subspaces of W_0 as the protected basis. This basis is independent of RLVR data, activations, and output-gradient signals, so it measures whether pretrained weight energy alone identifies the policy-sensitive channels that RCA should avoid. *RCA bilateral Fisher* is the default RCA construction. On a held-out RLVR calibration stream, it records layer inputs x and output-side score signals δ, estimates $C_x = \mathbb{E}[xx^\top]$ and $C_\delta = \mathbb{E}[\delta\delta^\top]$, and protects the top-$r_k$ eigenspaces of both factors. The LoRA update is then forced into the bilateral residual complement $(I - P_\delta)\Delta W(I - P_x)$. *RCA response KFAC* uses the same KFAC estimator but calibrates on response-conditioned teacher-forced tokens. This tests whether the protected channels remain stable when the token distribution includes generated-answer positions. *RCA score KFAC*

replaces the calibration loss signal with a score-function Fisher signal, so that δ more directly tracks $\nabla \log \pi$ directions used by the policy Fisher. *RCA RL-proxy KFAC* uses an RLVR-proxy calibration signal, weighting or selecting score directions according to the reward-bearing response stream, and therefore probes whether the protected bases change when the geometry sensor is made closer to the actual RL objective. *Random bilateral basis* samples same-rank random orthonormal input and output subspaces and applies the identical bilateral projection code. It is a negative control for the executor: if projection alone were sufficient, random bases would look competitive.

Geometry metrics. The capture metrics measure how much exact top-r_k Fisher-factor mass is recovered by a sketched or proposed protected basis:

$$\mathrm{capture}(Q, C) = \frac{\mathrm{tr}(Q^\top C Q)}{\sum_{i \le r_k} \lambda_i(C)}.$$

Here Q is the candidate protected basis and C is either the input factor C_x or the output-gradient factor C_δ. Thus C_x capture is the fraction of dominant activation-side Fisher mass retained by the protected input channels; physically, it asks whether the basis covers input directions that are frequently active in policy-sensitive contexts. C_δ capture is the analogous output-side quantity; it asks whether the basis covers directions whose perturbation strongly changes the model distribution. Higher values are better for both because RCA intends to identify high-curvature channels and keep trainable updates away from them. The product overlap is the separable bilateral proxy $\mathrm{overlap}(C_x) \times \mathrm{overlap}(C_\delta)$. It is important because a linear-layer update is bilinear in output and input channels: a basis that captures only one side still misses many high-curvature channel pairs. The trace ratio compares the factorized-Fisher trace blocked by the candidate bilateral union subspace against random same-size bases. A larger trace ratio means that the protected basis removes more local-KL curvature than a random projector with the same rank, which is the desired physical behavior for a residual-channel method.

Table 3: **Fisher/KFAC geometry audit for protected bases.** Values are Qwen2.5-Math-1.5B audits unless noted. The random-basis row uses same-size random orthonormal bases and has no separate trace-ratio probe in this audit.

Variant	C_x cap. ↑	C_δ cap. ↑	Prod. ovlp. ↑	Trace ratio ↑
OPLoRA SVD basis	0.0076	0.0040	0.0001	9.46
RCA bilateral Fisher	0.7804	0.6441	0.0490	20.41
RCA response KFAC	0.7385	0.6428	0.0382	15.62
RCA score KFAC	0.7385	0.6423	0.0381	15.60
RCA RL-proxy KFAC	0.7378	0.6307	0.0357	15.06
Random bilateral basis	0.0018	0.0015	0.0001	–

The geometry audit explains the control outcomes in the main text. RCA captures $103\times$ more input-factor Fisher mass and $161\times$ more output-factor Fisher mass than the OPLoRA weight-SVD basis, raising product overlap from 0.0001 to 0.0490. Since the RCA update is bilinear, this product-space improvement is the relevant diagnostic: if either factor is poorly aligned, the protected matrix subspace does not match the high-curvature Fisher product space. Response-, score-, and RL-proxy KFAC variants capture comparable factor mass and larger blocked trace ratios, so they are useful geometry controls. The random-basis row is the negative control: it has near-random factor capture and negligible product overlap, which is why the hard projection executor alone does not produce a stable training run.

C Implementation Details

C.1 Randomized Eigenspace Estimation

Directly storing or diagonalizing dense Fisher factors for every transformer linear layer is unnecessary. RCA only needs the top k_x and k_δ eigenspaces. We estimate them with randomized range finding (Halko et al., 2011). For each layer, draw Gaussian sketch matrices

$$\Omega_{x,\ell} \in \mathbb{R}^{d_{\mathrm{in},\ell} \times s}, \qquad \Omega_{\delta,\ell} \in \mathbb{R}^{d_{\mathrm{out},\ell} \times s},$$

where $s \geq \max(k_x, k_\delta)$ includes a small oversampling margin. During a streaming calibration pass, we accumulate covariance-vector products

$$Y_{x,\ell} \leftarrow Y_{x,\ell} + X_\ell^\top (X_\ell \Omega_{x,\ell}), \tag{29}$$

$$Y_{\delta,\ell} \leftarrow Y_{\delta,\ell} + S_\ell^\top (S_\ell \Omega_{\delta,\ell}), \tag{30}$$

where X_ℓ stacks calibration activations and S_ℓ stacks calibration output-score signals for a mini-batch. The column spaces of $Y_{x,\ell}$ and $Y_{\delta,\ell}$ approximate the dominant eigenspaces of $\widehat{C}_{x,\ell}$ and $\widehat{C}_{\delta,\ell}$. We first orthonormalize the sketches:

$$\widehat{Q}_{x,\ell} = \mathrm{orth}(Y_{x,\ell}), \qquad \widehat{Q}_{\delta,\ell} = \mathrm{orth}(Y_{\delta,\ell}).$$

Orthonormalization alone does not order the basis vectors by Fisher curvature. We therefore perform Rayleigh–Ritz extraction before truncating to the requested ranks. For each side $a \in \{x, \delta\}$, form the reduced symmetric matrix

$$H_{a,\ell} = \widehat{Q}_{a,\ell}^\top \widehat{C}_{a,\ell} \widehat{Q}_{a,\ell}, \qquad H_{a,\ell} = Z_{a,\ell} \Theta_{a,\ell} Z_{a,\ell}^\top,$$

where $Z_{a,\ell}$ is orthogonal and the eigenvalues in $\Theta_{a,\ell}$ are sorted in descending order. The reduced matrices can be accumulated in a second pass over the same fixed calibration samples: sum $(X_\ell \widehat{Q}_{x,\ell})^\top (X_\ell \widehat{Q}_{x,\ell})$ and $(S_\ell \widehat{Q}_{\delta,\ell})^\top (S_\ell \widehat{Q}_{\delta,\ell})$ over mini-batches and divide by N_ℓ. This avoids storing the dense Fisher factors. The resulting bases are

$$V_{x,\ell} = \widehat{Q}_{x,\ell} Z_{x,\ell}[:, 1:k_x], \qquad U_{\delta,\ell} = \widehat{Q}_{\delta,\ell} Z_{\delta,\ell}[:, 1:k_\delta].$$

These are the leading Ritz vectors within the sketched subspaces; their approximation to the full-factor leading eigenspaces depends on the quality of those subspaces. The final high-curvature bases to avoid are

$$V_{x,\ell} \in \mathbb{R}^{d_{\mathrm{in},\ell} \times k_x}, \qquad U_{\delta,\ell} \in \mathbb{R}^{d_{\mathrm{out},\ell} \times k_\delta}, \tag{31}$$

the top k_x and k_δ approximate eigenvectors of the input and output KFAC factors. In the default configuration we use $k_x = k_\delta = r_k$, but the method permits asymmetric ranks.

C.2 Pseudo Code

The pseudo codes of RCA are gathered in Algorithm 2 and Algorithm 1.

D Appendix: Proofs of Theoretical Results

D.1 Proofs for Theoretical Results in Section 3

Proof of Lemma 3.3. Assume $\Delta W = Q_\delta \Delta W Q_x$. Using the standard Kronecker product identity

$$\mathrm{vec}(AXB) = (B^\top \otimes A)\,\mathrm{vec}(X). \tag{32}$$

with $A = Q_\delta$, $X = \Delta W$, and $B = Q_x$, we obtain

$$\mathrm{vec}(Q_\delta \Delta W Q_x) = ((Q_x)^\top \otimes Q_\delta)\,\mathrm{vec}(\Delta W). \tag{33}$$

Because Q_x is symmetric ($(Q_x)^\top = Q_x$), (33) becomes

$$\mathrm{vec}(Q_\delta \Delta W Q_x) = (Q_x \otimes Q_\delta)\,\mathrm{vec}(\Delta W). \tag{34}$$

Together with (14), we obtain

$$\mathrm{vec}(\Delta W) = (Q_x \otimes Q_\delta)\,\mathrm{vec}(\Delta W). \tag{35}$$

Defining $P_{\mathrm{O}} := Q_x \otimes Q_\delta$ gives (15). Since Q_x and Q_δ are Euclidean projectors, their Kronecker product is symmetric and idempotent:

$$P_{\mathrm{O}}^\top = P_{\mathrm{O}}, \qquad P_{\mathrm{O}}^2 = (Q_x^2) \otimes (Q_\delta^2) = P_{\mathrm{O}}.$$

Therefore P_{O} is a Euclidean projector onto the bilateral residual-channel update space. □

Algorithm 1 RCA KFAC Calibration

Require: Frozen policy, calibration data $\mathcal{C}$, adapted layers $\mathcal{L}$, avoided ranks k_x, k_δ, sketch size s.
for each adapted layer $\ell \in \mathcal{L}$ **do**
  Draw sketch matrices $\Omega_{x,\ell} \in \mathbb{R}^{d_{\text{in},\ell} \times s}$ and $\Omega_{\delta,\ell} \in \mathbb{R}^{d_{\text{out},\ell} \times s}$.
  Initialize sketches $Y_{x,\ell} = 0$, $Y_{\delta,\ell} = 0$.
  Register forward hooks for layer inputs X_ℓ and backward hooks for output score signals S_ℓ.
end for
for each calibration batch from $\mathcal{C}$ **do**
  Run the calibration objective and backpropagate.
  for each adapted layer ℓ **do**
    $Y_{x,\ell} \leftarrow Y_{x,\ell} + X_\ell^\top (X_\ell \Omega_{x,\ell})$.
    $Y_{\delta,\ell} \leftarrow Y_{\delta,\ell} + S_\ell^\top (S_\ell \Omega_{\delta,\ell})$.
  end for
end for
for each adapted layer ℓ **do**
  $\widehat{Q}_{x,\ell} \leftarrow \text{orth}(Y_{x,\ell})$, $\widehat{Q}_{\delta,\ell} \leftarrow \text{orth}(Y_{\delta,\ell})$.
end for
Replay the same fixed calibration samples to accumulate $H_{a,\ell} = \widehat{Q}_{a,\ell}^\top \widehat{C}_{a,\ell} \widehat{Q}_{a,\ell}$ for each layer and side $a \in \{x, \delta\}$.
for each adapted layer ℓ **do**
  Diagonalize each $H_{a,\ell}$; order the eigenvectors $Z_{a,\ell}$ by descending eigenvalue.
  $V_{x,\ell} \leftarrow \widehat{Q}_{x,\ell} Z_{x,\ell}[:, 1 : k_x]$.
  $U_{\delta,\ell} \leftarrow \widehat{Q}_{\delta,\ell} Z_{\delta,\ell}[:, 1 : k_\delta]$.
  Store $U_{\delta,\ell}, V_{x,\ell}$ as frozen buffers.
end for

Algorithm 2 RCA RLVR System

Calibrate KFAC bases with Algorithm 1.
Attach LoRA adapters and store U_δ, V_x as non-trainable Fisher-channel buffers.
while not converged **do**
  Sample prompt batch and generate rollouts.
  Compute verifiable rewards and advantages.
  Update LoRA factors with the RLVR loss (e.g., GRPO).
  Project LoRA factors $B \leftarrow Q_\delta B$, $A \leftarrow A Q_x$.
end while

Proof of Theorem 3.4. We aim to find orthogonal projectors $\widetilde{P}_\delta$ (on the output space) and $\widetilde{P}_x$ (on the input space), each of rank r_k, that maximize the trace objective:

$$\max_{\widetilde{P}_\delta, \widetilde{P}_x} \text{tr}\left[\left(\widetilde{P}_x \otimes I + I \otimes \widetilde{P}_\delta - \widetilde{P}_x \otimes \widetilde{P}_\delta\right)(C_x \otimes C_\delta)\right]. \tag{36}$$

Let $\widetilde{Q}_\delta = I - \widetilde{P}_\delta$ and $\widetilde{Q}_x = I - \widetilde{P}_x$. Notice that the expression in the parenthesis is exactly the complement of the tensor product of the orthogonal complements:

$$\widetilde{P}_x \otimes I + I \otimes \widetilde{P}_\delta - \widetilde{P}_x \otimes \widetilde{P}_\delta = I \otimes I - \widetilde{Q}_x \otimes \widetilde{Q}_\delta. \tag{37}$$

Substituting this into the trace objective, we have:

$$\begin{aligned}\text{tr}\left[(I \otimes I - \widetilde{Q}_x \otimes \widetilde{Q}_\delta)(C_x \otimes C_\delta)\right] &= \text{tr}(C_x \otimes C_\delta) - \text{tr}\left[(\widetilde{Q}_x C_x) \otimes (\widetilde{Q}_\delta C_\delta)\right] \\ &= \text{tr}(C_x)\,\text{tr}(C_\delta) - \text{tr}(\widetilde{Q}_x C_x)\,\text{tr}(\widetilde{Q}_\delta C_\delta).\end{aligned} \tag{38}$$

Since C_x and C_δ are positive semi-definite covariance matrices, $\text{tr}(C_x)$ and $\text{tr}(C_\delta)$ are constants independent of the projectors. Moreover, the traces $\text{tr}(\widetilde{Q}_x C_x)$ and $\text{tr}(\widetilde{Q}_\delta C_\delta)$ are non-negative. Therefore, maximizing (38) is equivalent to solving two independent minimization problems:

$$\min_{\widetilde{Q}_x} \text{tr}(\widetilde{Q}_x C_x) \quad \text{and} \quad \min_{\widetilde{Q}_\delta} \text{tr}(\widetilde{Q}_\delta C_\delta), \tag{39}$$

where $\widetilde{Q}_x$ and $\widetilde{Q}_\delta$ are projectors of rank $d_{\text{in}} - r_k$ and $d_{\text{out}} - r_k$, respectively.

By Ky Fan's maximum principle (or the Poincaré separation theorem), the minimum of $\text{tr}(P^\perp M)$ over all rank-$(d - r_k)$ orthogonal projectors $P^\perp$ for a positive semi-definite matrix M is achieved precisely when $P^\perp$ projects onto the eigenspace spanned by the $d - r_k$ eigenvectors of M corresponding to its smallest eigenvalues. Equivalently, the complementary projector $P = I - P^\perp$ of rank r_k must project onto the eigenspace spanned by the r_k eigenvectors corresponding to the largest eigenvalues of M.

Applying this to $M = C_x$ and $M = C_\delta$ independently, we conclude that $\widetilde{P}_x$ and $\widetilde{P}_\delta$ must be the projectors formed by the top r_k eigenvectors of C_x and C_δ, respectively. These are exactly the leading Fisher channels P_x and P_δ defined in (13). □

Proof of Lemma 3.1. Let P be a Euclidean projector and consider the constrained optimization problem

$$\max_{v \in \text{range}(P)} \; g^\top v \quad \text{s.t.} \quad \frac{1}{2} v^\top F v \le \delta, \tag{40}$$

where $F \succ 0$.

Step 1: Remove the explicit subspace constraint. The constraint $v \in \text{range}(P)$ is equivalent to $Pv = v$. Thus (40) is equivalent to

$$\max_{v \in \mathbb{R}^d} \; g^\top v \quad \text{s.t.} \quad \frac{1}{2} v^\top F v \le \delta, \qquad Pv = v.$$

Step 2: Form the Lagrangian. Introduce a scalar multiplier $\lambda \ge 0$ for the quadratic constraint and a vector multiplier $\mu \in \mathbb{R}^d$ for the linear constraint $Pv = v$. The Lagrangian is

$$\mathcal{L}(v, \lambda, \mu) = g^\top v - \lambda\Big(\frac{1}{2} v^\top F v - \delta\Big) + \mu^\top (Pv - v). \tag{41}$$

Step 3: Stationarity yields a linear system on $\text{range}(P)$. Taking the gradient of (41) with respect to v gives the stationarity condition

$$\nabla_v \mathcal{L}(v, \lambda, \mu) = g - \lambda F v + P^\top \mu - \mu = g - \lambda F v + (P - I)\mu = 0, \tag{42}$$

where we used $P^\top = P$ since P is a Euclidean projector. Left-multiplying (42) by P and using $P(P - I) = 0$ gives

$$Pg = \lambda P F v.$$

Since $Pv = v$, we have $PFv = PFPv = (PFP)v$, hence

$$Pg = \lambda (PFP)\, v. \tag{43}$$

Step 4: Solve using the Moore–Penrose pseudoinverse. Equation (43) admits the solution

$$v^\star = \frac{1}{\lambda} (PFP)^\dagger P g, \tag{44}$$

which lies in $\text{range}(P)$ because $(PFP)^\dagger = P(PFP)^\dagger P$.

Step 5: The quadratic constraint is active and determines λ. At the optimum, the quadratic constraint is active; otherwise one could scale v up to increase $g^\top v$ while remaining feasible. Therefore, $\frac{1}{2}(v^\star)^\top F v^\star = \delta$. Using $Pv^\star = v^\star$, we have $(v^\star)^\top F v^\star = (v^\star)^\top (PFP) v^\star$. Substituting (44) yields

$$2\delta = (v^\star)^\top (PFP) v^\star = \frac{1}{\lambda^2} (Pg)^\top (PFP)^\dagger (PFP)(PFP)^\dagger (Pg) = \frac{1}{\lambda^2} (Pg)^\top (PFP)^\dagger (Pg),$$

where we used the pseudoinverse identity $A^\dagger A A^\dagger = A^\dagger$ with $A = PFP$. Thus

$$\lambda = \sqrt{\frac{(Pg)^\top (PFP)^\dagger (Pg)}{2\delta}}. \tag{45}$$

Step 6: Compute the optimal value. Since $v^\star \in \mathrm{range}(P)$, we have $g^\top v^\star = (Pg)^\top v^\star = g^\top P v^\star$. Combining (44) and (45),

$$\begin{aligned} g^\top v^\star &= \frac{1}{\lambda}\, g^\top (PFP)^\dagger (Pg) \\ &= \sqrt{2\delta}\, \sqrt{\frac{(Pg)^\top (PFP)^\dagger (Pg)}{1}} \\ &= \sqrt{2\delta}\, \sqrt{g^\top (PFP)^\dagger g} = \sqrt{2\delta}\, \sqrt{\Gamma_F(P)}. \end{aligned}$$

□

Proof of Lemma 3.6. Fix any residual mode $(i,j) \in \mathcal{I}_{\mathrm{res}}$ and any blocked mode $(i',j') \in \mathcal{I}_{\mathrm{blk}}$. By (19),

$$g_{i'j'}^2 \le C\, g_{ij}^2. \tag{46}$$

By (18), $\lambda_{i'j'} \ge \kappa \lambda_{ij}$, hence

$$\lambda_{i'j'}^{-1} \le (\kappa \lambda_{ij})^{-1}. \tag{47}$$

Multiply (46) and (47):

$$\lambda_{i'j'}^{-1} g_{i'j'}^2 \le (\kappa\lambda_{ij})^{-1} \cdot C g_{ij}^2 = \frac{C}{\kappa}\, \lambda_{ij}^{-1} g_{ij}^2. \tag{48}$$

Therefore, the KL-efficiency margin between the residual and blocked mode is lower-bounded by:

$$\lambda_{ij}^{-1} g_{ij}^2 - \lambda_{i'j'}^{-1} g_{i'j'}^2 \ge \Big(1 - \frac{C}{\kappa}\Big) \lambda_{ij}^{-1} g_{ij}^2. \tag{49}$$

If $C < \kappa$, then $1 - \frac{C}{\kappa} > 0$. If additionally the minimum residual-mode efficiency is strictly positive, i.e., $\inf_{(i,j)\in\mathcal{I}_{\mathrm{res}}} \lambda_{ij}^{-1} g_{ij}^2 > 0$, then the right-hand side of (49) is uniformly bounded away from zero over all admissible pairs. This implies that the infimum over all such pairs, defined as α in (20), satisfies $\alpha > 0$. □

We also provide auxiliary perturbation lemmas for Theorem 3.7.

Lemma D.1 (Loewner sandwich under relative perturbation)**.** *Under* (21) *with* $\varepsilon < 1$*,*

$$(1-\varepsilon) F_0 \preceq F \preceq (1+\varepsilon) F_0, \tag{50}$$

and for any spectral-selector projector P in the ideal factorized Fisher basis $\{f_{ij}\}$,

$$\frac{1}{1+\varepsilon}\, \Gamma_0(P) \;\le\; \Gamma_F(P) \;\le\; \frac{1}{1-\varepsilon}\, \Gamma_0(P), \qquad \textit{where} \quad \Gamma_0(P) := g^\top (P F_0 P)^\dagger g. \tag{51}$$

Proof. Let $A := F_0^{-1/2} E F_0^{-1/2}$. By the definition of ε in (21), we have $\|A\|_{\mathrm{op}} = \varepsilon$. Since A is symmetric, all eigenvalues of A lie in $[-\varepsilon, \varepsilon]$, which implies $-\varepsilon I \preceq A \preceq \varepsilon I$. Adding I yields $(1-\varepsilon) I \preceq I + A \preceq (1+\varepsilon) I$. Conjugating by $F_0^{1/2}$ and substituting $F = F_0 + E = F_0^{1/2}(I+A)F_0^{1/2}$ gives (50).

For a spectral-selector P in the factorized Fisher basis, $PF_0 = F_0 P$. Left- and right-multiplying (50) by P yields $(1-\varepsilon) P F_0 P \preceq PFP \preceq (1+\varepsilon) P F_0 P$. Taking the Moore–Penrose pseudoinverses on $\mathrm{range}(P)$ reverses the Loewner order:

$$\frac{1}{1+\varepsilon} (P F_0 P)^\dagger \preceq (PFP)^\dagger \preceq \frac{1}{1-\varepsilon} (P F_0 P)^\dagger. \tag{52}$$

Applying the quadratic form with g to this inequality immediately yields (51). □

Proof of Theorem 3.7. By a spectral-selector projector in the factorized Fisher basis $\{f_{ij}\}$ we mean $P = \sum_{(i,j)\in\mathcal{I}} f_{ij} f_{ij}^\top$ for some index set $\mathcal{I} \subseteq [d_{\mathrm{out}}] \times [d_{\mathrm{in}}]$.

Step 1: An ideal exchange bound for Γ_0. Because P is a spectral selector, $\Gamma_0(P)$ is diagonal. Using $PF_0 = F_0 P$, we have $(PF_0P)^\dagger = PF_0^{-1}P$. Thus:

$$\Gamma_0(P) = g^\top P F_0^{-1} P g = \sum_{(i,j)\in\mathcal{I}(P)} \lambda_{ij}^{-1} g_{ij}^2. \tag{53}$$

Similarly, since the RCA projector P_{O} exactly selects $\mathcal{I}_{\mathrm{res}}$ under the ideal model:

$$\Gamma_0(P_{\mathrm{O}}) = \sum_{(i,j)\in\mathcal{I}_{\mathrm{res}}} \lambda_{ij}^{-1} g_{ij}^2. \tag{54}$$

Assume P contains t blocked modes ($t = |\mathcal{I}(P) \cap \mathcal{I}_{\mathrm{blk}}| \geq 1$). Since P and P_{O} have the same rank, P must omit exactly t residual modes relative to P_{O}. We can construct a sequence of t exchanges, where each step swaps one blocked mode in $\mathcal{I}(P)$ for one omitted residual mode from $\mathcal{I}_{\mathrm{res}}$. By the definition of α in (20), each exchange strictly increases Γ_0 by at least α. After t exchanges, the resulting set $\mathcal{I}^{(t)}$ contains no blocked modes, meaning $\mathcal{I}^{(t)} \subseteq \mathcal{I}_{\mathrm{res}}$. Consequently:

$$\Gamma_0(P_{\mathrm{O}}) \geq \Gamma_0(P_{\mathcal{I}^{(t)}}) \geq \Gamma_0(P) + t\alpha. \tag{55}$$

Step 2: Subtract and insert the ideal margin. From Lemma D.1, we have $\Gamma_F(P_{\mathrm{O}}) \geq \frac{1}{1+\varepsilon}\Gamma_0(P_{\mathrm{O}})$ and $\Gamma_F(P) \leq \frac{1}{1-\varepsilon}\Gamma_0(P)$. Subtracting the second from the first gives:

$$\Gamma_F(P_{\mathrm{O}}) - \Gamma_F(P) \geq \frac{1}{1+\varepsilon}\Gamma_0(P_{\mathrm{O}}) - \frac{1}{1-\varepsilon}\Gamma_0(P). \tag{56}$$

Inserting (55) into (56):

$$\begin{aligned}
\Gamma_F(P_{\mathrm{O}}) - \Gamma_F(P) &\geq \frac{1}{1+\varepsilon}\big(\Gamma_0(P) + t\alpha\big) - \frac{1}{1-\varepsilon}\Gamma_0(P) \\
&= \frac{t\alpha}{1+\varepsilon} + \Gamma_0(P)\Big(\frac{1}{1+\varepsilon} - \frac{1}{1-\varepsilon}\Big) \\
&= \frac{t\alpha}{1+\varepsilon} - \Gamma_0(P)\Big(\frac{2\varepsilon}{(1-\varepsilon)(1+\varepsilon)}\Big),
\end{aligned} \tag{57}$$

which is exactly (22).

Step 3: Sufficient condition for strict positivity. If the right-hand side of (22) is strictly positive, then $\Gamma_F(P_{\mathrm{O}}) > \Gamma_F(P)$. The condition

$$\frac{t\alpha}{1+\varepsilon} > \frac{2\varepsilon}{(1-\varepsilon)(1+\varepsilon)}\Gamma_0(P) \tag{58}$$

is equivalent (upon multiplying both sides by $1+\varepsilon$) to (23). □

D.2 Fisher-Channel Alignment Metric

To audit whether updates leak into the protected high-curvature channels, we define a Fisher-channel alignment metric. For a gradient or update matrix G, define the blocked-union projection

$$\mathcal{P}_{\mathrm{F}}(G) = P_\delta G + G P_x - P_\delta G P_x. \tag{59}$$

The Fisher-channel alignment metric is

$$\eta_{\mathrm{F}}(G) = \frac{\|\mathcal{P}_{\mathrm{F}}(G)\|_F^2}{\|G\|_F^2}. \tag{60}$$

Corollary D.2 (Zero protected-channel alignment)**.** *For any effective update D generated by RCA, $\eta_{\mathrm{F}}(D) = 0$.*

In experiments, we estimate η_{F} using the same KFAC factors that define the protected channels, together with exact-factor audits on tractable sentinel layers.

D.3 Proof of Corollary D.2 (Zero Alignment Property)

Proof. By Lemma 3.3, any RCA effective update $D = \Delta W_{\mathrm{eff}} = Q_\delta B A Q_x$ lies entirely in the residual subspace. This implies that $P_\delta D = 0$ and $D P_x = 0$. The alignment metric $\eta_{\mathrm{F}}(D)$ is defined via the Frobenius norm of the blocked-union projection:

$$\mathcal{P}_{\mathrm{F}}(D) = P_\delta D + D P_x - P_\delta D P_x. \tag{61}$$

Substituting $P_\delta D = 0$ and $D P_x = 0$ into this expression yields:

$$\mathcal{P}_{\mathrm{F}}(D) = 0 + 0 - 0 = 0. \tag{62}$$

Therefore, $\|\mathcal{P}_{\mathrm{F}}(D)\|_F^2 = 0$, giving $\eta_{\mathrm{F}}(D) = 0$. This confirms that the effective update exhibits zero leakage into the high-curvature Fisher channels. □

E More Experimental Setup

E.1 System Efficiency Comparison

To evaluate whether the theoretical and performance gains of RCA incur a practical systems penalty, we summarize the optimization cost and storage footprint on Qwen2.5-Math-1.5B in Table 4.

Table 4: **Efficiency comparison on Qwen2.5-Math-1.5B.** This table summarizes optimization cost and storage footprint to show whether the gains come with a practical systems penalty.

Method	Trainable Params	Checkpoint Size	Peak Train GPU Mem	Train Step Time
FFT	1.54B	17.3 GiB	39.0 GB	182.4 s/step
LoRA	36.9M	140.9 MiB	34.5 GB	111.8 s/step
RCA	36.9M	140.9 MiB	34.5 GB	104.6 s/step

As shown in Table 4, compared to Full Finetuning (FFT), all LoRA-based methods including RCA drastically reduce the trainable parameter count from 1.54B to 36.9M, leading to a massive decrease in checkpoint size (from 17.3 GiB to 140.9 MiB). Furthermore, RCA maintains a peak training GPU memory of 34.5 GB, which is identical to standard LoRA and noticeably lower than FFT's 39.0 GB. In terms of training step time, RCA takes 104.6 s/step, which is similar to standard LoRA (111.8 s/step), while being significantly more efficient than FFT (182.4 s/step). This confirms that RCA introduces negligible computational overhead over standard LoRA, preserving the core efficiency benefits of parameter-efficient fine-tuning while achieving substantially better performance.

E.2 Prompt Templates

We employ a standard chain-of-thought prompt structure. The user message concatenates the problem statement with specific formatting instructions to elicit reasoning within `<think>` tags and the final answer within `\boxed{}`.

```
<user>
{QUESTION}

Let's think step by step and enclose the reasoning process within <think>
    and </think> tags.
The final result in the answer MUST BE within \boxed{}.
</user>
```

E.3 Hyperparameters

We utilize the **VERL** library for all experiments with the GRPO algorithm. All models are trained for 1 epoch.

Table 5: Training Hyperparameters.

Configuration	Value
Training Batch Size	128
Rollouts per Prompt (G)	8
Max Prompt Length	512 tokens
Max Response Length	3072 tokens
Optimizer	AdamW
Model-Specific Settings	
Learning Rate	1e-5
Rollout Temperature	1.0 (Qwen2.5-Math / DeepSeek) 0.6 (Llama / Qwen3)
Entropy Coefficient	0.0 (Default) / -1e-3 (Llama)
KL Coefficient (β)	0.001

Table 6: LoRA and RCA hyperparameters.

Configuration	Value
Common LoRA Settings	
LoRA Rank (r)	32
LoRA Alpha (α)	32
Target Modules	All Linear Layers
RCA	
Blocked Rank (r_k)	16
Block Frequency	Every Step
RCA Basis	Fisher/KFAC calibration sketch
Reference Policy	Frozen Base Model

E.4 Fisher/KFAC Calibration Details

We collect layer inputs and output gradients through hooks, build randomized sketches of C_x and C_δ, extract $r_k = 16$ orthonormal basis vectors per side, and store them as frozen buffers. The calibration does not change the base model or LoRA weights.

E.5 Datasets

Training.

- **DeepScaleR** (Luo et al., 2025): the math-reasoning corpus used for RLVR rollouts and policy updates.

Evaluation. We evaluate with the standard official/problem releases for each benchmark and apply the same rule-based programmatic checking protocol throughout.

- **MATH500** (Lightman et al., 2024): a verifiable subset of math reasoning problems for reliable evaluation.
- **AIME 2024** (HuggingFaceH4, 2024): the 2024 American Invitational Mathematics Examination set.
- **AIME 2025** (OpenCompass, 2025): the 2025 AIME set.
- **AMC 2023** (math-ai, 2025): problems from the 2023 American Mathematics Competitions.
- **Minerva Math** (Lewkowycz et al., 2022): the Minerva math evaluation suite.
- **OlympiadBench** (He et al., 2024): olympiad-level mathematics benchmark.
- **MMLU** (Hendrycks et al., 2021): a broad multi-task benchmark covering 57 academic and professional subjects for evaluating world knowledge and problem solving.
- **ARC-C** (Clark et al., 2018): the Challenge split of the AI2 Reasoning Challenge, consisting of grade-school science questions that are difficult for retrieval and word co-occurrence baselines.
- **HellaSwag** (Zellers et al., 2019): a commonsense natural language inference benchmark for physically grounded sentence-completion reasoning.
- **GPQA** (Rein et al., 2024): a graduate-level, Google-proof multiple-choice QA benchmark written by domain experts in biology, physics, and chemistry.

F Additional Experimental Results

The following studies examine reward–curvature efficiency, the persistence of the calibrated basis, curvature-aware alternatives, and retention. Unless specified otherwise, we use Qwen2.5-Math-1.5B and the main experimental protocol. Retention results use the Llama-3.2-3B-Instruct setting of Table 1. The shared-learning-rate experiment is not repeated here.

F.1 Reward–Curvature Efficiency

We compare equal-cardinality residual and blocked selectors in the calibrated KFAC geometry. The per-coordinate efficiency is $\rho_{ij} = g_{ij}^2/\lambda_{ij}$, and the aggregate comparison concerns Γ_{F_0}. Tables 7 and 8 report the aggregate efficiency statistics and pairwise win rates. The coordinate audit uses 392 equal-cardinality projected coordinates and 2,000 prompt-cluster bootstrap replicates.

Table 7: Aggregate reward–curvature statistics in the calibrated proxy geometry. Brackets denote reported 95% confidence intervals.

Step	$\Gamma_{F_0}^{\mathrm{res}}$ [95% CI]	$\Gamma_{F_0}^{\mathrm{blk}}$ [95% CI]	R_{local,F_0} [95% CI]
0	17.984 [11.674, 68.857]	2.032 [1.190, 7.475]	2.975 [1.883, 4.006]
100	1.985 [0.806, 10.093]	0.0423 [0.0209, 0.270]	6.852 [5.057, 8.146]
200	8.107 [3.477, 31.626]	0.188 [0.0753, 0.814]	6.571 [4.597, 8.454]
300	5.881 [1.347, 42.114]	0.193 [0.0306, 1.083]	5.518 [5.076, 6.002]

Table 8: Residual-versus-blocked pairwise efficiency win rates.

Step	Residual pairwise win rate [95% CI]
0	0.811 [0.760, 0.878]
100	0.898 [0.862, 0.944]
200	0.918 [0.862, 0.949]
300	0.913 [0.872, 0.949]

The reported R_{local,F_0} values range from 2.975 to 6.852, and their reported confidence intervals lie above one at all four checkpoints. Residual coordinates win 81.1–91.8% of pairwise comparisons, with all reported intervals above 0.5. These results support aggregate and typical-case separation in the measured frozen KFAC grid. They do not establish the stronger uniform all-pairs minimum margin or a full-space true-Fisher perturbation bound. The general comparison depends on a positive comparator-specific ideal gap $\Delta_0(P) = \Gamma_{F_0}(P_{\mathrm{O}}) - \Gamma_{F_0}(P) > 0$; the uniform pairwise condition remains a sufficient condition.

F.2 Fixed versus Refreshed Calibration Bases

We compare the initial basis with checkpoint recalibration using rotation-invariant input/output overlaps and current-Fisher capture,

$$\mathrm{Capture}_{0\to t} = \frac{\mathrm{tr}(P_{\mathrm{blk},0} F_t^{\mathrm{KFAC}})}{\mathrm{tr}(P_{\mathrm{blk},t} F_t^{\mathrm{KFAC}})}.$$

Fixed and refreshed bases are compared through matched 10% continuation runs from identical weights. This evaluates subsequent learning rather than an immediate change of projector alone.

Table 9: Initial-basis overlap, current-Fisher capture, and fixed/refreshed continuation pass@1 (%).

Checkpoint	Input overlap	Output overlap	Initial capture	Random capture	Fixed p@1	Refresh p@1
25%	0.94	0.91	0.96	0.018	34.5	34.6
50%	0.90	0.85	0.92	0.021	36.7	36.3
75%	0.86	0.79	0.87	0.019	37.8	38.7
100%	0.82	0.74	0.89	0.020	38.4	39.8

At the final checkpoint, the initial basis retains 0.89 of checkpoint-optimal blocked Fisher mass, compared with 0.020 for a random basis. The final fixed and refreshed scores are 38.4 and 39.8, respectively. Thus the initial calibration remains informative, while refresh can improve late-stage

performance. Refresh requires another Fisher computation; the table does not quantify its computational cost. These observations do not imply exact eigenspace stationarity or that refresh is never useful.

F.3 Curvature-Aware Baselines

EWC-LoRA (Zheng et al., 2026) uses a soft Fisher-weighted penalty; KFAC-preconditioned LoRA modifies the gradient; GeoRA uses a geometry-aware initialization and adapter structure; RCA enforces hard bilateral support in calibrated policy-Fisher channels. The comparisons use the same trainer. The EWC comparison uses $\lambda_{\mathrm{EWC}} = 10^{-3}$, while RCA uses $r_k = 16$ and projection after every optimizer step.

Table 10: Curvature-aware baseline comparison. The $\pm$ summaries are retained as reported; no additional seed count or uncertainty definition is inferred.

Method	Mechanism	Pass@1
EWC-LoRA	Soft Fisher penalty	36.2 ± 0.7
KFAC-LoRA	Gradient preconditioning	36.8 ± 0.7
GeoRA	Geometric initialization/anchor	37.6 ± 0.6
RCA	Hard policy-Fisher support	38.4 ± 0.5

Table 11: Performance and reward/KL summaries for soft regularization and geometry-aware adaptation. Reward/KL is normalized to LoRA.

Method	Pass@1	Log-k AUC	Reward / KL
Shared-config LoRA	24.2 ± 0.8	44.0 ± 0.9	$1.00\times$ normalized
EWC-LoRA	36.2 ± 0.7	65.8 ± 0.8	$1.18\times$ LoRA
GeoRA	37.6 ± 0.6	68.2 ± 0.6	$1.38\times$ LoRA
RCA	38.4 ± 0.5	69.3 ± 0.5	$1.58\times$ LoRA

RCA exceeds EWC-LoRA by 2.2 pass@1 points and 3.5 log-k AUC points, and exceeds GeoRA by 0.8 and 1.1 points, respectively. Its reported reward/KL is $1.58\times$ LoRA, versus $1.18\times$ for EWC-LoRA and $1.38\times$ for GeoRA. EWC-LoRA leakage is 0.14, while RCA leakage is 1.06×10^{-15} into the calibrated blocked union. These comparisons distinguish the tested update rules within this setting.

F.4 Expanded Knowledge-Retention Evaluation

We extend the Llama-3.2-3B-Instruct retention comparison with PiSSA, OPLoRA, EWC-LoRA, and GeoRA, using mathematical-RLVR checkpoints and the same MMLU, ARC-Challenge, HellaSwag, and GPQA evaluation. Pass@1 is the primary statistic; high-k results in the main table remain secondary.

Table 12: Expanded Llama-3.2-3B-Instruct retention comparison: pass@1 (%).

Method	MMLU	ARC-C	HellaSwag	GPQA
Base	60.2	73.3	54.5	23.5
FFT	62.1	74.6	56.1	28.3
LoRA	49.5	44.0	32.9	19.3
PiSSA	38.0	32.0	25.0	16.0
OPLoRA	60.5	73.5	54.6	28.0
EWC LoRA	61.0	74.5	55.5	29.0
GeoRA	61.4	74.8	56.2	29.1
RCA	62.1	75.3	59.1	29.7

RCA exceeds OPLoRA on all four datasets, ties FFT for the highest MMLU pass@1 at 62.1, and has the highest pass@1 among the listed methods on ARC-Challenge, HellaSwag, and GPQA. Its GPQA score is 29.7, compared with 29.1 for GeoRA. These results support comparison against the listed adapters on this backbone rather than a universal retention advantage. No numerical second-backbone retention result is provided here.

F.5 Aggregate Performance Comparisons

Table 13 reports macro pass@1 differences, log-k AUC differences, relative failure-rate reductions, and paired-bootstrap intervals. The reported intervals for pass@1 differences are all positive. Counts from the original four-backbone, three-dataset, six-k comparison are 53/72 wins against FFT, 61/72 against AdaLoRA, 67/72 against OLoRA, 68/72 against OPLoRA, and 72/72 against RSLoRA, LoRA, and PiSSA. The original retention comparison has 22 wins, one tie, and one loss across 24 entries against FFT, and 24/24 wins against LoRA. These counts refer to their original benchmark–k grids.

Table 13: Reported aggregate differences and paired-bootstrap confidence intervals. "pp denotes percentage points.

Comparison	Macro p@1 diff.	Log-k AUC diff.	Failure reduction	Paired-bootstrap CI
RCA vs Base	$+20.8$ pp	$+20.3$	25.2%	$[+18.5, +23.1]$ pp
RCA vs FFT	$+2.5$ pp	$+3.3$	3.9%	$[+1.2, +3.8]$ pp
RCA vs shared-config LoRA	$+14.2$ pp	$+25.3$	18.7%	$[+12.8, +15.6]$ pp
RCA vs best non-Fisher PEFT	$+2.8$ pp	$+2.8$	4.3%	$[+1.6, +4.0]$ pp
RCA vs best curvature-aware PEFT	$+0.8$ pp	$+1.1$	1.3%	$[+0.1, +1.5]$ pp

The reported macro pass@1 gain over trained alternatives ranges from 0.8 points against the best curvature-aware PEFT baseline to 14.2 against shared-configuration LoRA. The AUC differences in Table 13 and the absolute AUC summaries in Table 11 are retained as separately reported summaries.

F.6 Implicit Geometry versus Explicit Bilateral Support

We compare full finetuning, shared-configuration LoRA, GeoRA, RCA, and a random bilateral projector using weight off-principal alignment, policy-Fisher blocked leakage, KL per update norm, reward/KL, and pass@1. Weight-space off-principal alignment and leakage into calibrated policy-Fisher channels are distinct diagnostics.

Table 14: Implicit off-principal geometry and explicit policy-Fisher support. A dash indicates that leakage is not supplied for the random control.

Method	Off-princ. align.	Blocked leakage	KL / update norm	Reward / KL	Pass@1
Full finetuning	0.84	0.11	0.38	$1.28\times$ LoRA	35.9 ± 0.8
Shared-config LoRA	0.73	0.27	0.78	$1.00\times$ normalized	24.2 ± 0.8
GeoRA	0.91	0.08	0.30	$1.38\times$ LoRA	37.6 ± 0.6
RCA	0.93	1.06×10^{-15}	0.22	$1.58\times$ LoRA	38.4 ± 0.5
Random bilateral projection	0.50	-	0.95	$0.08\times$ LoRA	2.0 ± 1.5

Full finetuning has off-principal alignment 0.84 but blocked policy-Fisher leakage 0.11. RCA has alignment 0.93 and leakage 1.06×10^{-15}; GeoRA has alignment 0.91 and leakage 0.08. The random bilateral control has pass@1 2.0 ± 1.5. The comparison illustrates that a strong weight-space off-principal tendency does not itself enforce RCA's calibrated support invariant.