# RANK IS NOT A TRUST REGION: OPTIMIZER-INDUCED GEOMETRY IN LORA REINFORCEMENT LEARNING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Low-rank adaptation (LoRA) is widely used for reinforcement learning with verifiable rewards (RLVR), and a low rank is commonly expected to keep the policy close to where it started. We show that rank is not a trust region: in GRPO training of Qwen3.5-4B, rank-4 and rank-64 adapters that update the weights equally far differ by $11\times$ in on-policy KL, and by $9\times$ under an explicit KL penalty. Two updates matched in rank, norm, and row and column subspaces differ by $240\times$ in held-out KL. The reason is that policy KL depends on the direction of an update and not only on its size. Under standard initialization, the down-projection of LoRA barely moves, so its rows act as nearly fixed random probes of the gradient. As the gradient concentrates in directions to which the policy is most sensitive, a higher rank supplies more probes that capture this sensitivity, so the KL cost of a matched displacement grows with rank. Changing the probes tests this mechanism: probes aligned with the gradient reverse the rank ordering of curvature and, when KL stays local, reverse the rank ordering of KL as well. Directional Fisher sensitivity, which measures curvature along the realized update per unit norm, predicts KL on held-out sequences in two model families. Policy movement also does not determine whether an update helps, because KL is locally even in the update direction while utility change is odd. Stability in LoRA-based RLVR should therefore be monitored and constrained in distribution space, with rank chosen for capacity rather than as a trust-region parameter.

## 1 INTRODUCTION

Low-rank adaptation (LoRA) (Hu et al., 2022) is the standard way to fine-tune large policies with reinforcement learning from verifiable rewards (RLVR). For a linear module, LoRA represents the added weight as a scaled product of a down-projection $A$ and an up-projection $B$ with their shared dimension $r$ as the adapter rank. We ask whether $r$ actually controls how far the policy moves in distribution space. Rank is commonly treated as a stability knob, where low-rank adapters learn less and forget less than full fine-tuning (Biderman et al., 2024), so lower rank and smaller effective-weight displacement are expected to keep the policy near its initialization. In practice, when RLVR runs without an explicit KL penalty, rank is the only structural constraint on how far the policy moves. Beyond this, drift is tracked through weight norms; adaptation methods for RLVR are designed and analyzed through the subspaces their updates occupy (Zhu et al., 2025; Zhang et al., 2026a;b); and initializations from weight or gradient singular vectors are adopted for faster convergence (Meng et al., 2024; Wang et al., 2024). Each of these practices treats rank or displacement as a proxy for how far the policy moves. In effect, rank is used as an implicit trust region, which has two roles: it bounds how far each update moves the policy, and by doing so it makes the local objective a reliable guide to improvement (Schulman et al., 2015).

We observe the proxy failing in the GRPO training of Qwen3.5-4B on mathematical reasoning, where rank-stabilized adapters of rank 4 and 64 move the weights equally far (Figure 1a), yet the rank-64 policy moves about $11\times$ more in on-policy KL. The gap persists on held-out prompts and under an explicit KL penalty, recurs on a second model family, and is not explained by larger or more coherent steps. At a single rank the failure is sharper: two updates with identical rank, norm, row space, and column space differ by more than two orders of magnitude in held-out KL (Figure 1c; Section 3).



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.

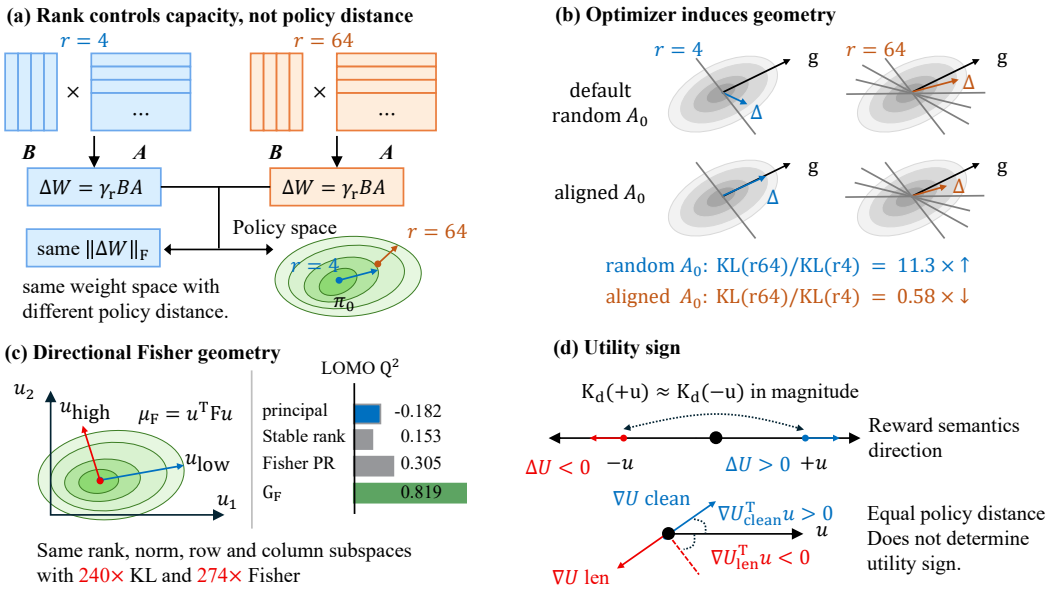



Figure 1: **Overview: LoRA rank changes where the optimizer moves.** (a) Rank sets the capacity of the update manifold, but rank-stabilized adapters with equal weight displacement $\|\Delta W\|_{\mathrm{F}}$ move the policy by different amounts. (b) Under the standard initialization, the rows of $A_0$ act as random probes and the update $\Delta$ is the gradient $g$ projected onto them; more probes capture more high-curvature gradient (random $A_0$: rank-64/rank-4 on-policy KL ratio $11.3\times$ at matched displacement and local KL), while probes aligned with the leading gradient directions reverse the ordering ($0.58\times$). (c) Directional Fisher sensitivity separates updates that share rank, norm, and row and column subspaces and predicts held-out KL ratios where spectral summaries do not. (d) KL is locally even in the update direction while utility change is odd, so equal policy movement can raise or lower utility depending on how the reward gradient aligns with the update.

The issue is not unique to LoRA. PPO ratio clipping does not keep the probability ratio within its bounds (Wang et al., 2020; Engstrom et al., 2020), and the KL trust region PPO maintains in practice comes from how its objective is optimized rather than from the clipping (Engstrom et al., 2020). Parameter-space sharpness likewise changes under reparameterizations that leave the function intact (Dinh et al., 2017). For LoRA, equal task accuracy can hide a spectral structure different from full fine-tuning (Shuttleworth et al., 2025). We therefore ask what determines policy movement under LoRA if rank and displacement do not, how rank enters it, and whether bounding the movement would decide if an update helps.

We explain the failure with a mechanistic theory in Section 4. For an effective update $\Delta$, finite KL is half the squared effective-weight displacement multiplied by the path-averaged curvature along that update, so the KL budget is an ellipsoid, and rank, norm, or subspace alone cannot determine it. Adapters of different rank therefore realize different directions, and LoRA's factorization selects among them. Under the standard initialization, $B$ starts at zero while $A$ starts random and barely moves. The rows of this nearly fixed $A$ act as random directions, which we call *probes*, and $B$ accumulates the gradient after projection through those probes (Figure 1b). Because the gradient concentrates in high-curvature directions of the policy Fisher, more probes capture more of this curvature. Since the number of probes is the rank, the KL cost of a matched displacement grows with rank. We call this growth the *rank effect*. A bounded KL would still leave the second role of a trust region unfilled, because KL is locally even in the update direction while the first-order change of a utility is odd (Figure 1d).

The account makes predictions that can be tested by changing the probes in Section 5. Freezing $A$ strengthens the rank effect, while aligning its rows with the leading gradient directions reverses it. In online training, the aligned design reverses the KL gap itself while KL stays local. Allowing $A$ to move faster weakens the effect on a fixed batch, where its drift is directed, but not online, where it

is a random walk. Directional Fisher sensitivity predicts KL on held-out sequences in two model families (Section 5.3). For utility, a length term in the reward lowers general-text likelihood, while a high-rank run that moves the policy much farther loses no conditional accuracy.

**Related work.** Three lines of work are closest. First, with $A$ near initialization LoRA acts as a random projection of the gradient, a view used to explain memory savings and factor asymmetry (Hao et al., 2024; Zhang et al., 2023; Zhu et al., 2024), while equivalent-gradient analyses characterize the update the factors induce (Wang et al., 2025a; Zhang & Pilanci, 2024). These works study optimization and memory, whereas we use the projection to explain policy KL. Second, RLVR updates avoid the principal directions of the weights, and a KL budget implies a local Fisher-norm constraint (Zhu et al., 2025). This work locates where updates go, whereas we ask how rank sets the curvature along them. Third, forgetting tracks KL on the new task (Shenfeld et al., 2026). We instead ask whether the magnitude of KL fixes the sign of utility. Appendix A covers adapter designs and trust-region methods.

**Contributions.** We (i) show that matched weight displacement and matched subspaces leave order-of-magnitude gaps in policy KL under LoRA; (ii) explain the rank effect by the random probes that $A_0$ defines, with predictions for frozen, aligned, and drifting probes; (iii) reverse the rank ordering of curvature in online RLVR, and of KL while it stays local, by aligning the probes with the gradient; (iv) show that directional Fisher sensitivity predicts held-out KL beyond what rank explains and where spectral summaries fail; and (v) show that policy movement does not fix the sign of utility change, which follows the reward rather than the size of the movement.

## 2 PRELIMINARIES

**Policy movement.** For a fixed prefix occupancy measure $d$ and reference parameters $\theta_0$, the conditional policy KL is

$$K_d(\theta, \theta_0) = \int D_{\mathrm{KL}}(\pi_\theta(\cdot \mid s) \,\|\, \pi_{\theta_0}(\cdot \mid s))\; d(s). \tag{1}$$

Policy movement is $K_d$ with $d = N^{-1} \sum_i \sum_{t=1}^{L_i} \delta_{s_{it}}$ over $N$ completions that are sampled once and then held fixed for evaluation, i.e., sequence-summed token KL. On-policy KL uses each policy's generated prefixes, $d = d_\theta$; fixed-prefix KL uses frozen reference-policy prefixes, $d = d_{\mathrm{ref}}$, which removes state-visitation shift. Held-out denotes prompts or sequences disjoint from those used for training or estimation.

**LoRA factors.** A LoRA module updates $X = \gamma_r BA$ with $B \in \mathbb{R}^{d_{\mathrm{out}} \times r}$, $A \in \mathbb{R}^{r \times d_{\mathrm{in}}}$, and multiplier $\gamma_r = \alpha/r$ (standard) or $\alpha/\sqrt{r}$ (rank-stabilized, rsLoRA (Kalajdzievski, 2023)). The standard initialization sets $B_0 = 0$ and draws $A_0$ at random. The effective update $\Delta$ stacks the vectorized module updates $X_m = \gamma_r B_m A_m$ of all trained modules, and the effective-weight displacement is $\|\Delta W\|_{\mathrm{F}} = \|\Delta\|_{\mathrm{F}} = (\sum_m \|X_m\|_{\mathrm{F}}^2)^{1/2}$. We write $K_d(\Delta) = K_d(\theta_0 + \Delta, \theta_0)$ for the policy movement of an update and $u$ for unit directions.

**Curvature.** The second-order term of $K_d$ along $u$ is $\frac{1}{2} u^\top H_d u$, where $H_d(\theta)$ is the full-vocabulary conditional KL Hessian. We measure it through the empirical sequence Fisher $F_d = N^{-1} \sum_i z_i z_i^\top$ of summed token scores $z_i$, a surrogate for $H_d$ (detailed in Appendix B), and use

$$q_F(\Delta; d) = \Delta^\top F_d \Delta, \qquad \mu_F(\Delta; d) = \frac{\Delta^\top F_d \Delta}{\|\Delta\|_{\mathrm{F}}^2}, \qquad \bar{\mu}_K(\Delta; d) = \frac{2K_d(\Delta)}{\|\Delta\|_{\mathrm{F}}^2}. \tag{2}$$

The *directional Fisher sensitivity* $\mu_F$ is the Fisher curvature along $\Delta$ per unit norm, and $\bar{\mu}_K$ is the path-averaged curvature of finite KL when $d$ is fixed.

**Setting.** Our main experiments train Qwen3.5-4B (Qwen Team, 2026) with GRPO (Shao et al., 2024) on GSM8K mathematical reasoning (Cobbe et al., 2021) without a KL penalty, at ranks $r \in \{4, 16, 64\}$ for 30 rollout rounds of two optimizer steps. Displacement-matched runs use rsLoRA, and a long-horizon pair extends ranks 4 and 256 to 200 rounds. We call these runs, in which

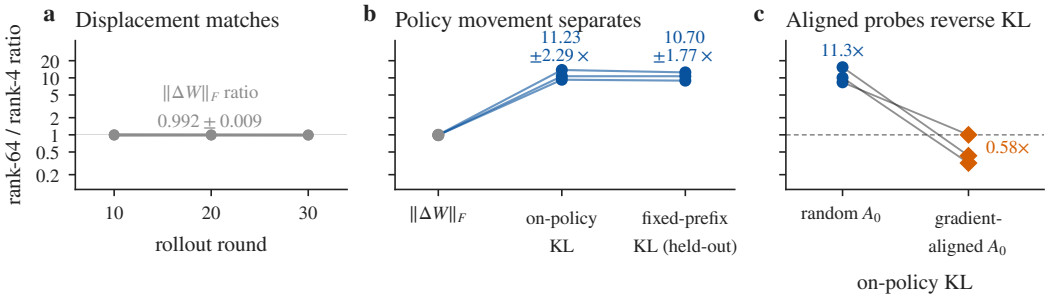



Figure 2: **Matched displacement, different policy movement, and its reversal by aligned probes.** All panels show rank-64/rank-4 ratios on a common log scale, one line per seed. (a) Effective-weight norm ratio over training for three rsLoRA seeds. (b) Endpoint norm ratio and KL ratios for the same seeds: on-policy under training evaluation and fixed-prefix on held-out prompts. (c) On-policy KL at matched displacement and a learning rate at which KL stays local, with random and with gradient-aligned $A_0$ (Section 5.4); aligned probes reverse the gap on average.

all adapted modules train on fresh rollouts, online training. Qwen2.5-1.5B-Instruct (Qwen Team, 2024) repeats the matched comparison in a second family. The mechanism is tested on single modules of Qwen3.5-4B and of Granite-4.2-3B (IBM Granite Team, 2026), trained on a fixed batch so that the two arms of each comparison see the same data and differ only in how the update is parameterized. Table 2 in Appendix C.1 lists what each protocol holds fixed and where it is used. Utility is evaluated on GSM8K (Cobbe et al., 2021), MATH-500 (Hendrycks et al., 2021b; Lightman et al., 2024), and IFEval (Zhou et al., 2023) with greedy decoding, and by FP32 negative log-likelihood (NLL) on 512 WikiText-103 passages (Merity et al., 2017) and MMLU choice margins (Hendrycks et al., 2021a). *Common-support conditional accuracy* compares two conditions on the prompts both complete without truncation. Values written as $x \pm y$ are the mean and sample standard deviation over three training seeds, and intervals are paired 95% bootstrap intervals over the evaluated texts, items, or prompts. Predictors are scored by out-of-fold $Q^2$ under leave-one-module-out (LOMO) cross-validation. Budgets, sample sizes, and statistical units are in Appendix C.1.

## 3 Policy Movement Varies at Matched Displacement and Subspace

If rank and displacement acted as a trust region, adapters that move the weights equally far would hold the policy to comparable KL, and updates of equal norm confined to the same subspace would move it by comparable amounts. Both expectations fail by an order of magnitude or more.

**Equal displacement, unequal policy movement.** Rank-stabilized adapters of rank 4 and 64 move the weights equally far, yet the rank-64 policy moves $11.23 \pm 2.29\times$ farther in on-policy KL over three seeds. Rank-stabilized scaling keeps the rank-64/rank-4 norm ratio at $0.992 \pm 0.009$, within $2\%$ throughout training, and the final rewards are comparable.

The gap persists across KL estimators and settings. It holds with frozen reference prefixes on training and held-out prompts (Figure 2b; Appendix C.2) and under an explicit KL penalty, which lowers the KL of both ranks but leaves a $9.25\times$ ratio (Appendix C.10). On Qwen2.5-1.5B-Instruct, held-out fixed-prefix KL differs by $12.8 \pm 4.5\times$ at norm ratios of 1.01–1.04. In a standard-LoRA pair trained on MATH problems instead of GSM8K, rank 64 moves the weights $3.98\times$ farther than rank 4, and its fixed-prefix KL per unit squared displacement, $\bar{\mu}_K$, is $11.2\times$ larger (Appendix C.2).

**Step sizes and path coherence match.** With per-round updates $\delta W_t$ and path coherence $\zeta_T = \|\sum_t \delta W_t\|_{\mathrm{F}}^2 \,/\, \sum_t \|\delta W_t\|_{\mathrm{F}}^2$, the rank-64/rank-4 coherence ratio is $0.970 \pm 0.032$ and the ratio of summed squared step norms $\sum_t \|\delta W_t\|_{\mathrm{F}}^2$ is $1.015 \pm 0.025$, so high rank takes neither larger nor more coherent steps.

**Equal subspace, unequal policy movement.** Mixing the factors of a trained update by an orthogonal matrix, $B \leftarrow BQ$, keeps its rank, row space, and column space but changes the update. The normalized updates $BQA/\|BQA\|_{\mathrm{F}}$, $Q \in O(r)$, form the orientation orbit $\mathcal{O}$. From a standard LoRA rank-64 checkpoint we compare the identity and 15 seeded orthogonal mixes, rescaled to a common norm, on 32 calibration prompts, where the original update has the largest $q_F$, and evaluate the frozen maximum–minimum pair on independent held-out prompts. Held-out fixed-prefix KL of this orientation pair differs by $240\times$ (Figure 3c; protocol in Appendix C.11), and endpoint sequence-Fisher sensitivity on a separate shared 64-sequence batch differs by $274\times$. The 15 rotations themselves span only $3.36\times$ in calibration $q_F$, so the gap comes from the direction the optimizer selected within the subspace, not from the spread of directions the subspace contains.

Rank, norm, and subspace thus leave policy movement undetermined, and the next section asks what determines it.

# 4 A Mechanistic Theory of Policy Movement Under LoRA

How far the policy moves depends on more than how far the weights move. We first make this precise, then derive how LoRA's factors select the direction of an update and why that selection depends on rank, and finally turn to the second role of a trust region, making improvement reliable. Section 5 tests the resulting predictions.

## 4.1 Policy movement is set by curvature along the update

For a fixed $d$, finite KL is $K_d(\Delta) = \frac{1}{2}\|\Delta\|_{\mathrm{F}}^2 \bar{\mu}_K(\Delta; d)$, squared displacement times the curvature averaged along the update path (Appendix C.3). Displacement fixes only the first factor. Locally the second is the curvature $u^\top H_d u$ of the direction, so the KL budget is an ellipsoid rather than a ball.

**Proposition 1** (Directional trust-region radius)**.** *For a unit direction $u$, let $h_d(u) = u^\top H_d(\theta) u > 0$. Since $K_d(\theta + Ru, \theta) = \frac{1}{2}R^2 h_d(u) + O(R^3)$, a local quadratic KL budget $\varepsilon$ has directional radius*

$$R(u) = \sqrt{\frac{2\varepsilon}{h_d(u)}}. \tag{3}$$

Norm bounds the step and rank the admissible directions. A low rank does not exclude high-curvature directions, so neither determines $h_d(u)$. The subspaces of an update do not determine it either: every statistic that depends only on $\mathrm{col}(u)$ and $\mathrm{row}(u)$ is constant on the orientation orbit $\mathcal{O}$, while $h_d$ need not be.

**Proposition 2** (Subspace-only prediction lower bound)**.** *Let $u_h, u_l \in \mathcal{O}$, let a predictor depend on $u$ only through $\mathrm{col}(u)$ and $\mathrm{row}(u)$, and suppose it predicts a positive target $Y_i$ within multiplicative factor $C$ for both $u_h$ and $u_l$. Then*

$$C \geq \sqrt{\frac{Y_h}{Y_l}}. \tag{4}$$

The predictor outputs one value $c$ for both members, and $Y_i/C \leq c \leq CY_i$ for $i \in \{h, l\}$ implies $Y_h/Y_l \leq C^2$ (Appendix B.6). For the orientation pair of Section 3, $C \geq \sqrt{240} \approx 15$. Curvature along the realized update, rather than rank, norm, or subspace, should therefore predict policy movement (prediction P1).

## 4.2 LoRA selects directions through random probes

Section 4.1 makes policy movement a property of the direction of an update, so the question becomes which directions LoRA realizes and how they depend on rank. Rank could matter through the feasible manifold it defines or through the directions the optimizer selects on it. Section 3 points to the second: within one subspace, rotations differ little while the optimizer's direction stands far apart. Equivalent-gradient and Riemannian analyses (Wang et al., 2025a; Zhang & Pilanci, 2024) make the second mechanism precise. With $X = \gamma_r BA$ and factor-to-weight Jacobian $J_\phi(\delta B, \delta A) = \gamma_r(\delta BA + B\delta A)$, factor-space SGD moves the effective weights to first order by

$$\delta X_{\mathrm{LoRA}} = -\eta \mathcal{M}_r[g], \qquad \mathcal{M}_r[g] = J_\phi J_\phi^\top[g] = \gamma_r^2(BB^\top g + gA^\top A), \tag{5}$$

which is steepest descent under a metric induced by the factors rather than the Euclidean one (Proposition 5 in Appendix B). The Fisher sensitivity of the realized direction is then the generalized Rayleigh quotient $g^\top \mathcal{M}^\top F_d \mathcal{M} g / g^\top \mathcal{M}^\top \mathcal{M} g$, which a scalar step size leaves unchanged (Proposition 6). The sensitivity is thus set by how the optimizer reorients the gradient relative to the Fisher.

The factors determine which directions $\mathcal{M}_r$ favors, because it has the Kronecker-sum spectrum $\gamma_r^2(b_i^2 + a_j^2)$ over the eigenvalues of $BB^\top$ and $A^\top A$. The experiments use AdamW (Loshchilov & Hutter, 2019), whose factor steps keep the update in the form $\delta B\, A_0$ when $A$ barely moves. Each factor entry typically moves by about $\eta_t$ per step, and at the learning rate $2 \times 10^{-5}$ used here $A$ drifts by 0.7–1.1% over 30 rounds and at most 5.6% over 200 (Appendix C.6). Because $A$ barely moves, the term $B\,\delta A$ stays small and the update is dominated by $\delta B\, A_0$, the gradient projected onto the rows of $A_0$. With $A_0^\top A_0 \approx \frac{1}{3} P_R$, where $P_R$ projects onto their span, $\mathcal{M}_r$ reduces to a random right projection. The rows of $A_0$ thus act as fixed random probes, and rank sets how many of them intercept the gradient.

How much curvature such an update carries depends on the Fisher spectrum, which the proposition below describes with a row-side model. Its statements about signs and orderings are what we test by intervention in Section 5. Its magnitudes leave out how the input and output sides of the scores couple within each sequence, so effect sizes are measured with the full sequence Fisher.

**Proposition 3** (Probe curvature)**.** *Let the Fisher act on the row side of a module as $I \otimes \Sigma$, where $\Sigma \in \mathbb{R}^{n \times n}$, $n = d_{\text{in}}$, has eigenvalues $\lambda_1 \geq \cdots \geq \lambda_n$, mean $\bar{\lambda} = \operatorname{tr}\Sigma / n$, and participation ratio $p = (\operatorname{tr}\Sigma)^2 / \operatorname{tr}\Sigma^2$. Let the update project a gradient whose row-side second moment is proportional to $\Sigma$ onto the row span of $r$ probes.*

1. *If the span is a uniformly random $r$-dimensional subspace, then under SGD, up to relative corrections of order $r/n$, the ratio of expected curvature to expected update energy is $\bar{\lambda}\big(1 + (r+1)/p\big)$.*

2. *If the probes are the top $r$ eigenvectors of $\Sigma$ and probe $i$ carries update energy $w_i$, then $\mu_F(r) = \sum_{i \leq r} w_i \lambda_i / \sum_{i \leq r} w_i$. For any fixed $w_i \geq 0$, including $w_i \propto \lambda_i$ as under SGD and $w_i = 1$ as under sign-normalized updates, $\mu_F(r)$ is nonincreasing in $r$ and at least $\lambda_r$.*

3. *If the probes drift from $A_0$ to $A_0 + \delta A$ with $\|\delta A\|_{\text{F}} \leq \delta \|A_0\|_{\text{F}}$, at most $\delta^2/(1-\delta)^2$ of the energy of the probes lies outside the row space of $A_0$.*

A random $r$-dimensional projection keeps a fraction $r/n$ of every direction (proof in Appendix C.6), so it inherits the average curvature $\bar{\lambda}$. Because the gradient concentrates on about $p$ effective directions of the Fisher, the probes that intercept them add an excess that grows as $(r+1)/p$. Since $p \geq 1$, the rank-64/rank-4 ratio of this quantity is $(p+65)/(p+5) \leq 11$ under this model. Aligned probes start with the highest-curvature directions, so each further probe can only add flatter ones and the ordering reverses. Sign-normalized AdamW gives every probe equal energy and makes this dilution strong, whereas SGD weights probes by the curvature they capture.

Part 3 bounds how far a given drift moves the probes, and how large the drift becomes depends on the training regime. Freezing $A$ keeps the probes exactly random, so the rank effect persists; a small drift leaves them nearly random. On a fixed batch the gradient is coherent, the drift is directed and grows linearly with the number of steps, and the probes turn toward the gradient. In online RLVR, where batch gradients are nearly uncorrelated, the drift is a random walk that grows only as the square root of the number of steps and leaves the probes random. The proposition thus predicts that with random probes the rank effect grows with rank and persists when $A$ is frozen (P2), that aligned probes reverse it (P3), and that a faster-moving $A$ attenuates it on a fixed batch but not online (P4).

### 4.3 Utility change is odd in the update direction

A trust region is meant to make improvement safe. In trust-region policy optimization, bounding the KL of each update is what makes the local surrogate objective a reliable guide to improvement (Schulman et al., 2015). The previous subsections show that rank does not provide such a bound. The bound itself, however, controls only how far the policy moves, as the local expansion of utility and KL shows.

**Proposition 4** (Policy movement is locally even; utility change is locally odd)**.** *Let $U(\theta)$ be a twice continuously differentiable expected utility and $u$ a unit direction. For sufficiently small $\tau$,*

$$\begin{aligned} U(\theta + \tau u) - U(\theta) &= \tau \nabla U^\top u + O(\tau^2), \\ K_d(\theta \pm \tau u, \theta) &= \tfrac{1}{2}\tau^2 u^\top H_d u + O(\tau^3). \end{aligned} \tag{6}$$

*Thus the two rays have the same local policy movement to second order but opposite first-order utility change. Writing $H_d = H_d(\theta)$, the first-order utility change $\Delta U^{(1)} = \nabla U^\top \delta$, restricted to the support of $H_d$ under $\delta^\top H_d \delta \le 2\varepsilon$, has extrema $\Delta U^{(1)}_{\max\,/\,\min} = \pm\sqrt{2\varepsilon\, \nabla U^\top H_d^\dagger \nabla U}$; Appendix B gives the derivation.*

A KL budget therefore contains improving and degrading directions of equal size, and which of them training realizes is decided by how the reward gradient aligns with $\nabla U$, so utility should follow the reward rather than the size of the movement (P5). A large KL need not be harmful, and a small one need not be safe.

## 5 Testing the Theory

This mechanism applies only if $A$ stays near its random initialization, so we first check that standard training is in this regime. We then change the probes on single modules and test whether curvature predicts the resulting KL, carry the strongest intervention into online training, and finally ask whether policy movement decides utility.

### 5.1 Standard LoRA training is in the random-probe regime

The rank effect comes from the gradient projected onto the probes, not from the subspace they span. At a common norm, random directions in the same subspace give rank-64/rank-4 fixed-prefix KL ratios of $0.87$–$1.26$, against $8.39$ for the optimizer direction (Appendix C.1). Standard training keeps the probes in place. In every run we checked at the default learning rate, $A$ drifts by at most $5.6\%$ of its norm, and on fixed batches the trained $B$ matches the sign of the gradient projected onto the probes (Appendix C.6).

Part 1 of Proposition 3, on which P2 rests, is derived for SGD, and its random-probe scale carries over quantitatively to the AdamW runs. The row-side second moment of the gradient is nearly proportional to the Fisher, as it assumes (median matrix cosine $0.96$). At the median row-side participation ratio of the adapted modules, $p = 2.8$, it gives a rank-64/rank-4 ratio of $8.7$, and the ratios measured under the row-side Fisher are $7.0$–$8.2$. Projecting the actual gradient onto the actual $A_0$ reproduces the ratio under the full sequence Fisher within a factor of two, for frozen $A$ and, with 48 sequences, for all online seeds. Online, the ratio matches while the realized direction varies with the gradient noise, as expected when the rank effect is a property of the probe subspace (Appendix C.7).

### 5.2 Changing the probes changes the rank effect

The optimizer-coordinate intervention compares a LoRA arm (factor-space AdamW) with a projected arm (weight-space AdamW) whose step is $\delta X_{P,t} = \|\delta X_{L,t}\|_{\mathrm{F}}\, \Pi_r(V_t) / \|\Pi_r(V_t)\|_{\mathrm{F}}$, with $V_t$ the full-weight AdamW proposal and $\Pi_r$ the best rank-$r$ approximation (Eckart & Young, 1936). Both arms start from the same policy and fixed batch and take 30 norm-matched steps (Appendix C.5). Under the default initialization, the projected-to-LoRA KL ratio $G_K(r)$ decreases with rank in all eight Qwen3.5 and eight Granite modules (seed 0). A steeper decrease means a stronger rank effect, because the LoRA arm's KL then grows faster with rank. We change one property of $A$ at a time on three Granite modules and compare each outcome with Proposition 3 (Table 1).

Each intervention in Table 1 moves the rank effect in the direction P2–P4 predict. On the seed-0 batches, freezing $A$ raises $G_K(4)$ from 41–62 to 82–307 (Table 10). Aligning $A_0$ with the leading gradient modes, which carry $54$–$94\%$ of the energy of the initial gradient at rank 4, brings $G_K(4)$ to about 1; the ordering breaks and the Fisher-sensitivity ratio falls below one in all nine module–seed pairs, whereas with random $A_0$ the ordering holds in seven. A larger learning rate for $A$ strongly attenuates the effect on the fixed batch, where the drift is directed.

Table 1: **Interventions on the probes: prediction and measurement.** Single-module Granite interventions on three modules at LoRA displacements within 0.50–1.01 of the default; the default and aligned rows pool three seeds (nine module–seed pairs), the other rows use one seed, and PiSSA and SGD are in Appendix C.8. The last column is the rank-64/rank-4 ratio of the LoRA arm's Fisher sensitivity, and the ordering column counts cases with $G_K(4) > G_K(16) > G_K(64)$.

| Intervention | Change to the probes | Predicted rank effect | $G_K(4)$ | Ordering | $\mu_F$ ratio |
|---|---|---|---|---|---|
| Default | random | grows with rank | 13.8–134.5 | 7/9 | 0.64–19.3 |
| Frozen $A$ | exactly random | grows, no drift | 82–307 | 3/3 | 7.8–27.3 |
| Gradient-aligned $A_0$ | leading gradient modes | reverses | 0.24–1.36 | 0/9 | 0.03–0.34 |
| $A$ learning rate $\times 20$ | coherent drift | attenuates | 1.3–4.5 | 2/3 | 0.21–2.4 |

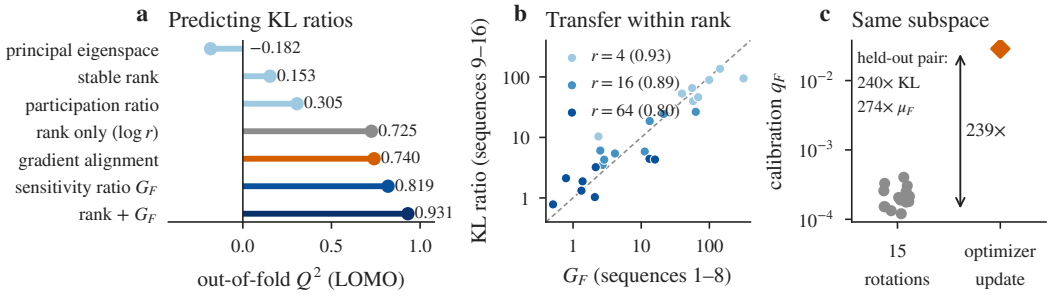



Figure 3: **Directional Fisher sensitivity predicts policy movement.** (a) Out-of-fold $Q^2$ on the same 24 rows and LOMO folds for spectral summaries, rank, the gradient alignment of Section 5.2, $G_F$, and rank combined with $G_F$. (b) $G_F$ against the KL ratio on disjoint sequences, colored by rank, with within-rank correlations in parentheses; the dashed line is $y = x$. (c) Calibration $q_F$ of the optimizer update and 15 rotations of it within the same subspace; the arrow gives the calibration $q_F$ ratio of the selected pair ($239\times$), and the pair differs by $240\times$ in held-out fixed-prefix KL and $274\times$ in endpoint Fisher sensitivity.

Across 54 designs, which include runs at unmatched displacement, the rank effect tracks how the realized update aligns with the gradient. The alignment ratio $\log(\text{align}_P/\text{align}_L)$, with $\text{align}_\cdot = \cos^2(g, \Delta_\cdot)$ for the gradient $g$ and each arm's cumulative update, correlates with $\log G_K$ at 0.92, whereas the initial probe coverage of $g$ gives $-0.63$.

The drift of $A$ separates the two regimes of part 3 of Proposition 3. On a fixed batch its successive increments are nearly parallel (cosine 0.985). In online training, gradients of independent batches are nearly uncorrelated (mean cosine 0.003), and the increments of $A$ are close to a random walk (cosine 0.16–0.46; Appendix C.9).

### 5.3 Curvature along the update predicts policy movement

If curvature sets policy movement (P1), the Fisher sensitivity of an update should predict its KL on sequences it was not estimated on, also within a rank, where the number of probes is fixed. We test this on 24 Granite module–rank rows, estimating Fisher sensitivity on sequences 1–8 and measuring the projected-to-LoRA KL ratio on disjoint sequences 9–16. The predictor is the sensitivity ratio $G_F = \mu_F(\Delta_P; d)/\mu_F(\Delta_L; d)$ of the two arms' cumulative updates, the counterpart of $G_K$. All predictors share the target and LOMO folds.

$G_F$ reaches out-of-fold $Q^2 = 0.819$, against 0.725 for rank alone and at most 0.31 for spectral summaries that ignore the update direction (Figure 3a). Combined with rank it reaches 0.931, and 0.918 on a second target of sequences 65–80 (Appendix C.4). Within each rank, $G_F$ tracks the KL ratio with correlations of 0.80–0.93, against 0.27–0.58 for gradient alignment (Figure 3b). The number of probes thus accounts for the difference between ranks, and the curvature for the differences between updates of equal rank. On eight Qwen3.5 modules, $G_F$ gives $Q^2 = 0.939$ against 0.684 for rank (Appendix C.4).

### 5.4 ALIGNED PROBES REVERSE THE RANK EFFECT IN ONLINE TRAINING

We repeat the displacement-matched rsLoRA experiment of Section 3 with one change. $A_0$ now holds the top-$r$ right singular vectors of the average of eight base-model batch gradients, which approximate the leading row-side modes of the gradient second moment, with default row norms and $B_0 = 0$ (Appendix C.9). Rank 4 should then realize the highest curvature (P3). At a learning rate of $2 \times 10^{-6}$, where KL stays local, the rank-64/rank-4 on-policy KL ratio at matched displacement is $11.3$ with random $A_0$ and $0.58$ on average with aligned $A_0$, below one in two of three seeds and equal to one in the third (Figure 2c). The base-model Fisher matches the mean ratio of both initializations, and each seed within a factor of two.

At the default learning rate, on-policy KL with aligned $A_0$ reaches 6–13 nats per sequence. The aligned KL ratio is then $0.84$–$1.67$, against $9.3$–$13.8$ with random $A_0$, and the Fisher-sensitivity ratio of the trained policies is below one in all three seeds ($0.22$–$0.61$). Raising the learning rate of $A$ $20\times$ instead leaves the rank effect in place, as P4 predicts: online the drift of $A$ is a random walk and the probes stay random (Appendix C.9).

### 5.5 UTILITY FOLLOWS THE REWARD

Proposition 4 addresses the second role of a trust region: even a controlled KL does not determine whether a change helps (P5). If policy movement decided utility, the updates that move the policy more would hurt more. On 512 fixed general-text sequences, standard and rsLoRA rank growth and the high-sensitivity member of the orientation pair all move the policy more, yet all reduce FP32 NLL, with paired intervals excluding zero. Adding a length term to the reward, $R_\beta = R_{\text{acc}} + \beta\, \ell / T_{\max}$ with $\beta : 0 \to 2$, instead raises NLL and lowers MMLU choice margins in both seeds (Appendix C.12).

Large movement can also come without loss. The 200-round rank-256/rank-4 pair reaches $1039\times$ higher $q_F$ and $74.9\times$ higher held-out on-policy KL. Overall GSM8K accuracy rises by $7.20$ points, largely because rank 256 truncates fewer responses ($1.1\%$ against $13.3\%$); on the 1138 prompts both complete, accuracy changes by $-0.26$ points (95% CI $[-1.41, +0.88]$; Figure 5).

The parity itself shows along an update ray $\theta_0 + \tau\Delta$ of a trained Qwen2.5-1.5B-Instruct update (Figure 4). Relative to the base model, GSM8K accuracy rises by $4.25$ points at $\tau = +2$ and falls by $13.04$ points at $\tau = -2$, so the change is dominated by its odd part (Appendix C.13).

## 6 IMPLICATIONS FOR LORA-BASED RLVR

Rank should be chosen for capacity, on which reports for RL differ (Schulman & Thinking Machines Lab, 2025; Yin et al., 2026; Khan et al., 2026), rather than used as a trust-region parameter: it acts on KL only through the directions it realizes, which also depend on the probes, the learning rate of $A$, and the coherence of the gradient. At matched weight norm, where the norm carries no information about the KL gap, directional Fisher sensitivity predicts policy movement from a few dozen sequences. Stability should therefore be controlled directly in distribution space, through a KL penalty or a natural-gradient step of fixed Fisher norm (Amari, 1998; Schulman et al., 2015); the penalty lowers KL at nearly unchanged displacement (Appendix C.10). The meaning of rank also depends on initialization: probes aligned with the gradient make low rank the most sensitive setting (Appendix C.9), whereas PiSSA (Meng et al., 2024), which aligns the probes with weight directions, behaves like the default under AdamW. Finally, policy movement does not determine utility, which should be measured outside the training reward, net of truncation changes (Section 5.5).

## 7 CONCLUSION

LoRA rank is not a policy-space trust region. Rank sets how many random probes intercept the gradient, and the curvature along the realized directions, rather than weight displacement, sets their KL cost. Aligning the probes reverses the rank effect, directional Fisher sensitivity predicts held-out movement beyond rank, and utility follows the reward rather than the size of the movement. Capacity, policy movement, and utility are thus distinct roles in LoRA-based RLVR, as shown here for mathematical reasoning with Qwen3.5-4B, Qwen2.5, and Granite.

### Reproducibility Statement

Section 2 summarizes models, controls, sample sizes, and repeated runs. The appendix gives complete derivations, estimator definitions, and training and evaluation settings. Code, configurations, and result summaries will be released in a public repository.

### AI Use Statement

Generative AI tools assisted with literature discovery, experiment-protocol critique, interpretation of results, mathematical derivation and proof checking, analysis-code development, scientific-figure design, and manuscript drafting and editing. The authors made all final research decisions, executed the experiments, verified the generated code and figures against the saved artifacts, checked the mathematical statements and citations, and reviewed the final manuscript. No synthetic experimental results were used. The authors take responsibility for all claims, text, code, and artifacts in this work.

### Ethics Statement

This work studies how LoRA rank affects policy movement when publicly available language models are fine-tuned with reinforcement learning. It involves no human subjects, annotation, or personal data. All models (Qwen3.5-4B, Qwen2.5-1.5B-Instruct, and Granite-4.2-3B) and evaluation data (GSM8K, MATH-500, IFEval, MMLU, and WikiText-103) are publicly released and used within their research licenses, and we release no new models or datasets. The findings bear on the safety of fine-tuning: a low rank does not keep a policy close to its initialization, and a misspecified reward term can degrade general-text behavior (Section 5.5). Practitioners who rely on rank for stability should therefore monitor and constrain policy movement directly.

## References

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

Dan Biderman, Jacob 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. LoRA learns less and forgets less. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=aloEru2qCG.

Franz Louis Cesista, Katherine Crowson, Cédric Simal, and Stella Biderman. LoRA-Muon: Spectral steepest descent on the low-rank manifold. *arXiv preprint arXiv:2606.12921*, 2026. URL https://arxiv.org/abs/2606.12921.

Nan Chen, Soledad Villar, and Soufiane Hayou. Learning rate scaling across LoRA ranks and transfer to full finetuning. *arXiv preprint arXiv:2602.06204*, 2026. URL https://arxiv.org/abs/2602.06204.

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.

Laurent Dinh, Razvan Pascanu, Samy Bengio, and Yoshua Bengio. Sharp minima can generalize for deep nets. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1019–1028. PMLR, 2017. URL https://proceedings.mlr.press/v70/dinh17b.html.

Carl Eckart and Gale Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.

Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Firdaus Janoos, Larry Rudolph, and Aleksander Madry. Implementation matters in deep RL: A case study on PPO and TRPO. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=r1etN1rtPB.

Zhi-Quan Feng, Ying-Jia Lin, and Hung-Yu Kao. Learning in the fisher subspace: A guided initialization for LoRA fine-tuning. *arXiv preprint arXiv:2605.01046*, 2026. URL https://arxiv.org/abs/2605.01046.

Yongchang Hao, Yanshuai Cao, and Lili Mou. Flora: Low-rank adapters are secretly gradient compressors. In *International Conference on Machine Learning*, 2024.

Soufiane Hayou, Nikhil Ghosh, and Bin Yu. LoRA+: Efficient low rank adaptation of large models. In *International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 17783–17806, 2024. URL https://proceedings.mlr.press/v235/hayou24a.html.

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*, 2021a.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Advances in Neural Information Processing Systems Datasets and Benchmarks Track*, 2021b.

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. In *International Conference on Learning Representations*, 2022.

IBM Granite Team. Granite 4.2 language models. https://github.com/ibm-granite/granite-4.2-language-models, 2026.

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

Zohaib Khan, Omer Tafveez, and Zoha Hayat Bhatti. Plasticity vs. rigidity: The impact of low-rank adapters on reasoning on a micro-budget. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pp. 493–501, 2026. URL https://aclanthology.org/2026.eacl-srw.37/.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, 2024.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. PiSSA: Principal singular values and singular vectors adaptation of large language models. In *Advances in Neural Information Processing Systems*, 2024.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017.

Sagnik Mukherjee, Lifan Yuan, Dilek Hakkani-Tür, and Hao Peng. Reinforcement learning finetunes small subnetworks in large language models. In *Advances in Neural Information Processing Systems*, volume 38, pp. 132119–132138, 2025.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024. URL https://arxiv.org/abs/2412.15115.

Qwen Team. Qwen3.5: Towards native multimodal agents. https://qwen.ai/blog?id=qwen3.5, February 2026.

John Schulman and Thinking Machines Lab. LoRA without regret. Thinking Machines Lab: Connectionism, 2025.

John Schulman, Sergey Levine, Philipp Moritz, Michael I. Jordan, and Pieter Abbeel. Trust region policy optimization. In *International Conference on Machine Learning*, 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.

Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. RL's razor: Why online reinforcement learning forgets less. In *International Conference on Learning Representations*, 2026.

Reece Shuttleworth, Jacob Andreas, Antonio Torralba, and Pratyusha Sharma. LoRA vs full fine-tuning: An illusion of equivalence. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/ff541950d1e885af90f523571564a401-Abstract-Conference.html.

Shaowen Wang, Linxi Yu, and Jian Li. LoRA-GA: Low-rank adaptation with gradient approximation. In *Advances in Neural Information Processing Systems*, volume 37, pp. 54905–54931, 2024.

Yuhui Wang, Hao He, and Xiaoyang Tan. Truly proximal policy optimization. In *Proceedings of the 35th Uncertainty in Artificial Intelligence Conference*, volume 115 of *Proceedings of Machine Learning Research*, pp. 113–122. PMLR, 2020. URL https://proceedings.mlr.press/v115/wang20b.html.

Zhengbo Wang, Jian Liang, Ran He, Zilei Wang, and Tieniu Tan. LoRA-Pro: Are low-rank adapters properly optimized? In *International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=gTwRMU3lJ5.

Zhuxuanzi Wang, Mingqiao Mo, Xi Xiao, Chen Liu, Chenrui Ma, Yunbei Zhang, Xiao Wang, Smita Krishnaswamy, and Tianyang Wang. CTR-LoRA: Curvature-aware and trust-region guided low-rank adaptation for large language models. *arXiv preprint arXiv:2510.15962*, 2025b. URL https://arxiv.org/abs/2510.15962.

Chenrui Wu, Zexi Li, Jiajun Bu, Jiangchuan Liu, and Haishuai Wang. Sparsity curse: Understanding RLVR model parameter space from model merging. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2026. URL https://arxiv.org/abs/2606.18521.

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. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2512.23165.

Fangzhao Zhang and Mert Pilanci. Riemannian preconditioned LoRA for fine-tuning foundation models. In *International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 59641–59669, 2024. URL https://proceedings.mlr.press/v235/zhang24ax.html.

Jiaying Zhang, Lei Shi, Jiguo Li, Jun Xu, Jiuchong Gao, Jinghua Hao, and Renqing He. GeoRA: Geometry-aware low-rank adaptation for RLVR. In *Annual Meeting of the Association for Computational Linguistics*, 2026a. URL https://arxiv.org/abs/2601.09361.

Longteng Zhang, Lin Zhang, Shaohuai Shi, Xiaowen Chu, and Bo Li. LoRA-FA: Efficient and effective low rank representation fine-tuning. *arXiv preprint arXiv:2308.03303*, 2023.

Ruijia Zhang, Jiacheng Zhu, Hanqing Zhu, and Laixi Shi. Geometry-preserving orthonormal initialization for low-rank adaptation in RLVR. In *International Conference on Machine Learning*, 2026b. URL https://arxiv.org/abs/2606.31813.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

Hanqing Zhu, Zhenyu Zhang, Hanxian Huang, DiJia Su, Zechun Liu, Jiawei Zhao, Igor Fedorov, Hamed Pirsiavash, Zhizhou Sha, Jinwon Lee, David Z. Pan, Zhangyang Wang, Yuandong Tian, and Kai Sheng Tai. The path not taken: RLVR provably learns off the principals. *arXiv preprint arXiv:2511.08567*, 2025. URL https://arxiv.org/abs/2511.08567.

Jiacheng Zhu, Kristjan Greenewald, Kimia Nadjahi, Haitz Sáez de Ocáriz Borde, Rickard Brüel Gabrielsson, Leshem Choshen, Marzyeh Ghassemi, Mikhail Yurochkin, and Justin Solomon. Asymmetry in low-rank adapters of foundation models. In *International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 62369–62385, 2024. URL https://proceedings.mlr.press/v235/zhu24c.html.

# A Extended Related Work

**LoRA optimization geometry.** LoRA dynamics depend on rank scaling and factor learning rates (Hu et al., 2022; Kalajdzievski, 2023; Hayou et al., 2024; Chen et al., 2026), and equivalent-gradient, Riemannian, and spectral steepest-descent analyses characterize and modify the weight-space update that the factors induce (Wang et al., 2025a; Zhang & Pilanci, 2024; Cesista et al., 2026). When $A$ stays near initialization, LoRA acts as a random projection of the gradient (Hao et al., 2024; Zhu et al., 2024; Zhang et al., 2023), and initializations from weight or gradient singular vectors change that projection (Meng et al., 2024; Wang et al., 2024). Matching an aggregate quantity does not make two adaptations equivalent: LoRA and full fine-tuning that reach the same accuracy differ in spectral structure and forgetting (Biderman et al., 2024; Shuttleworth et al., 2025). We identify the projection regime in RLVR and show that it, rather than capacity, sets how policy movement depends on rank.

**Rank and stability in RLVR.** RLVR studies connect initialization, subspaces, rank, and adapter structure to stability and reasoning outcomes (Zhang et al., 2026a;b; Yin et al., 2026; Khan et al., 2026), and reports on rank differ: low ranks can match full fine-tuning when learning rates and adapter placement are set appropriately (Schulman & Thinking Machines Lab, 2025), whereas very low ranks limit reasoning gains in other settings (Yin et al., 2026; Khan et al., 2026). These results are often read as low rank constraining how far the policy moves. We test this reading at matched displacement and use coordinate interventions to identify its optimizer dependence.

**Trust regions and parameter-space measures.** Natural gradients and trust-region methods constrain policy updates in distribution space (Amari, 1998; Schulman et al., 2015; 2017), and Fisher information has guided LoRA initialization and curvature-aware trust regions (Feng et al., 2026; Wang et al., 2025b). Mechanisms assumed to act as trust regions can fail to do so: PPO's ratio clipping does not bound the probability ratio (Wang et al., 2020; Engstrom et al., 2020), and the KL trust region PPO maintains in practice depends on implementation-level optimizations rather than on the clipping (Engstrom et al., 2020). Parameter-space quantities can also fail to track function-space behavior, as sharpness does under reparameterization (Dinh et al., 2017). For RLVR, Zhu et al. (2025) derive a local Fisher-norm constraint from a policy-KL budget and show that updates lie off the principal directions, a bias that also explains the apparent sparsity of RLVR updates (Mukherjee et al., 2025; Wu et al., 2026), and Shenfeld et al. (2026) relate forgetting to KL on the new-task distribution. We show that rank is such a parameter-space quantity, identify the directional curvature that replaces it, and ask whether the magnitude of KL on a fixed state measure fixes the sign of utility.

# B Theoretical Details

## B.1 Factor-space update geometry

Section 4.2 uses the following two statements.

**Proposition 5** (Factor-SGD local pushforward and effective preconditioner)**.** *Under factor-space SGD, the first-order effective update is*

$$\delta X_{\mathrm{LoRA}} = -\eta \mathcal{M}_r[g], \qquad \mathcal{M}_r[g] = J_\phi J_\phi^\top[g] = \gamma_r^2 (BB^\top g + gA^\top A). \tag{7}$$

*The positive semidefinite (PSD) operator $\mathcal{M}_r$ is the effective preconditioner that maps a weight-space gradient to the local first-order update. Moreover, on the tangent space $\mathcal{T}_X$,*

$$\begin{aligned} \delta X_{\mathrm{LoRA}} &= \arg\min_{\Delta \in \mathcal{T}_X} \left\{ \langle g, \Delta \rangle + \frac{1}{2\eta} \langle \Delta, \mathcal{M}_r^\dagger \Delta \rangle \right\}, \\ \delta X_{\mathrm{proj}} &= \arg\min_{\Delta \in \mathcal{T}_X} \left\{ \langle g, \Delta \rangle + \frac{1}{2\eta} \|\Delta\|_{\mathrm{F}}^2 \right\} = -\eta P_{\mathcal{T}_X} g. \end{aligned} \tag{8}$$

*Thus $\mathcal{M}_r^\dagger$ defines the induced tangent-space quadratic metric, while projected weight-space SGD uses the Euclidean metric. The two methods perform steepest descent under different metrics on the same feasible tangent space.*

**Proposition 6** (Optimizer–Fisher generalized Rayleigh identity)**.** *At a fixed score-evaluation point, let $\mathcal{M}$ be a linear effective update map with $\mathcal{M}g \neq 0$ and $u_{\mathcal{M}} = \mathcal{M}g/\left\|\mathcal{M}g\right\|_{\mathrm{F}}$. Then*

$$\mu_F(u_{\mathcal{M}}; d) = \frac{g^\top \mathcal{M}^\top F_d \mathcal{M} g}{g^\top \mathcal{M}^\top \mathcal{M} g}. \tag{9}$$

*Writing $F_{\mathcal{M}} = \mathcal{M}^\top F_d \mathcal{M}$ and $N_{\mathcal{M}} = \mathcal{M}^\top \mathcal{M}$, the attainable sensitivity lies between the generalized eigenvalues of $(F_{\mathcal{M}}, N_{\mathcal{M}})$ restricted to $\ker(\mathcal{M})^\perp = \mathrm{range}(N_{\mathcal{M}})$.*

### B.2 Notation and regularity assumptions

We identify a weight matrix with the Euclidean space $\mathcal{H}_W = \mathbb{R}^{d_{\mathrm{out}} \times d_{\mathrm{in}}}$ equipped with the Frobenius inner product $\langle X, Y \rangle = \mathrm{tr}(X^\top Y)$. The factor space is the product Hilbert space

$$\mathcal{H}_\phi = \mathbb{R}^{d_{\mathrm{out}} \times r} \times \mathbb{R}^{r \times d_{\mathrm{in}}}, \qquad \langle (B_1, A_1), (B_2, A_2) \rangle = \langle B_1, B_2 \rangle + \langle A_1, A_2 \rangle. \tag{10}$$

The Jacobian and local first-order pushforward identities below hold for arbitrary factor states with $\gamma_r \neq 0$. For the fixed-rank manifold, tangent-space, and metric statements, we additionally assume that $B \in \mathbb{R}^{d_{\mathrm{out}} \times r}$ and $A \in \mathbb{R}^{r \times d_{\mathrm{in}}}$ have rank $r$, so $X = \gamma_r BA$ is a smooth point of the rank-$r$ matrix manifold. At a singular factor state, including a zero-factor initialization such as $B = 0$, the Jacobian and $J_\phi J_\phi^\top$ identities remain valid, while the smooth rank-$r$ tangent and metric interpretation applies only after the factors reach a full-rank state. The prefix occupancy measure $d$ and the sequence batch used for empirical Fisher measurement are held fixed. For the KL expansions, the policy is three-times differentiable along the stated parameter ray, differentiation may be interchanged with integration against $d$, and the reference policy has common support with nearby policies.

**Conditional KL curvature and empirical sequence Fisher.** For frozen completions $\mathcal{D} = \{(s_{it}, y_{it})\}$, the evaluation measure is $d = N^{-1} \sum_i \sum_{t=1}^{L_i} \delta_{s_{it}}$. Consequently,

$$K_d(\theta', \theta) = \frac{1}{N} \sum_{i=1}^{N} \sum_{t=1}^{L_i} D_{\mathrm{KL}}(\pi_{\theta'}(\cdot \mid s_{it}) \| \pi_\theta(\cdot \mid s_{it}))\,. \tag{11}$$

Its mass is the mean completion length $\bar{L}$. If $\bar{d} = d/\bar{L}$ is the normalized prefix distribution, then $K_d = \bar{L} K_{\bar{d}}$. Writing $a_\theta(s, y) = \nabla_\theta \log \pi_\theta(y \mid s)$, the Hessian at $\theta' = \theta$ is

$$H_d(\theta) = \frac{1}{N} \sum_{i,t} \sum_{y} \pi_\theta(y \mid s_{it}) a_\theta(s_{it}, y) a_\theta(s_{it}, y)^\top. \tag{12}$$

This full-vocabulary conditional Fisher determines the local expansion of $K_d$.

The measured sequence Fisher instead uses the observed completion tokens:

$$\widehat{F}_{\mathcal{D}}(\theta_*) = \frac{1}{N} \sum_i z_i(\theta_*) z_i(\theta_*)^\top, \qquad z_i(\theta_*) = \sum_t a_{\theta_*}(s_{it}, y_{it}). \tag{13}$$

We abbreviate this matrix as $F_d$ in the empirical Rayleigh and spectral formulas. The implemented quadratic form is exactly $q_F(\Delta; d) = N^{-1} \sum_i (z_i^\top \Delta)^2$ for the frozen sample, up to numerical precision. Sequence-score outer products contain cross-token terms. Under on-policy sequence sampling at $\theta_*$, conditional zero-mean scores cancel these terms in expectation, recovering the expected sum of token Fishers. A finite frozen reference batch evaluated at an adapted endpoint gives a distinct empirical matrix. In the optimizer-coordinate comparison both cumulative directions are evaluated at the common reference model; trained adapters are each evaluated at their own endpoint on identical frozen sequences.

### B.3 Proof of Proposition 5

We divide the proof into the Jacobian calculation, the effective preconditioner, and the variational characterization.

**Step 1: differential and adjoint.** For perturbations $(\delta B, \delta A) \in \mathcal{H}_\phi$,

$$\gamma_r(B+\delta B)(A+\delta A) - \gamma_r BA = \gamma_r(\delta BA + B\delta A) + \gamma_r \delta B \delta A. \tag{14}$$

The linear term defines

$$J_\phi(\delta B, \delta A) = \gamma_r(\delta BA + B\delta A). \tag{15}$$

For any $g \in \mathcal{H}_W$, cyclicity of trace gives

$$\langle J_\phi(\delta B, \delta A), g\rangle = \gamma_r \operatorname{tr}(A^\top \delta B^\top g) + \gamma_r \operatorname{tr}(\delta A^\top B^\top g) \tag{16}$$

$$= \langle \delta B, \gamma_r g A^\top\rangle + \langle \delta A, \gamma_r B^\top g\rangle. \tag{17}$$

Hence the adjoint is

$$J_\phi^\top g = (\gamma_r g A^\top, \gamma_r B^\top g). \tag{18}$$

**Step 2: effective SGD preconditioner.** Factor-space SGD uses $(\delta B, \delta A) = -\eta J_\phi^\top g$. Applying $J_\phi$ and using Equation 18,

$$J_\phi J_\phi^\top[g] = \gamma_r\left[(\gamma_r g A^\top)A + B(\gamma_r B^\top g)\right] \tag{19}$$

$$= \gamma_r^2\left(gA^\top A + BB^\top g\right). \tag{20}$$

Therefore the first-order effective update is $\delta X_{\mathrm{LoRA}} = -\eta \mathcal{M}_r[g]$ with $\mathcal{M}_r = J_\phi J_\phi^\top$, proving Equation 7. Thus $\mathcal{M}_r$ is the PSD effective preconditioner that maps the weight-space gradient to the local first-order update.

**Step 3: range and positive definiteness on the tangent.** The image of $J_\phi$ is

$$\operatorname{Im} J_\phi = \{\gamma_r(\delta BA + B\delta A) : (\delta B, \delta A) \in \mathcal{H}_\phi\} = \mathcal{T}_X. \tag{21}$$

For a finite-dimensional linear map, $\operatorname{range}(J_\phi J_\phi^\top) = \operatorname{range}(J_\phi)$; thus $\operatorname{range}(\mathcal{M}_r) = \mathcal{T}_X$. Moreover,

$$\langle \Delta, \mathcal{M}_r \Delta\rangle = \langle \Delta, J_\phi J_\phi^\top \Delta\rangle = \|J_\phi^\top \Delta\|^2 \geq 0. \tag{22}$$

If $\Delta \in \mathcal{T}_X$ and the right-hand side is zero, then $\Delta \in \ker J_\phi^\top = (\operatorname{Im} J_\phi)^\perp = \mathcal{T}_X^\perp$. Hence $\Delta = 0$. Therefore $\mathcal{M}_r$ is positive definite after restriction to $\mathcal{T}_X$, although it is generally singular on the ambient weight space. Its pseudoinverse $\mathcal{M}_r^\dagger$ therefore defines the tangent-space quadratic metric used below.

**Step 4: variational characterization.** Consider

$$\Phi(\Delta) = \langle g, \Delta\rangle + \frac{1}{2\eta}\langle \Delta, \mathcal{M}_r^\dagger \Delta\rangle, \qquad \Delta \in \mathcal{T}_X. \tag{23}$$

Because $\mathcal{M}_r^\dagger$ is positive definite on $\mathcal{T}_X$, $\Phi$ is strictly convex there. Its directional derivative in any $v \in \mathcal{T}_X$ is

$$D\Phi(\Delta)[v] = \left\langle P_{\mathcal{T}_X} g + \eta^{-1}\mathcal{M}_r^\dagger \Delta, v\right\rangle. \tag{24}$$

The unique stationary point therefore satisfies

$$P_{\mathcal{T}_X} g + \eta^{-1}\mathcal{M}_r^\dagger \Delta = 0. \tag{25}$$

Multiplying by $\mathcal{M}_r$, using $\mathcal{M}_r P_{\mathcal{T}_X} = \mathcal{M}_r$ and $\mathcal{M}_r \mathcal{M}_r^\dagger \Delta = \Delta$ for $\Delta \in \mathcal{T}_X$, yields

$$\Delta = -\eta \mathcal{M}_r g. \tag{26}$$

This proves the LoRA line of Equation 8. Replacing the quadratic term by $\|\Delta\|_{\mathrm{F}}^2/(2\eta)$ gives the stationarity condition $P_{\mathcal{T}_X} g + \Delta/\eta = 0$, hence $\Delta = -\eta P_{\mathcal{T}_X} g$, which proves the projected line of Equation 8. Thus the two updates optimize the same first-order loss over the same tangent space but under the metrics $\mathcal{M}_r^\dagger$ and $I$, respectively.

### B.4 Spectrum of the factor-SGD effective preconditioner

Let $BB^\top p_i = b_i^2 p_i$ and $A^\top A v_j = a_j^2 v_j$. Since left and right multiplication commute,

$$\mathcal{M}_r[p_i v_j^\top] = \gamma_r^2 (b_i^2 + a_j^2) p_i v_j^\top. \tag{27}$$

Equivalently, under column-major vectorization,

$$\mathrm{vec}(\mathcal{M}_r[Z]) = \gamma_r^2 \left( I_{d_{\mathrm{in}}} \otimes BB^\top + A^\top A \otimes I_{d_{\mathrm{out}}} \right) \mathrm{vec}(Z). \tag{28}$$

Thus the effective spectrum is a Kronecker sum of the two factor Gram spectra. Directions with $b_i = a_j = 0$ are normal directions and have zero eigenvalue.

### B.5 Proof of Proposition 6

Let $\mathcal{M} : \mathcal{H}_W \to \mathcal{H}_W$ be linear and let $g \notin \ker \mathcal{M}$. Vectorizing matrices if necessary, define

$$F_{\mathcal{M}} = \mathcal{M}^\top F_d \mathcal{M}, \qquad N_{\mathcal{M}} = \mathcal{M}^\top \mathcal{M}. \tag{29}$$

Substituting $u_{\mathcal{M}} = \mathcal{M}g / \|\mathcal{M}g\|_{\mathrm{F}}$ gives

$$\mu_F(u_{\mathcal{M}}; d) = \frac{(\mathcal{M}g)^\top F_d (\mathcal{M}g)}{\|\mathcal{M}g\|_{\mathrm{F}}^2} \tag{30}$$

$$= \frac{g^\top F_{\mathcal{M}} g}{g^\top N_{\mathcal{M}} g}. \tag{31}$$

Now let $\mathcal{R} = \ker(\mathcal{M})^\perp$. In finite dimensions, $\mathcal{R} = \mathrm{range}(\mathcal{M}^\top) = \mathrm{range}(N_{\mathcal{M}})$. For every nonzero $z \in \mathcal{R}$,

$$z^\top N_{\mathcal{M}} z = \|\mathcal{M}z\|_{\mathrm{F}}^2 > 0, \tag{32}$$

so $N_{\mathcal{M}}|_{\mathcal{R}}$ is positive definite. Whiten with $y = N_{\mathcal{M}}^{1/2} z$ on $\mathcal{R}$ to obtain

$$\frac{z^\top F_{\mathcal{M}} z}{z^\top N_{\mathcal{M}} z} = \frac{y^\top \left( N_{\mathcal{M}}^{-1/2} F_{\mathcal{M}} N_{\mathcal{M}}^{-1/2} \right) y}{y^\top y}. \tag{33}$$

The ordinary Rayleigh bound for the symmetric PSD matrix in parentheses gives the generalized-eigenvalue interval stated in Proposition 6.

For any nonzero scalar $a$, $F_{a\mathcal{M}} = a^2 F_{\mathcal{M}}$ and $N_{a\mathcal{M}} = a^2 N_{\mathcal{M}}$; therefore the quotient and all generalized eigenvalues are invariant. More generally, if $F_{\mathcal{M}} = \lambda N_{\mathcal{M}}$ on a gradient subspace, every nonzero update generated from that subspace has the same normalized sensitivity $\lambda$.

### B.6 Proof of Proposition 2

For $Q \in O(r)$ define the normalized orientation

$$\widehat{u}(Q) = \frac{BQA}{\|BQA\|_{\mathrm{F}}}. \tag{34}$$

Because $Q$ is invertible and $A, B$ have rank $r$,

$$\mathrm{col}(BQ) = \mathrm{col}(B), \qquad \mathrm{row}(QA) = \mathrm{row}(A). \tag{35}$$

Left multiplication by the full-column-rank matrix $B$ does not change the row space of $QA$, and scalar normalization changes neither subspace. Hence all members of the normalized orbit have identical rank, norm, column space, and row space.

Let a subspace-only predictor be

$$\widehat{Y}(u) = f(\mathrm{col}(u), \mathrm{row}(u)). \tag{36}$$

For the selected high and low orientations, the invariants coincide, so $\widehat{Y}(u_h) = \widehat{Y}(u_l) = c$. Assume $Y_h \geq Y_l > 0$. Multiplicative factor-$C$ accuracy on both points means

$$\frac{Y_h}{C} \leq c \leq C Y_h, \qquad \frac{Y_l}{C} \leq c \leq C Y_l. \tag{37}$$

For these intervals to intersect, their inner endpoints must satisfy

$$\frac{Y_h}{C} \le CY_l. \tag{38}$$

Therefore

$$C^2 \ge \frac{Y_h}{Y_l}, \qquad C \ge \sqrt{\frac{Y_h}{Y_l}}. \tag{39}$$

This lower bound is tight for the two-point problem: choosing $c = \sqrt{Y_h Y_l}$ achieves worst-case multiplicative factor $\sqrt{Y_h/Y_l}$. Substituting the measured ratios gives $C \ge 15.48$ for held-out fixed-prefix KL and $C \ge 16.56$ for directional Fisher sensitivity.

### B.7 Proof of Proposition 1

At a fixed reference $\theta$, let $H_d = H_d(\theta)$ and $h_d(u) = u^\top H_d u$ for a unit direction $u$. The local expansion is

$$K_d(\theta + Ru, \theta) = \tfrac{1}{2} R^2 h_d(u) + O(R^3). \tag{40}$$

Solving the quadratic budget $\frac{1}{2}R^2 h_d(u) = \varepsilon$ gives $R(u) = \sqrt{2\varepsilon/h_d(u)}$ for positive curvature. For two directions at this reference,

$$\frac{R_l}{R_h} = \sqrt{\frac{h_d(u_h)}{h_d(u_l)}}. \tag{41}$$

Thus a spherical boundary cannot represent unequal directional quadratic-KL boundaries.

For the empirical orientation comparison, let $F_h = \widehat{F}_{\mathcal{D}}(\theta_h)$ and $F_l = \widehat{F}_{\mathcal{D}}(\theta_l)$, where $\theta_i$ is the corresponding adapted endpoint. Define the endpoint sensitivity $\hat{\mu}_i = \widehat{u}_i^\top F_i \widehat{u}_i$ and the empirical radius $\widehat{R}_i = \sqrt{2\varepsilon/\hat{\mu}_i}$. The measured ratio gives

$$\frac{\widehat{R}_l}{\widehat{R}_h} = \sqrt{\frac{\hat{\mu}_h}{\hat{\mu}_l}} = \sqrt{274.2} \simeq 16.6. \tag{42}$$

This compares two endpoint empirical quadratic forms. A condition-number bound for a single matrix would require both Rayleigh quotients to use that same matrix.

### B.8 Proof of Proposition 4

Let $v = \nabla U(\theta)$ and $H_d = H_d(\theta)$. Taylor expansion yields

$$U(\theta + \tau u) - U(\theta) = \tau v^\top u + O(\tau^2), \tag{43}$$

$$U(\theta - \tau u) - U(\theta) = -\tau v^\top u + O(\tau^2). \tag{44}$$

For KL, the zero first derivative and Fisher Hessian imply

$$K_d(\theta + \tau u, \theta) = \frac{1}{2}\tau^2 u^\top H_d u + O(\tau^3), \tag{45}$$

$$K_d(\theta - \tau u, \theta) = \frac{1}{2}\tau^2 u^\top H_d u + O(\tau^3). \tag{46}$$

Thus policy movement is locally even in direction while first-order utility is odd.

It remains to solve the KL-constrained first-order utility problem. A component of $v$ in $\ker H_d$ is not controlled by the quadratic KL budget, so the finite extremum statement is made on $\mathcal{R} = \mathrm{range}(H_d)$, or equivalently after projecting $v$ onto $\mathcal{R}$. On this subspace,

$$v^\top \delta = \left(H_d^{\dagger/2} v\right)^\top \left(H_d^{1/2} \delta\right) \tag{47}$$

$$\le \sqrt{v^\top H_d^\dagger v}\sqrt{\delta^\top H_d \delta} \tag{48}$$

$$\le \sqrt{2\varepsilon\, v^\top H_d^\dagger v}, \tag{49}$$

where the first inequality is Cauchy–Schwarz. If $v^\top H_d^\dagger v > 0$, equality is attained by

$$\delta_\star^\pm = \pm\sqrt{\frac{2\varepsilon}{v^\top H_d^\dagger v}} H_d^\dagger v. \tag{50}$$

Substitution shows

$$v^\top \delta_\star^\pm = \pm\sqrt{2\varepsilon\, v^\top H_d^\dagger v}, \tag{51}$$

which proves the extremum formula in Proposition 4. If the projected gradient is zero, both extrema are zero to first order. Without the Fisher-support restriction, a nonzero utility-gradient component in $\ker H_d$ would make the first-order problem unbounded, which is why the support condition is essential.

# C Additional Experimental Details

## C.1 Protocol matrix and statistical units

Table 2: **Experimental protocols.** What each protocol holds fixed determines which claim it can test.

| Protocol | Models | Held fixed | Used for |
|---|---|---|---|
| Online GRPO | Qwen3.5-4B; Qwen2.5-1.5B-Instruct | model, task, and schedule; effective-weight displacement in the rsLoRA runs | KL at matched displacement (§3); aligned-probe reversal (§5.4); utility (§5.5) |
| Fixed batch, one module | Qwen3.5-4B and Granite-4.2-3B, eight modules each | policy, batch, and step norm; only the parameterization differs | probe regime and interventions (§5.1, §5.2); curvature prediction (§5.3) |
| Trained updates, no further training | Qwen3.5-4B rank 64; Qwen2.5-1.5B-Instruct | rank, norm, and subspace, or the line through $\Delta$ | orientation pair (§3); update rays (§5.5) |

**Evaluation and uncertainty.** Generation evaluation uses GSM8K (Cobbe et al., 2021) ($n = 1319$, 512/1024-token caps), MATH-500 (Hendrycks et al., 2021b; Lightman et al., 2024) ($n = 500$, 1024/2048-token caps), and IFEval (Zhou et al., 2023) ($n = 541$, 1024-token cap), with shared prompts and greedy decoding. *Common-support conditional accuracy* compares two conditions on the prompts both complete without truncation; condition-specific nontruncated accuracy uses each condition's own completions. The *anti-ray* compares $\tau = -2$ with the trained endpoint $\tau = +1$. Continuous out-of-distribution (OOD) evaluation uses FP32 negative log-likelihood (NLL) on 512 fixed texts and Massive Multitask Language Understanding (MMLU) choice margins on 1000 items (Hendrycks et al., 2021a). Item comparisons are paired. Training runs report three raw seeds and sample standard deviation. Prediction uses grouped leave-one-module-out (LOMO) and leave-one-rank-out (LORO) cross-validation, scored by out-of-fold $Q^2 = 1 - \sum_j (y_j - \hat{y}_j)^2 / \sum_j (y_j - \bar{y})^2$ on log ratios, with $\hat{y}_j$ fit without row $j$'s fold.

The held-out FP32 fixed-prefix estimator is a full-vocabulary conditional KL on fixed shared prompt indices. It estimates $K_{d_{\mathrm{ref}}}(\theta, \theta_0)$ using frozen reference-policy prefixes, and is paired by prompt. It is not the sequence-distribution KL $D_{\mathrm{KL}}(P_\theta \| P_{\theta_0})$, whose chain-rule state weighting depends on the adapted policy. Three-seed summaries use arithmetic means and sample standard deviations. Item-level generation and continuous comparisons are paired.

For the fixed-prefix estimator, reference-policy completions are sampled once and then held immutable across conditions. At every completion position we compute $D_{\mathrm{KL}}(\pi_\theta(\cdot \mid s_t) \| \pi_{\theta_0}(\cdot \mid s_t))$ over the full vocabulary, sum over the completion, and average the sequence sums across prompts. This sequence-summed estimator fixes the prefix occupancy measure across conditions. The on-policy estimator uses the same full-vocabulary conditional KL and sequence-sum normalization on each adapted policy's generated prefixes. Held-out refers to prompts or sequences disjoint from those used for training or estimation and applies to either estimator. Tables write $\mathrm{KL}_{\mathrm{on}}$ for on-policy and

Table 3: **Experimental designs and statistical units.** Training seeds are the independent units for multi-run summaries; module and rank rows are finite controlled designs rather than iid replications.

| Experiment | Model/design | Control | Repeats and evaluation |
|---|---|---|---|
| Matched displacement | Qwen3.5-4B, ranks 4/16/64, 30 rounds | rank-stabilized scaling; displacement, coherence, and cumulative step energy | 3 training seeds; rounds 10/20/30; training and held-out FP32 KL |
| Random directions | Qwen3.5-4B, ranks 4/64 | subspace and global families; 16 norm-matched candidates per family/rank/seed | 3 seeds; FP32 fixed-prefix conditional KL and candidate bootstrap |
| Optimizer-coordinate intervention | Qwen3.5 eight modules and Granite eight modules, ranks 4/16/64 | LoRA arm (factor-space AdamW) versus projected arm (weight-space AdamW plus best rank-$r$ SVD); projected step norm matched for 30 optimizer steps | frozen 32-response training batch and 16-sequence FP32 evaluation batch |
| Fisher tracking | Granite eight modules $\times$ three ranks | sequences 1–8 for Fisher; disjoint sequences 9–16 for finite KL | 24 rows; log-ratio ordinary least squares (OLS) with LOMO and LORO refits |
| Orientation pair | Qwen3.5 standard LoRA rank 64 | identity plus 15 seeded orthogonal mixes normalized to the same norm; calibration-prompt $q_F$ selects high/low, then pair is frozen | Set A: 32 train prompts; Set B: 64 test prompts; shared 64-sequence Fisher and full held-out FP32 evaluation |
| Utility assays | Qwen3.5/Qwen2.5 endpoints and rays | paired clean, rank, direction, and length-reward contrasts | NLL $n = 512$; MMLU $n = 1000$; GSM8K $n = 1319$; MATH-500 $n = 500$; IFEval $n = 541$ |
| Long schedule | Qwen3.5 ranks 4/256, 200 rounds | same model, task, reward, response batch, and 400-step cosine schedule | one matched trajectory pair; five quality checkpoints and 6400 trajectories per arm |

$\mathrm{KL}_{\mathrm{prefix}}$ for fixed-prefix KL. The global effective-weight norm is the root-sum-square of module Frobenius norms.

The standard LoRA random-direction control uses 16 nonidentity candidates per family, rank, and seed. In the subspace family, each module applies $B_m \leftarrow B_m Q_m$ with a deterministic orthogonal $Q_m$ obtained by QR factorization of a seeded Gaussian matrix, while $A_m$ is fixed. In the global family, both factors are replaced by independent seeded Gaussian matrices with entry scales $d_{\mathrm{out}}^{-1/2}$ and $d_{\mathrm{in}}^{-1/2}$. Every candidate is rescaled to the same seed-specific rank-64 endpoint norm and evaluated on the same 64 fixed test prompts. The rank-64/rank-4 fixed-prefix KL ratio of the random candidates is $1.26$, $0.87$, and $1.07$ in the subspace family and $0.93$ in the global family for seeds 0–2, against $8.76$, $7.74$, and $8.67$ (mean $8.39$) for the optimizer direction.

The main Qwen3.5 runs use learning rate $2 \times 10^{-5}$, cosine scheduling with 3% warmup, GRPO clip $0.2$, group size 4, eight rollout prompts per round, gradient accumulation 8, one optimizer epoch, and no KL penalty or weight decay. LoRA targets $q$, $v$, and fused $qkv$ projections with zero dropout and FP32 factors; geometry evaluation is FP32. Standard LoRA uses $\alpha = r$ ($\gamma_r = 1$); rsLoRA uses $\alpha = 8$ ($\gamma_r = 8/\sqrt{r}$). Rollouts use temperature $0.8$, and gradients are clipped at norm $1.0$. Training runs on two data-parallel GPUs with micro-batch 1, so each round takes two optimizer steps: 60 steps for 30 rounds and 400 for 200 rounds, with the cosine schedule and 3% warmup defined over all steps.

### C.2 Displacement and path controls

The displacement-matched study uses three independent training seeds and records effective updates at rounds 10, 20, and 30. The rank-64/rank-4 norm ratio remains within a factor of $1.018$ throughout. Endpoint training-evaluation on-policy KL ratios are $9.27$, $13.75$, and $10.67$; fixed-prefix ratios are $8.52$, $13.29$, and $9.94$. On held-out prompts, fixed-prefix ratios are $8.93$, $12.47$, and $10.68$

($10.70 \pm 1.77$); on-policy ratios are $8.88$, $10.70$, and $12.97$ ($10.85 \pm 2.05$). Path-coherence and cumulative-step-energy ratios are $0.970 \pm 0.032$ and $1.015 \pm 0.025$, respectively. Final-round training rewards are $0.896 \pm 0.154$ at rank 4 and $0.938 \pm 0.063$ at rank 64, measured on eight prompts with four sampled responses each. The paired seed-level reward ratios are $1.22$, $0.97$, and $1.00$. For the within-trajectory example, the standard LoRA rank-64 seed-1 effective norm increases from $0.21608$ to $0.26243$ between rounds 10 and 30, while FP32 fixed-prefix KL on the 32 calibration prompts of Appendix C.11 decreases from $1.5022$ to $0.4420$; on-policy KL averaged over six evaluation seeds falls from $1.413$ (95% CI $[1.193, 1.629]$) to $0.452$ ($[0.429, 0.475]$). On Qwen2.5-1.5B-Instruct (30 rounds), rsLoRA rank 4 uses $\alpha = 8$ and rank 64 uses $\alpha = 8.424$, set on seed 0 to match the rank-4 endpoint norm and kept fixed for seeds 1 and 2. The endpoint norm ratios are $1.009$, $1.036$, and $1.044$; FP32 fixed-prefix KL on shared held-out prefixes (64 prompts) gives rank-64/rank-4 ratios of $8.20$, $13.07$, and $17.27$ ($12.8 \pm 4.5$), and dividing by the squared norm ratio gives $8.05$, $12.2$, and $15.8$. A standard-LoRA pair of Qwen3.5-4B (seed 0, $\alpha = r$, 30 rounds) trained on MATH-500 problems of levels 2–4 instead of GSM8K moves rank 64 $3.98\times$ farther than rank 4 in effective-weight norm. Dividing the KL ratio by the squared norm ratio gives the rank-64/rank-4 ratio of $\bar{\mu}_K$: $10.6$ for on-policy and $11.2$ for fixed-prefix KL at round 30 ($13.5$ and $12.0$ at round 10), with a Fisher-sensitivity ratio of $20.8$.

### C.3 Finite-path decomposition

For a fixed evaluation measure $d$, write $K(\tau) = K_d(\theta_0 + \tau\Delta, \theta_0)$ along the update ray. Since $K(0) = K'(0) = 0$, Taylor's integral remainder gives

$$K(1) = \int_0^1 (1-\tau)K''(\tau)\, d\tau = \frac{1}{2}\,\|\Delta\|_{\mathrm{F}}^2\, \bar{\mu}_K(\Delta; d). \tag{52}$$

This identity is exact: $\bar{\mu}_K$ is a triangularly weighted average of curvature along the path, and finite KL is squared Euclidean magnitude times this curvature. In a standard Qwen3.5 rank sweep, the rank-64/rank-4 held-out fixed-prefix KL ratio factorizes as $144.76 \approx 16.26 \times 8.90$, the squared norm ratio times the curvature ratio. The displacement-matched rsLoRA runs remove the first factor and give $10.70 \pm 1.77\times$ ($8.93\times$ for seed 0).

Because finite KL weights curvature along the path, displacement and policy movement can move in opposite directions. Within a standard LoRA rank-64 trajectory (seed 1), the effective-weight norm increases by $21\%$ between rounds 10 and 30 while FP32 fixed-prefix KL decreases by $71\%$.

### C.4 Directional Fisher prediction

The Fisher tracking dataset contains eight Granite modules and ranks $4$, $16$, and $64$. Both directions' sequence scores are evaluated at the common reference model. The empirical sequence Fisher uses fixed sequences 1–8, while the target

$$G_{\mathrm{holdout}} = \frac{\mathrm{mean}_{9:16}\, K_{\mathrm{projected}}}{\mathrm{mean}_{9:16}\, K_{\mathrm{LoRA}}} \tag{53}$$

uses disjoint sequences 9–16. The primary predictor is

$$G_F = \frac{q_F(\Delta_P; d) / \|\Delta_P\|_{\mathrm{F}}^2}{q_F(\Delta_L; d) / \|\Delta_L\|_{\mathrm{F}}^2}. \tag{54}$$

The comparison predictors use $F_{d,T}$, the empirical Fisher of sequences 1–8 after projecting their scores onto the tangent space at the LoRA endpoint $\mathcal{T}_L$, with $\Delta_{\cdot,T} = P_{\mathcal{T}_L}\Delta_{\cdot}$. If $E_{0.9}$ contains all eigenmodes of $F_{d,T}$ with $\lambda_i \geq 0.9\lambda_{\max}$, principal alignment is

$$G_{\mathrm{top}} = \frac{\|P_{E_{0.9}}\Delta_{P,T}\|_{\mathrm{F}}^2 / \|\Delta_{P,T}\|_{\mathrm{F}}^2}{\|P_{E_{0.9}}\Delta_{L,T}\|_{\mathrm{F}}^2 / \|\Delta_{L,T}\|_{\mathrm{F}}^2}. \tag{55}$$

The spectrum-only summaries are stable rank $r_{\mathrm{stab}} = \mathrm{tr}(F_{d,T})/\lambda_{\max}$ and participation ratio $r_{\mathrm{PR}} = \mathrm{tr}(F_{d,T})^2 / \mathrm{tr}(F_{d,T}^2)$. Each predictor enters the same log-space OLS and grouped folds; no predictor is tuned within an evaluation fold. For every fold, ordinary least squares is refit as $\log G_{\mathrm{holdout}} =$

Table 4: **Fisher sketch-size results.** All $Q^2$ values use LOMO folds. The slope is from the all-row log-space fit.

| $k$ | $Q^2$ | Slope | No-fit score |
|---|---|---|---|
| 8 | 0.730 | 1.022 | 0.699 |
| 16 | 0.733 | 1.011 | 0.726 |
| 32 | 0.714 | 1.044 | 0.696 |
| 64 | 0.701 | 1.031 | 0.700 |

$a + b \log G_F$. LOMO holds out all three ranks of one module, and LORO holds out all eight modules of one rank. No positive row is clipped or removed.

On the identical 24 rows, OLS with an intercept and $\log r$ gives LOMO $Q^2 = 0.725$. Treating rank as a category, with each out-of-fold row predicted by the training-fold mean log target at that rank, gives $Q^2 = 0.719$. The sensitivity ratio $G_F$ gives $Q^2 = 0.819$, and rank combined with $G_F$ gives $0.931$. Principal alignment, stable rank, and participation ratio give $-0.182$, $0.153$, and $0.305$, and gradient alignment gives $0.740$. Within each rank, the descriptive Pearson correlations between $\log G_F$ and $\log G_{\text{holdout}}$ are $0.930$, $0.891$, and $0.802$ for ranks 4, 16, and 64 (eight modules each). For gradient alignment they are $0.37$, $0.27$, and $0.58$. Separate module-width fits use different row sets: for the predictor $\min(d_{\text{out}}, d_{\text{in}})/r$, LOMO $Q^2$ is $-1.086$ on six rank-4 Granite modules, $0.132$ across Granite ranks, $0.641$ across Qwen ranks, and $0.411$ on the joint dataset. On a second target, the KL ratio on sequences 65–80 of the 80-sequence batch, LOMO $Q^2$ is $0.763$ for $G_F$, $0.774$ for rank, $0.835$ for gradient alignment, $0.918$ for rank combined with $G_F$, and $0.952$ for gradient alignment combined with $G_F$. Within each rank, $G_F$ correlates with this target at $0.74$–$0.88$ and gradient alignment at $0.46$–$0.75$.

**Transfer across Fisher sketch sizes.** The 80-sequence evaluation batch supports nested Fisher sketches using the first $k \in \{8, 16, 32, 64\}$ sequences and a common KL target on sequences 65–80. For each of the 24 module–rank rows, let $P_{\mathcal{T}_L}$ project onto the tangent at the LoRA endpoint and set $\Delta_{L,T} = P_{\mathcal{T}_L}\Delta_L$, $\Delta_{P,T} = P_{\mathcal{T}_L}\Delta_P$, with sequence scores evaluated at the common reference model. The predictor is

$$G_{E,k} = \frac{\Delta_{P,T}^\top F_{d,k} \Delta_{P,T} / \left\|\Delta_{P,T}\right\|_{\text{F}}^2}{\Delta_{L,T}^\top F_{d,k} \Delta_{L,T} / \left\|\Delta_{L,T}\right\|_{\text{F}}^2}, \tag{56}$$

refit by log-space OLS in each LOMO fold. The no-fit score $1 - \sum_j (y_j - x_j)^2 / \sum_j (y_j - \bar{y})^2$, with $x_j = \log G_{E,k}$ and $y_j = \log G_{\text{holdout}}$, uses no fitted parameters. Prediction quality does not depend on the sketch size: from 8 to 64 sequences, LOMO $Q^2$ stays within $0.701$–$0.733$ and the slope near one (Table 4).

**Replication on Qwen3.5.** The same protocol runs on eight Qwen3.5 modules at ranks 4, 16, and 64 (24 rows): the four modules of the single-module interventions and four more chosen to cover depth and module type (layer-15 $q$ and $v$, the layer-21 fused QKV projection of a linear-attention block, and layer-31 $q$). $G_F$ uses the sensitivities stored on fixed sequences 1–8; the target is the fixed-prefix KL ratio on sequences 9–16, obtained by replaying the cumulative updates on the target modules (replayed and stored step-30 KL agree within $0.7\%$). Table 5 compares the models. The held-out KL ratio shrinks with rank in all eight modules, and the four added modules alone give the same result.

Table 5: **Directional Fisher sensitivity predicts held-out KL on two model families.**

| | Qwen3.5, 8 modules (24 rows) | Qwen3.5, 4 added (12 rows) | Granite (24 rows) |
|---|---|---|---|
| Spearman$(\log G_F, \log G_{\text{holdout}})$ | 0.981 | 0.965 | 0.922 |
| LOMO $Q^2$: $G_F$ / rank only | 0.939 / 0.684 | 0.899 / 0.552 | 0.819 / 0.725 |
| LOMO $Q^2$: rank and $G_F$ | 0.936 | 0.897 | 0.931 |
| Partial correlation given $\log r$ | 0.900 | 0.877 | 0.876 |

Table 6: **All optimizer-coordinate designs.** Each row reports the step-30 projected/LoRA fixed-prefix conditional KL gap; the final column is the range of cumulative projected/LoRA norm ratios across ranks.

| Model | Module | $G_K(4)$ | $G_K(16)$ | $G_K(64)$ | norm range |
|---|---|---|---|---|---|
| Qwen3.5 | layer 3 $v$ | 16.845 | 6.302 | 2.160 | 0.952–1.057 |
| Qwen3.5 | layer 3 $q$ | 93.688 | 20.939 | 3.465 | 0.936–1.014 |
| Qwen3.5 | layer 0 fused $qkv$ | 183.138 | 67.391 | 3.937 | 0.950–1.040 |
| Qwen3.5 | layer 31 $v$ | 52.913 | 19.403 | 6.783 | 1.003–1.026 |
| Qwen3.5 | layer 15 $q$ | 39.487 | 13.496 | 3.028 | 0.929–0.940 |
| Qwen3.5 | layer 15 $v$ | 49.654 | 8.540 | 1.329 | 0.974–0.991 |
| Qwen3.5 | layer 21 fused $qkv$ | 276.795 | 39.522 | 7.024 | 0.952–1.022 |
| Qwen3.5 | layer 31 $q$ | 111.789 | 24.910 | 12.080 | 0.935–0.963 |
| Granite | layer 20 $v$ | 40.686 | 3.716 | 0.858 | 0.963–0.984 |
| Granite | layer 20 $k$ | 137.624 | 36.143 | 6.169 | 0.961–0.996 |
| Granite | layer 20 $q$ | 44.965 | 5.910 | 1.048 | 0.952–0.968 |
| Granite | layer 20 $o$ | 13.576 | 5.521 | 4.577 | 0.900–0.937 |
| Granite | layer 20 up | 62.234 | 7.663 | 2.497 | 0.902–0.930 |
| Granite | layer 20 gate | 152.091 | 6.529 | 2.205 | 0.900–0.926 |
| Granite | layer 2 $v$ | 55.397 | 17.778 | 3.638 | 0.972–0.982 |
| Granite | layer 38 $v$ | 91.500 | 23.905 | 1.372 | 0.999–1.007 |

Table 7: **Absolute step-30 fixed-prefix KL for the optimizer-coordinate interventions.** Each cell gives LoRA / projected KL in units of $10^{-3}$, for the same trajectories as Table 6.

| Model | Module | Rank 4 | Rank 16 | Rank 64 |
|---|---|---|---|---|
| Qwen3.5 | layer 3 $v$ | 0.213 / 3.58 | 0.969 / 6.11 | 10.9 / 23.5 |
| Qwen3.5 | layer 3 $q$ | 0.120 / 11.2 | 1.06 / 22.3 | 16.2 / 56.1 |
| Qwen3.5 | layer 0 fused $qkv$ | 0.124 / 22.7 | 0.796 / 53.6 | 31.3 / 123 |
| Qwen3.5 | layer 31 $v$ | 0.00674 / 0.357 | 0.0634 / 1.23 | 0.782 / 5.31 |
| Qwen3.5 | layer 15 $q$ | 0.187 / 7.37 | 1.30 / 17.5 | 10.7 / 32.5 |
| Qwen3.5 | layer 15 $v$ | 0.188 / 9.36 | 3.17 / 27.1 | 36.9 / 49.1 |
| Qwen3.5 | layer 21 fused $qkv$ | 0.0334 / 9.24 | 0.801 / 31.6 | 11.3 / 79.5 |
| Qwen3.5 | layer 31 $q$ | 0.0167 / 1.86 | 0.433 / 10.8 | 2.69 / 32.5 |
| Granite | layer 20 $v$ | 0.563 / 22.9 | 11.8 / 43.7 | 69.9 / 59.9 |
| Granite | layer 20 $k$ | 0.0765 / 10.5 | 0.726 / 26.2 | 9.98 / 61.5 |
| Granite | layer 20 $q$ | 0.537 / 24.2 | 7.22 / 42.7 | 50.6 / 53.0 |
| Granite | layer 20 $o$ | 7.08 / 96.2 | 28.4 / 157 | 67.9 / 311 |
| Granite | layer 20 up | 0.969 / 60.3 | 13.3 / 102 | 58.2 / 145 |
| Granite | layer 20 gate | 0.399 / 60.6 | 15.6 / 102 | 61.8 / 136 |
| Granite | layer 2 $v$ | 0.0531 / 2.94 | 1.25 / 22.2 | 17.9 / 65.3 |
| Granite | layer 38 $v$ | 0.0246 / 2.25 | 0.260 / 6.23 | 7.02 / 9.63 |

### C.5 Optimizer-coordinate comparison

Both arms start from the same pre-step policy and use the same frozen 32-response training batch, old log probabilities, advantages, learning rate, and 16-sequence FP32 evaluation batch. Granite designs start from the base model, which also samples the batch and fixed prefixes. Qwen3.5 designs start from each rank's round-30 checkpoint, and all ranks share one batch and one prefix set sampled from the rank-16 round-30 policy. Both AdamW states are freshly initialized and then evolve independently. The LoRA arm updates factors. In the projected arm, let $V_t$ denote the raw full-weight AdamW proposal and let $\delta X_{L,t}$ be the LoRA effective step. The projected-arm step is exactly

$$\delta X_{P,t} = \|\delta X_{L,t}\|_{\mathrm{F}} \frac{\Pi_r(V_t)}{\|\Pi_r(V_t)\|_{\mathrm{F}}}, \qquad X_{P,t+1} = X_{P,t} + \delta X_{P,t}. \tag{57}$$

Thus the projected increment has rank at most $r$ and matches the LoRA increment's Frobenius norm. Both use the configured rank $r$. The LoRA increment $\gamma_r(B_{t+1}A_{t+1} - B_tA_t)$ can have rank up to $2r$ and is applied without an additional SVD truncation. Cumulative projected updates can exceed rank $r$. The primary gap is $G_K(r) = K_{\mathrm{projected}}/K_{\mathrm{LoRA}}$ after 30 optimizer steps.

### C.6 LAZY-$A$ REGIME AND PROBE SCALING

**Drift of $A$.** AdamW moves each entry by $\eta_t|\hat{m}_t|/(\sqrt{\hat{v}_t}+\epsilon)$ per step. This is typically about $\eta_t$, but a single step can reach $\eta_t(1-\beta_1)/\sqrt{1-\beta_2} \approx 3.2\,\eta_t$ for $\beta_1 = 0.9$ and $\beta_2 = 0.999$ (Kingma & Ba, 2015), so the following is a typical-scale estimate rather than a bound. With the default row scale $\|A_0\|_{\mathrm{F}}^2 \approx r/3$, the relative drift is of order $\sqrt{3d_{\mathrm{in}}}\sum_t \eta_t$. Rebuilding $A_0$ from the training seed gives the following measured drift $\|A_T - A_0\|_{\mathrm{F}} / \|A_0\|_{\mathrm{F}}$ and minimum row-space overlap: standard LoRA, 30 rounds, ranks 4 and 64: $0.83\%$ and $0.84\%$ (estimate $5.4\%$), overlap $\geq 0.9998$; Qwen2.5 ray source, 30 rounds: $0.68\%$ (estimate $4.2\%$); the 200-round pair, ranks 4 and 256: $5.6\%$ and $2.8\%$ (estimate $35\%$), overlap $\geq 0.992$. Every run we checked at the default learning rate therefore keeps $A$ at its random initialization.

**Coherent probes.** One backward pass on the fixed batch of each single-module intervention gives the start gradient $G$. The trained $B$ matches $-\operatorname{sign}(GA_0^\top)$ with cosine 0.94–0.99, 0.86–0.98, and 0.73–0.95 at ranks 4, 16, and 64 on Granite, and 0.92–0.94, 0.79–0.93, and 0.87–0.94 on Qwen3.5; every rank-one probe $b_k a_k^\top$ aligns positively with $G$. The coverage $\|GP_R\|_{\mathrm{F}}^2 / \|G\|_{\mathrm{F}}^2$ matches the random level $r/d_{\mathrm{in}}$ at every rank.

**Proof of Proposition 3.** For a module with row-side Fisher $I \otimes \Sigma$, the curvature of an update $\Delta \in \mathbb{R}^{m\times n}$ is $\operatorname{tr}(\Delta\Sigma\Delta^\top)$ and $\mu_F = \operatorname{tr}(\Delta\Sigma\Delta^\top)/\|\Delta\|_{\mathrm{F}}^2$. Let $P$ be the orthogonal projector onto the row span of the probes. In its first-order (SGD) form the lazy update is $\Delta \propto GP$, and $\mathbb{E}[G^\top G] = c\Sigma$ by assumption.

*Part 1.* $\mathbb{E}\operatorname{tr}(GP\Sigma PG^\top) = c\operatorname{tr}(P\Sigma P\Sigma)$ and $\mathbb{E}\|GP\|_{\mathrm{F}}^2 = c\operatorname{tr}(P\Sigma)$. For a uniformly random rank-$r$ projector, $\mathbb{E}[P] = (r/n)I$ and $\mathbb{E}[P\Sigma P] = \alpha\Sigma + \beta\operatorname{tr}(\Sigma)I$ with

$$\beta = \frac{r(n-r)}{n(n-1)(n+2)}, \qquad \alpha = \frac{r}{n} - \frac{r(n-r)}{(n-1)(n+2)}, \tag{58}$$

which follow from $P^2 = P$ and $(\operatorname{tr}P)^2 = r^2$. The ratio of expectations is $(\alpha\operatorname{tr}\Sigma^2 + \beta(\operatorname{tr}\Sigma)^2)/((r/n)\operatorname{tr}\Sigma)$; substituting $\alpha$ and $\beta$ gives exactly

$$\frac{\mathbb{E}\operatorname{tr}(P\Sigma P\Sigma)}{\bar{\lambda}\,\mathbb{E}\operatorname{tr}(P\Sigma)} = \frac{n\,(n(r+1)-2)}{(n-1)(n+2)}\,\frac{1}{p} + \frac{n(n-r)}{(n-1)(n+2)}, \tag{59}$$

which equals $1 + (r+1)/p$ up to relative corrections of order $r/n$. The term $r+1$ rather than $r$ arises because $\alpha = r(r+1)/n^2 + O(r^2/n^3)$: the fluctuation of the captured energy across draws of $P$ contributes at the same order as $\beta$. A Monte Carlo check reproduces the exact expression to within $0.5\%$ for $n \in \{60, 200, 1000\}$. Single draws of $P$ scatter around the expectation. Under sign-normalized updates the per-probe energies equalize; the rank dependence then enters through cross-probe terms whose signs follow $a_k^\top \Sigma a_l$, and the leading-order form is an approximation.

*Part 2.* If the probes are orthonormal eigenvectors $e_1, \ldots, e_r$ of $\Sigma$, then $\Delta = \sum_{i\leq r} b_i e_i^\top$, $\operatorname{tr}(\Delta\Sigma\Delta^\top) = \sum_i \|b_i\|_{\mathrm{F}}^2 \lambda_i$ because $e_i^\top \Sigma e_j = \lambda_i \delta_{ij}$, and $\|\Delta\|_{\mathrm{F}}^2 = \sum_i \|b_i\|_{\mathrm{F}}^2$. Hence $\mu_F(r) = \sum_{i\leq r} w_i\lambda_i / \sum_{i\leq r} w_i$ with $w_i = \|b_i\|_{\mathrm{F}}^2$. Under SGD, $b_i = Ge_i$ and $\mathbb{E}\,w_i = c\lambda_i$; under sign normalization every entry of $b_i$ moves by about $\eta_t$ per step and the $w_i$ are equal. Writing $N_r = \sum_{i\leq r} w_i\lambda_i$ and $D_r = \sum_{i\leq r} w_i$, $\mu_F(r+1) \leq \mu_F(r)$ is equivalent to $\lambda_{r+1} \leq N_r/D_r$, which holds because $N_r/D_r$ is a weighted mean of values at least $\lambda_r \geq \lambda_{r+1}$. The same weighted-mean property gives $\mu_F(r) \geq \lambda_r$.

*Part 3.* With $P_0$ the projector onto the row space of $A_0$ and $A = A_0 + \delta A$, the energy outside that row space is $\|A(I-P_0)\|_{\mathrm{F}}^2 = \|\delta A(I-P_0)\|_{\mathrm{F}}^2 \leq \delta^2\|A_0\|_{\mathrm{F}}^2$, and $\|A\|_{\mathrm{F}} \geq (1-\delta)\|A_0\|_{\mathrm{F}}$.

**Gradient consistency in online training.** Base-model gradients on eight independent training batches have mean pairwise cosine $0.003$ (range $-0.033$ to $0.059$). An AdamW simulation with $\beta_1 = 0.9$, $\beta_2 = 0.999$, the cosine schedule, and measured gradient scales reproduces the persistence of $B$ between rounds 10 and 30 (norm ratio 1.20, cosine 0.93, increment cosine 0.27; measured 1.20–1.24, 0.92–0.94, 0.27–0.33) with pure noise and rejects signal fractions of 0.01 and above. At this resolution the batch gradients of the online runs behave like noise.

### C.7 PROBE-PROJECTION PREDICTIONS

We test Proposition 3 with the actual factors. For each design we form the predicted update from the actual $A_0$ and gradient, $\Delta_{\mathrm{SGD}} = GA_0^\top A_0$ and $\Delta_{\mathrm{Adam}} = \mathrm{sign}(GA_0^\top)A_0$, and evaluate its Fisher sensitivity with the same $F_d$ as the measured update. On a fixed batch the gradient is a single matrix; online we use $\sum_j \mathrm{sign}(G_j A_0^\top)A_0$ over eight independent batch gradients, because with nearly uncorrelated batch gradients the trained $B$ accumulates signs of individual batch projections. Sensitivities of the online runs use the base-model $F_d$ on 8 and 48 fixed sequences for both predicted and measured updates. The row-side and column-side sensitivities use the low-rank form $\Delta = LR$, so that $\mathrm{tr}(\Delta\Sigma\Delta^\top) = \mathrm{tr}((L^\top L)(R\Sigma R^\top))$ requires only $r \times r$ products of per-sequence scores.

Table 8: **Predicted and measured rank-64/rank-4 Fisher-sensitivity ratios on Granite fixed batches** (modules $v$ / $q$ / up).

| Probes | Measured | SGD prediction | Adam prediction | $\cos(\Delta_{\mathrm{Adam}}, \Delta X)$ |
|---|---|---|---|---|
| Frozen $A$ | 27.3 / 7.8 / 9.3 | 23.8 / 8.4 / 16.6 | 41.6 / 7.2 / 16.7 | 0.86–0.997 |
| Default | 15.5 / 6.7 / 3.05 | 23.8 / 8.4 / 16.6 | 41.6 / 7.2 / 16.7 | 0.83–0.98 |
| Aligned $A_0$ | 0.039 / 0.092 / 0.033 | 0.98 / 0.96 / 0.84 | 0.105 / 0.129 / 0.231 | 0.69–0.86 |

On Granite the Adam form predicts the direction in all nine cells and the magnitude within a factor of two for frozen $A$ in all three modules; the SGD form stays near one for aligned probes. The row-side participation ratios are $1.92$, $1.86$, and $7.88$, giving random-probe predictions of $9.7$, $9.7$, and $5.7$; aligned curves are nonincreasing under both weightings; and aligned probes raise rank-4 sensitivity $500$–$600\times$ over random ones. The gradient's row-side second moment overlaps the top four eigenvectors of $\Sigma$ at $0.51$–$0.85$ (matrix cosine $0.86$–$0.98$), from a single fixed batch.

Table 9: **Predicted and measured online rank-64/rank-4 ratios on Qwen3.5** (seeds 0 / 1 / 2; base-model $F_d$; the aligned prediction is shared by the seeds).

| Runs | Fisher | Measured | Predicted |
|---|---|---|---|
| Random $A_0$ | $F_d$, 8 sequences | 4.35 / 16.5 / 8.8 | 16.0 / 14.4 / 11.4 |
| Random $A_0$ | $F_d$, 48 sequences | 8.9 / 14.0 / 10.0 | 14.4 / 12.7 / 15.2 |
| Aligned $A_0$ | $F_d$, 8 sequences | 1.40 / 0.46 / 1.11 | 0.39 |
| Aligned $A_0$ | $F_d$, 48 sequences | 1.01 / 0.81 / 0.78 | 0.71 |
| Aligned $A_0$ | $F_d$, 128 sequences | 0.80 / 0.80 / 0.70 | – |
| Random $A_0$ | row side only | 7.5 / 7.0 / 8.2 | 7.1 / 6.0 / 7.6 |
| Aligned $A_0$ | row side only | 0.089 / 0.090 / 0.090 | 0.079 |
| Aligned $A_0$ | two-sided K-FAC, 128 sequences | 0.084 / 0.097 / 0.092 | – |

On Qwen3.5 the random-probe predictions match within a factor of two for all three seeds under the 48-sequence Fisher. On the row side, predictions match the measured ratios within $1.2\times$ for random and aligned probes alike. Under the full sequence Fisher the aligned reversal is much weaker ($0.70$–$0.80$ with 128 sequences) than on the row side ($0.08$). A two-sided Kronecker (K-FAC) approximation $\Sigma_{\mathrm{out}} \otimes \Sigma_{\mathrm{in}}$ gives the same $0.08$–$0.10$ as the row side alone, so the difference comes from the coupling of input and output sides within each sequence, which no Kronecker factorization represents. Row-side and K-FAC ratios are the same with 8, 48, and 128 sequences, whereas the full $F_d$ is noisy with 8 sequences and stable from 48 on: the gap between them is structural. On the lower-learning-rate runs of Table 12, the 128-sequence base-model $F_d$ matches the mean on-policy KL ratio of both initializations and each seed within a factor of two. The median row-side participation ratio is $2.8$ ($1.4$–$13.6$), the median random-probe prediction $8.7$, aligned curves are nonincreasing in 40 of 40 modules, and aligned probes raise rank-4 sensitivity by a median of $527\times$. The gradient's row-side second moment has median matrix cosine $0.96$ with $\Sigma$ and top-four overlap $0.66$. The cosine between predicted and measured online updates is $-0.02$ to $0.04$: the predicted rank ratio holds while the realized direction, which depends on the particular noise realization, does not match.

### C.8 INTERVENTIONS ON THE PROBE GEOMETRY

All cells use the single-module protocol of Appendix C.5 on Granite layer 20. The default and aligned designs are repeated with seeds 1 and 2, each drawing a new $A_0$ and a new fixed batch

with shared held-out prefixes; across the nine module–seed pairs, $G_K(4)$ spans 13.8–134.5 with random and 0.24–1.36 with aligned $A_0$, and the LoRA-arm sensitivity ratio $\mu_L(64)/\mu_L(4)$, with $\mu_L(r) = \mu_F(\Delta_L; d)$ at rank $r$, spans 0.64–19.3 and 0.033–0.344. The default exceptions to the rank ordering are the up projection in seeds 1 and 2 ($G_K$ from rank 16 to 64: $8.31 \to 8.63$ and $5.54 \to 7.57$). The other designs use one seed. Frozen $A$ keeps $A$ at initialization; gradient-aligned $A_0$ holds the top-$r$ right singular vectors of $G$ with the default row norms and $B_0 = 0$; the $A$-learning-rate intervention multiplies the learning rate of $A$ by 20; PiSSA initializes both factors from the leading singular triplets of the weight (Meng et al., 2024); SGD replaces AdamW in both arms. The $A$-learning-rate and PiSSA runs rescale the learning rate toward the default displacement and reach 0.50–0.90 and 1.01–1.20 of it.

Table 10: **All probe-geometry interventions.** Step-30 projected/LoRA fixed-prefix KL ratio $G_K(r)$, LoRA displacement relative to the default across ranks, and the LoRA-arm sensitivity ratio $\mu_L(64)/\mu_L(4)$.

| Intervention | Module | $G_K(4)$ | $G_K(16)$ | $G_K(64)$ | Displacement | $\mu_L(64)/\mu_L(4)$ |
|---|---|---|---|---|---|---|
| Default | $v$ | 40.7 | 3.72 | 0.86 | 1.00–1.00 | 15.5 |
| | $q$ | 45.0 | 5.91 | 1.05 | 1.00–1.00 | 6.69 |
| | up | 62.2 | 7.66 | 2.50 | 1.00–1.00 | 3.05 |
| Frozen $A$ | $v$ | 82.0 | 6.30 | 1.24 | 0.98–1.00 | 27.3 |
| | $q$ | 91.0 | 11.5 | 1.90 | 0.97–1.00 | 7.84 |
| | up | 307.1 | 33.7 | 5.02 | 0.99–1.01 | 9.32 |
| Gradient-aligned $A_0$ | $v$ | 0.24 | 0.54 | 0.90 | 0.80–0.90 | 0.04 |
| | $q$ | 0.36 | 0.65 | 1.06 | 0.83–0.88 | 0.09 |
| | up | 1.14 | 1.52 | 2.09 | 0.77–0.80 | 0.03 |
| $A$ lr $\times 20$ | $v$ | 3.27 | 1.66 | 0.90 | 0.50–0.84 | 2.36 |
| | $q$ | 4.52 | 2.54 | 0.91 | 0.54–0.82 | 1.61 |
| | up | 1.31 | 0.87 | 1.15 | 0.64–0.90 | 0.21 |
| PiSSA | $v$ | 49.4 | 6.16 | 1.27 | 1.01–1.17 | 9.73 |
| | $q$ | 117.7 | 4.84 | 0.88 | 1.07–1.20 | 29.9 |
| | up | 33.0 | 4.96 | 3.60 | 1.12–1.17 | 1.00 |
| SGD | $v$ | 56.4 | 8.47 | 5.21 | 0.83–1.13 | 4.52 |
| | $q$ | 8.15 | 3.51 | 2.65 | 0.86–1.54 | 0.66 |
| | up | 8.49 | 3.95 | 3.39 | 0.88–1.36 | 0.40 |
| PiSSA + SGD | $v$ | 2.99 | 1.18 | 1.39 | 0.65–0.99 | 1.79 |
| | $q$ | 73.3 | 5.31 | 2.41 | 0.76–1.07 | 22.6 |
| | up | 15.2 | 4.02 | 2.93 | 0.84–1.02 | 0.77 |

**Gradient alignment.** At rank 4, the leading $r$ right singular vectors of the start gradient carry 54–94% of its energy across the three modules. Across the 54 designs of Table 10, including the first-pass runs at unmatched displacement, $\log(\text{align}_P/\text{align}_L)$ correlates with $\log G_K$ at 0.92, and the log initial probe coverage of $G$ at $-0.63$.

With SGD in both arms, the LoRA-arm sensitivity ratio falls below one on $q$ and up (0.66 and 0.40), contrary to part 1 of Proposition 3.

**PiSSA channels.** PiSSA's update lies exactly in the tangent space of the rank-$r$ weight approximation (normal share 0.000) and splits into a row-side channel $\delta B\, A_0$ and a column-side channel $B_0\, \delta A$. Under AdamW each factor entry moves by about $\eta$, so the column-side share of update energy equals the parameter-count ratio $d_{\text{in}}/(d_{\text{in}} + d_{\text{out}})$: measured 0.77–0.84, 0.48–0.50, and 0.21–0.23 for $v$, $q$, and up against 0.83, 0.50, and 0.24. The first-step sign operator $\text{sign}(GA_0^\top)A_0 + B_0\,\text{sign}(B_0^\top G)$ reproduces PiSSA's measured gradient alignment within 15% at ranks 4 and 16, including an alignment below the default on $v$. The leading 64 singular values of the Granite weights are nearly flat, so singular-value weighting does not slow the growth of the effective probe count ($r_{\text{eff}}(64)/r_{\text{eff}}(4) = 15.7$–$15.9$). With SGD, channels are weighted by gradient energy and PiSSA's gradient alignment rises to 2.3–9.5$\times$ the default; $G_K(4)$ falls on $v$ and up, and falls less on $q$, where the projected arm's alignment also rises about tenfold.

### C.9 GRADIENT-ALIGNED ONLINE RUNS

The runs repeat the displacement-matched rsLoRA protocol (Qwen3.5-4B, $\alpha = 8$, 30 rounds, seeds 0–2) with $A_0$ set to the top-$r$ right singular vectors of the base-model gradient averaged over eight training batches, row norms matched to the default initialization, and $B_0 = 0$. The probe account predicts a mean rank-64/rank-4 $\mu_F$ ratio of at most 2 (below 4 in every seed), a mean on-policy KL ratio of at most 3, and a higher rank-4 $\mu_F$ than with random $A_0$ in every seed.

Table 11: **Gradient-aligned $A_0$ in online RLVR.** Round-30 rank-64/rank-4 ratios; random-$A_0$ values are the runs of Section 3.

| Seed | $\mu_F$, random $A_0$ | $\mu_F$, aligned $A_0$ | $\mathrm{KL}_{\mathrm{on}}$, random | $\mathrm{KL}_{\mathrm{on}}$, aligned | Norm, aligned | Rank-4 $\mu_F$ gain |
|---|---|---|---|---|---|---|
| 0 | 13.34 | 0.457 | 9.27 | 1.10 | 1.047 | 721× |
| 1 | 19.53 | 0.614 | 13.75 | 1.67 | 1.014 | 780× |
| 2 | 13.70 | 0.221 | 10.67 | 0.84 | 1.021 | 2114× |

All three hold (Table 11). Gradient clipping at norm 1 was active for rank 4 in 14–19 of 30 rounds (mean coefficient 0.77–0.86) and never for rank 64; rerunning rank 4 without clipping gives $\mu_F$ ratios 0.40, 0.67, and 0.17 (mean 0.41) at displacement ratios 0.99–1.01. On-policy KL of 6–13 per sequence compresses the KL ratios toward 1. For batch gradients $G_1, \ldots, G_8$ with mean $\bar{G}$, $\bar{G}^\top \bar{G} = \frac{1}{64} \sum_i G_i^\top G_i + \frac{1}{64} \sum_{i \neq j} G_i^\top G_j$. The cross terms have expectation $M^\top M$ for the population mean gradient $M$, which is small at pairwise cosine 0.003, so the right singular vectors of $\bar{G}$ estimate, from eight samples, the leading eigenvectors of the row-side second moment $\mathbb{E}[G^\top G]$, which equals the row-side covariance up to the small mean term. The aligned $A_0$ therefore targets directions of large gradient variance rather than the mean gradient.

**Unsaturated KL.** A second set of aligned runs lowers the learning rate to $2 \times 10^{-6}$, the value at which a probe run kept rank-4 on-policy KL below one nat per sequence, and repeats both arms at ranks 4 and 64 for seeds 0–2 (Table 12). The random arm keeps the gap of Section 3. With aligned $A_0$ the rank-64/rank-4 KL ratio is below one on average, below one at rounds 10, 20, and 30 in seeds 1 and 2, and equal to one in seed 0.

Table 12: **Aligned probes at unsaturated KL** (learning rate $2 \times 10^{-6}$). Round-30 rank-64/rank-4 ratios for seeds 0, 1, 2. Fisher sensitivity is evaluated at the trained policies and with the base-model sequence Fisher on 128 sequences; rank-4 on-policy KL is in nats per sequence.

| $A_0$ | $\mathrm{KL}_{\mathrm{on}}$ | $\mathrm{KL}_{\mathrm{prefix}}$ | $\mu_F$, trained | $\mu_F$, base $F_d$ | Norm | Rank-4 $\mathrm{KL}_{\mathrm{on}}$ |
|---|---|---|---|---|---|---|
| Random | 8.35 / 10.08 / 15.37 | 8.55 / 10.25 / 14.19 | 7.0 / 14.6 / 18.2 | 8.57 / 11.36 / 17.27 | 1.01–1.03 | ≈0.001 |
| Aligned | 1.00 / 0.32 / 0.43 | 0.99 / 0.41 / 0.43 | 1.13 / 0.80 / 0.19 | 0.68 / 0.64 / 0.42 | 0.98–1.01 | 0.15–0.39 |

**$A$ learning rate ×20.** A second set of runs keeps random $A_0$ and multiplies the learning rate of every $A$ factor by 20, otherwise matching the runs of Section 3. On fixed batches this learning rate turns the probes toward the gradient and attenuates the rank effect (Table 1). Online the rank effect persists in every seed, with $\mu_F$ ratios of 3.8–202 (Table 13); the change from the default comes from rank 64, whose $\mu_F$ rises from 82 to 1151 in seed 2, while rank-4 $\mu_F$ stays near its random-$A_0$ value.

Table 13: **$A$ learning rate ×20 in online RLVR.** Round-30 rank-64/rank-4 ratios; no round is clipped.

| Seed | $\mu_F$, default | $\mu_F$, $A$ lr ×20 | $\mathrm{KL}_{\mathrm{on}}$, $A$ lr ×20 | Norm | Rank-4 $\mathrm{KL}_{\mathrm{on}}$ |
|---|---|---|---|---|---|
| 0 | 13.3 | 24.6 | 15.0 | 1.17 | 0.097 |
| 1 | 19.5 | 3.8 | 2.3 | 1.08 | 0.167 |
| 2 | 13.7 | 201.9 | 40.3 | 1.13 | 0.113 |

$A$ drifts by 15–18% of $\|A_0\|_{\mathrm{F}}$ (about 17× the default) and its row space keeps overlap 0.970–0.982 with $A_0$. The ratio $\|A_{30} - A_0\|_{\mathrm{F}} / \|A_{10} - A_0\|_{\mathrm{F}}$ is 1.27–1.38 (1.34–1.50 at the default learning rate). An AdamW simulation with $\beta_1 = 0.9$, $\beta_2 = 0.999$, the cosine schedule, and an $A$-gradient

scale that grows with the simulated $\|B_t\|_{\mathrm{F}}$ (whose ratio, 1.20, matches the measured 1.20–1.23) gives 1.28 for pure noise, 1.37 for gradient coherence 0.01, and 1.84 for a fixed direction, placing the runs at coherence below about 0.01. By part 3 of Proposition 3, a relative drift of 15–18% moves at most 3–5% of the probe energy out of the row space of $A_0$. The channel $\gamma_r B_T(A_T - A_0)$ carries 3.8–5.0% of the update energy at rank 4 and 20–25.5% at rank 64 (near zero at the default learning rate). With the default learning rate on the fixed batches of Appendix C.8, $\|A_{30} - A_0\|_{\mathrm{F}} / \|A_{10} - A_0\|_{\mathrm{F}}$ has median 3.84 (3.17–3.99) and successive increments have cosine 0.985 (3.53 and 0.854 with aligned $A_0$), against 1.27–1.38 and 0.16–0.32 for the online runs above and 1.34–1.50 and 0.19–0.46 at the default online learning rate; ratios above the constant-rate value of 3 reflect the growth of the $A$ gradient with $\|B\|_{\mathrm{F}}$. On the same fixed batches, a 20× learning rate moves every row of $A$ along its own Adam direction (95–100% of rows with positive cosine) and 39–92% of the update energy out of the initial row space (0.3–5% at the default learning rate).

### C.10 Explicit KL penalty

These runs repeat the displacement-matched rsLoRA protocol of Section 3 with random $A_0$ and a KL penalty of coefficient 0.04 at ranks 4 and 64, seeds 0–2 (Table 14). Relative to the same seeds without penalty, the effective-weight norm changes by −3.5% to +0.4% at both ranks, while rank-4 on-policy KL falls from 0.05–0.11 to 0.04 and rank-64 KL from 0.53–1.12 to 0.20–0.46 nats per sequence (29–65%); the rank-64/rank-4 KL ratio falls from 11.2 to 9.3 on average.

Table 14: **Rank effect under an explicit KL penalty** (coefficient 0.04). Round-30 rank-64/rank-4 ratios; $\bar{\mu}_K$ ratio is the KL ratio divided by the squared norm ratio; the last column is the run without penalty.

| Seed | Norm | $\mathrm{KL}_{\mathrm{on}}$ | $\bar{\mu}_K$ | $\mu_F$ | $\mathrm{KL}_{\mathrm{on}}$, no penalty |
|---|---|---|---|---|---|
| 0 | 0.990 | 5.42 | 5.53 | 9.65 | 9.27 |
| 1 | 0.997 | 12.55 | 12.63 | 7.61 | 13.75 |
| 2 | 0.991 | 9.79 | 9.97 | 9.66 | 10.67 |
| Mean | 0.993 | 9.25 | 9.38 | 8.97 | 11.23 |

### C.11 Orientation selection and independent confirmation

Starting from the standard LoRA Qwen3.5 rank-64 round-30 factors, we construct the identity ($Q = I$) and 15 seeded orthogonal mixes $B \leftarrow BQ$ and normalize every effective update to the rank-4 endpoint norm. On 32 fixed GSM8K training prompts used for calibration, the identity has the largest and one of the mixes the smallest $q_F$, and these two are selected. No finite-KL or downstream test result is used for this selection. The pair is then frozen and evaluated without reselection on 64 fixed GSM8K test prompts drawn with an independent shuffle seed. The test-prompt high/low fixed-prefix KL ratio is 238.73× (247.82× on-policy), and the 64-sequence directional Fisher ratio is 268.55× with paired state-bootstrap interval $[180.05, 419.91]$. On the full held-out FP32 evaluation, the KL ratio is 239.77× on fixed prefixes and 244.36× on-policy; on a separate shared 64-sequence batch, the normalized endpoint sequence-Fisher sensitivity ratio is 274.2×. Each adapter's score is evaluated at its own endpoint. The selected high direction is the normalized optimizer update, while the 15 nonidentity candidates span a 3.36× range in calibration $q_F$. This pair turns optimizer-direction selection within a fixed subspace into a quantitative prediction lower bound.

### C.12 Continuous utility contrasts

The general-text sequences are 512 passages from the validation split of WikiText-103 (Merity et al., 2017) (`wikitext-103-raw-v1`). Candidates are the lines whose stripped length is 256–1800 characters, which excludes empty lines and section headings; a seeded shuffle selects 512 of them, and each is truncated to 512 tokens under the Qwen3.5 tokenizer as plain text, without a chat template. The selected line indices and a hash of the concatenated text are released with the code. The four contrasts use these 512 sequences for FP32 NLL and the same 1000 MMLU items across conditions. We report perplexity (PPL) changes alongside NLL. Negative ΔNLL is improvement; positive Δ margin means the correct option separates further from the best incorrect option.

Table 15: **Continuous OOD utility changes sign with update semantics.** Intervals are paired 95% bootstrap intervals; NLL values and intervals are scaled by $10^4$. ΔNLL averages per-sequence differences; ΔPPL uses token-pooled NLL. Length-reward runs last 20 rounds; their endpoints are the final round. The seed-1 row uses rank 64 and the same texts, items, and bootstrap seeds.

| Contrast | ΔNLL | NLL CI | ΔPPL (%) | Δmargin | Item CI |
|---|---|---|---|---|---|
| Orientation low→high | $-2.190$ | $[-2.508, -1.879]$ | $-0.0238$ | $+0.0004$ | $[-0.0005, +0.0012]$ |
| Standard $r = 4 \to r = 64$ | $-7.877$ | $[-9.041, -6.721]$ | $-0.0867$ | $+0.0003$ | $[-0.0033, +0.0037]$ |
| rsLoRA $r = 4 \to r = 64$ | $-3.081$ | $[-4.028, -2.168]$ | $-0.0341$ | $+0.0155$ | $[+0.0120, +0.0189]$ |
| Length reward $\beta = 0 \to 2$ | $+8.686$ | $[+7.037, +10.335]$ | $+0.0991$ | $-0.0921$ | $[-0.1057, -0.0780]$ |
| Length reward, seed 1 | $+16.69$ | $[+15.1, +18.3]$ | – | $-0.0735$ | $[-0.087, -0.060]$ |

Relative to the base model, $\beta = 0$ lowers NLL by $7.97 \times 10^{-4}$ and $12.0 \times 10^{-4}$ in seeds 0 and 1, while $\beta = 2$ changes it by $+0.7 \times 10^{-4}$ ($[-0.7, +2.1] \times 10^{-4}$) and $+4.7 \times 10^{-4}$. Subject-level MMLU intervals are $[+0.0113, +0.0200]$ for rsLoRA and $[-0.1061, -0.0776]$ for the length-reward contrast; the direction and standard-rank intervals include zero. On shared sequences, the length-reward endpoints have Fisher sensitivities $\mu_F = 20.559$ at $\beta = 0$ and $890.956$ at $\beta = 2$, a $43.34\times$ ratio.

### C.13 Generation accounting

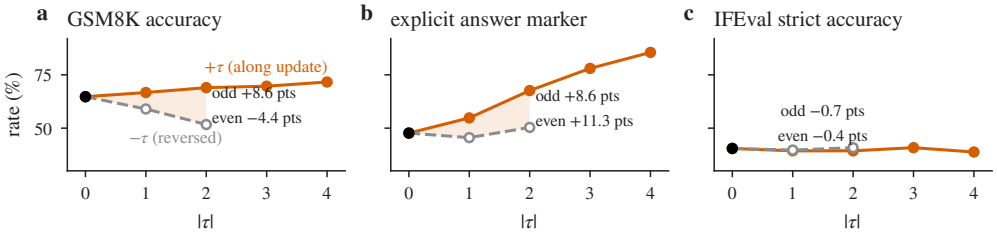



Figure 4: **GSM8K utility changes sign along an update ray.** Qwen2.5-1.5B-Instruct metrics at $\theta_0 + \tau\Delta$ for $\tau > 0$ (vermillion), $\tau < 0$ (grey), and the base model at $\tau = 0$ (black). Labels at $|\tau| = 2$ give the odd part $[f(+\tau) - f(-\tau)]/2$ and even part $[f(+\tau) + f(-\tau)]/2 - f(0)$. GSM8K accuracy is dominated by its odd part; truncation stays at or below 0.08%.

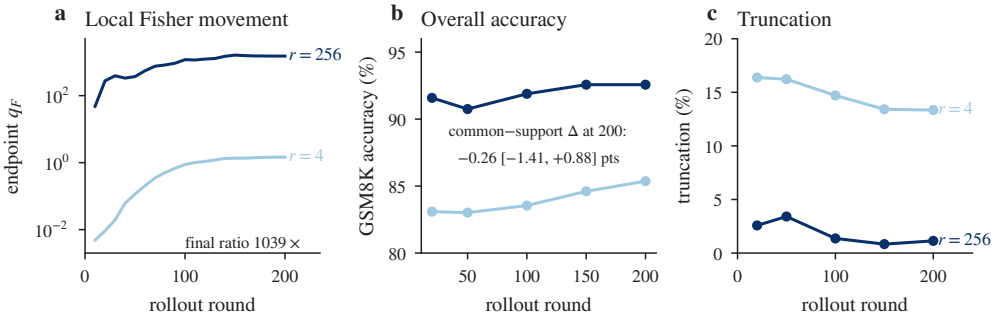



Figure 5: **Large local policy movement without conditional loss.** Matched 200-round rank-256 and rank-4 runs. (a) Endpoint $q_F$ separates by three orders of magnitude. (b) Overall GSM8K accuracy is higher for rank 256; at round 200, common-support conditional accuracy differs by $-0.26$ points. (c) Truncation falls for rank 256.

For a generation condition $j$ with truncation probability $t_j$ and accuracies $c_j$ and $c'_j$ on nontruncated and truncated outputs, $\mathrm{Acc}_j = (1 - t_j)c_j + t_j c'_j$, so with midpoint values

$$\Delta\,\mathrm{Acc} = (1 - \bar{t})\Delta c + \bar{t}\Delta c' + (\bar{c}' - \bar{c})\Delta t. \tag{60}$$

The first term is the conditional channel and the last the completion channel.

**Budgets and denominators.** The main generation evaluations cover 22 conditions with 512-token GSM8K and 1024-token MATH-500 budgets. A reevaluation at 1024 and 2048 tokens gives the following paired contrasts. At the longer budgets, overall GSM8K accuracy changes by $[-0.76, +0.99]$ percentage points relative to base, and MATH-500 by $[-1.0, +2.6]$ points. Each condition's nontruncated accuracy uses its own completion set; common-support conditional accuracy uses the intersection of the two completion sets.

Along the Qwen2.5-1.5B-Instruct ray, GSM8K accuracy changes by $+4.25$ points at $\tau = +2$ and $-13.04$ points at $\tau = -2$ relative to the base model. For the anti-ray ($\tau = -2$ minus $\tau = +1$), overall GSM8K accuracy changes by $-15.01$ percentage points on 1319 prompts. On the 1318 common-support prompts, 86 change from incorrect to correct and 284 from correct to incorrect, giving $100(86 - 284)/1318 = -15.02$ points, with paired 95% CI $[-17.75, -12.29]$. Truncation decreases from $1/1319$ to zero.

For the 200-round rank-256 minus rank-4 contrast, endpoint $q_F$ is $1039\times$ and held-out on-policy KL $74.9\times$ higher, and GSM8K truncation is $1.1\%$ against $13.3\%$. Overall GSM8K accuracy increases by $7.20$ points. On the 1138 common-support prompts, 20 improve and 23 worsen, giving $100(20 - 23)/1138 = -0.26$ points, with paired 95% CI $[-1.41, +0.88]$.