ISO: AN RLVR-NATIVE OPTIMIZATION STACK

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning with verifiable rewards (RLVR) advances the reasoning capabilities of language models, yet the optimization layer that converts reward feedback into weight-space updates remains poorly understood. We study this layer and functionally validate *spectral inheritance*: RLVR can reuse the base model's weight spectra while acquiring new behavior by adapting both the associated input and output singular frames. We operationalize this finding as **Isospectral Optimization (ISO)**, an RLVR-native optimization stack with complementary offline and online instantiations. *Offline*, **ISO-Merger** composes shared-base specialists in fixed-spectrum frame coordinates without post-merge data, rollouts, or distillation, and achieves the highest aggregate scores among the data-free merging methods. *Online*, **ISO-Optimizer** applies AdamW or Muon directly to the frame variables under fixed base spectra, improving aggregate accuracy in the reported math and coding experiments. On Qwen3-4B-Base, ISO-AdamW reaches AdamW's 400-update accuracy in 260 updates with 1.5 $\times$ fewer updates and 34% less wall-clock time, and exceeds AdamW by 2.2 points at update 400. These results support a reusable RLVR design principle: *inherit the spectrum, optimize the frames*.

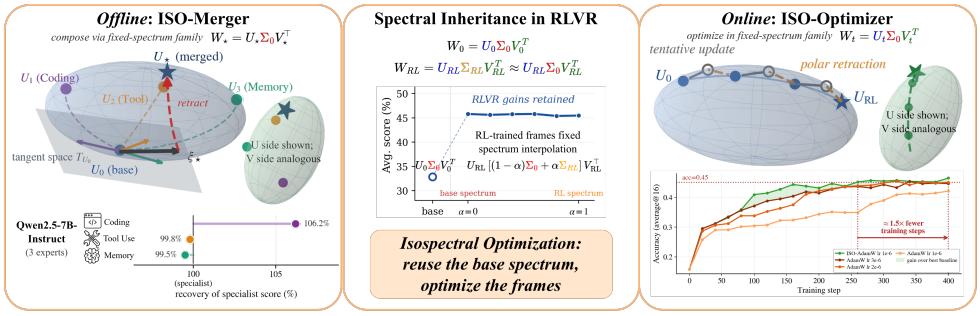


Figure 1: **From spectral inheritance to Isospectral Optimization. Middle.** Unconstrained RLVR changes both singular frames while its spectra remain close to their base values: restoring the base spectra while retaining RL-trained frames preserves most acquired gains. ISO turns this regularity into an inductive bias: reuse Σ_0 and optimize (U, V). **Left.** ISO-Merger (offline) applies project–mask–merge–retract to shared-base specialists' frame changes and reconstructs a fixed-spectrum model that recovers their specialist capabilities. **Right.** ISO-Optimizer (online) applies a conventional optimizer (e.g., AdamW) to (U, V) under fixed Σ_0; on Qwen3-4B-Base, ISO-AdamW reaches AdamW's end-of-run accuracy with $1.5\times$ fewer actor updates.

1 INTRODUCTION

Reinforcement learning with verifiable rewards (RLVR) has become a major scaling axis for reasoning models (xAI, 2025), with advances in data (Scale AI, 2026), objectives (Guo et al., 2025; Tajwar et al., 2026), and systems (Zhu et al., 2025c). Yet the optimizers and parameterizations that translate reward feedback into weight-space updates remain largely inherited from pre-training (Loshchilov & Hutter, 2019; Liu et al., 2025a; Zhu et al., 2025a; Wen et al., 2025). This default is not obviously natural: pre-training learns from dense token-level supervision, whereas RLVR adapts an already capable policy using comparatively sparse outcome-level rewards.

We study this missing optimization layer and uncover a separation between *what RLVR reuses and what it changes*. We call this **spectral inheritance**: RLVR can *reuse the base model's weight spectra* while *acquiring new behavior* by *changing both the associated input and output singular frames*.

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.

Observation: spectra stay. Building on prior observations of limited spectral drift in RLVR (Zhu et al., 2025b), we formalize and quantify *near-isospectrality*: in the unconstrained RLVR runs studied here, learned weight matrices remain close to the *fixed-spectrum families* of their base weights. Specifically, $W_t = U_t \Sigma_t V_t^\top$ is close to $U_t \Sigma_0 V_t^\top$, which shares the base spectrum Σ_0. Since this proximity arises without an explicit spectral constraint, we distinguish this observed proximity from an implicit optimization preference: dimension-aware calibration reveals no strong additional suppression of spectrum-changing motion beyond an isotropic reference. The key question is therefore not merely whether the spectra remain close, but whether they can be functionally reused: *are the remaining spectral changes necessary for the acquired behavior, or can the base spectra be reused?*

Functional regularity: spectral inheritance. We test this through complementary interventions. Restoring the base spectra after RLVR while retaining the learned frames preserves most of the acquired gains. Keeping the base spectra fixed throughout training while optimizing the frames yields gains over standard training, whereas spectrum-only training yields limited improvement.

Structure: two-sided frame adaptability. Spectral inheritance shows that the base spectra can be reused, but leaves open how the associated frame changes are structured. Could the learned endpoint be explained by a simpler transformation that remixes only within the incoming input and output singular subspaces, or that adapts only one side? At the tested truncation ranks, both alternatives leave substantially more of the checkpoint change unexplained than reusing the incoming spectrum while adapting both frames. This pattern also recurs across sequential RL stages with distinct objectives.

ISO: From spectral inheritance to isospectral Optimization. These findings motivate a simple design principle: *inherit the spectrum, optimize the frames*. We operationalize this inductive bias in **Isospectral Optimization (ISO)**, an RLVR-native framework that represents post-training change within the fixed-spectrum families of the base weights while keeping both singular frames adaptable. ISO has two complementary instantiations spanning the RLVR workflow (Fig. 1):

Offline: ISO-Merger enables checkpoint-only composition of shared-base specialists. A modular RLVR workflow trains domain specialists from a shared base and later consolidates them, often via an on-policy distillation stage (Zeng et al., 2026). Guided by spectral inheritance, ISO-Merger instead combines their singular-frame changes under the shared base spectra to recover complementary capabilities in a single model, without post-merge data, rollouts, gradient updates, or distillation.

Online: ISO-Optimizer for online RL learning. Conventional weight-space training applies optimizers such as AdamW or Muon directly to the model weights. ISO-Optimizer instead applies their update rules to the singular-frame variables while keeping the base spectra fixed, with retraction restoring frame orthogonality after each update. Across the reported reasoning and coding experiments at multiple model scales, its variants improve aggregate accuracy and reach matched accuracy in fewer actor updates than their weight-space counterparts.

Our contributions are summarized as follows:

- **Spectral inheritance in RLVR.** We quantify near-isospectrality in RLVR, validate the functional reuse of base spectra, and show that both singular frames must remain adaptable for a low-residual characterization of the observed checkpoint changes.
- **RLVR-Native Isospectral Optimization Framework.** We then introduce ISO, an RL-native optimziation framework that reuses the base spectra and represents reward-driven post-training change through the associated singular-frame coordinates.
- **Data-free offline RL expert composition.** We develop ISO-Merger for checkpoint-only composition of shared-base specialists. It recovers complementary specialist capabilities and achieves the highest aggregate scores among the evaluated data-free merging methods in both tested settings.
- **Fixed-spectrum online RLVR.** We develop ISO-Optimizer, instantiated with AdamW and Muon, and demonstrate improved aggregate accuracy in the reported mathematical-reasoning and coding experiments with 1.5B–4B models. On Qwen3-4B-Base, ISO-AdamW reaches AdamW's 400-update accuracy in 260 actor updates with 34% less wall-clock time, and exceeds AdamW by 2.2 percentage points at update 400.

2 Spectra Stay: Near-Isospectrality to Spectral Inheritance

Probe spectral structure. Building on prior observations of limited spectral drift (Zhu et al., 2025b), we formalize and quantify *near-isospectrality* and calibrate spectrum-changing motion

against an isotropic reference. We then establish *spectral inheritance* by answering: *are the remaining spectral changes necessary for the acquired behavior, or can the base spectra be reused?*

Notation and fixed-spectrum families. For $W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, let $q = \min\{d_{\text{out}}, d_{\text{in}}\}$ and write an ordered thin SVD $W = U\Sigma V^\top$, with $\Sigma = \mathrm{Diag}(\sigma(W))$. The vector $\sigma(W)$ lists all q singular values in nonincreasing order, including zeros. The frames satisfy $U \in \mathrm{St}(d_{\text{out}}, q)$ and $V \in \mathrm{St}(d_{\text{in}}, q)$, where $\mathrm{St}(d, r) = \{X \in \mathbb{R}^{d \times r} : X^\top X = I_r\}$ is the Stiefel manifold of matrices with orthonormal columns. For a base weight $W_0 = U_0\Sigma_0 V_0^\top$, define its *fixed-spectrum family*:

$$\begin{aligned} \mathcal{F}(W_0) &:= \{Z \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}} : \sigma(Z) = \sigma(W_0)\} \\ &= \{U\Sigma_0 V^\top : U \in \mathrm{St}(d_{\text{out}}, q),\ V \in \mathrm{St}(d_{\text{in}}, q)\}. \end{aligned} \tag{1}$$

This family fixes the singular values while allowing both frames to vary. The *base* is the checkpoint initializing the RL stage under study; it may be pretrained, SFT-trained, or already RL-trained.

2.1 A Precise Observation: Spectra Stay

Proposition 2.1 (Exact distance to the fixed-spectrum family)**.** *(Appendix C.1) For any $W, W_0 \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$,*

$$\mathrm{dist}_F\big(W, \mathcal{F}(W_0)\big) := \min_{Z \in \mathcal{F}(W_0)} \|W - Z\|_F = \|\sigma(W) - \sigma(W_0)\|_2. \tag{2}$$

For an ordered thin SVD $W = U\Sigma V^\top$, a minimizer is $U\Sigma_0 V^\top$, where $\Sigma_0 = \mathrm{Diag}(\sigma(W_0))$.

We normalize the distance by the base-weight norm and the full checkpoint displacement:

$$\delta_\Sigma := \|\sigma(W) - \sigma(W_0)\|_2 / \|W_0\|_F, \qquad \rho_\Sigma := \|\sigma(W) - \sigma(W_0)\|_2 / \|W - W_0\|_F. \tag{3}$$

These ratios require nonzero denominators; aggregation is specified in Appendix A.

Endpoint and training-time evidence. We analyze a released reasoning RLVR endpoint trained for over 3,000 updates (Liu et al., 2025b), with `DeepSeek-R1-Distill-Qwen-1.5B` (DS-1.5B) as its base. Across the analyzed layers, $\delta_\Sigma \approx 10^{-2}\%$, and the reported pooled relative residual is $\rho_\Sigma^{\text{pool}} = 0.03$. Thus, the endpoint is close to the base fixed-spectrum families relative to both the base-weight scale and the overall displacement. An independent unconstrained AdamW run, sampled every 10 actor updates, shows the same qualitative pattern at intermediate checkpoints (Appendix A.1; Figures 7 and 8). We call this descriptive proximity *near-isospectrality*.

Calibration: proximity versus preference. Near-isospectrality does not by itself establish preferential suppression of spectrum-changing motion. We therefore measure the dimension-normalized first-order spectrum-changing energy κ_{spec}, whose isotropic reference is one (Appendix A.2). In the DS-1.5B case, matrix-type means range from 1.02 to 1.35 for RLVR, versus 89 to 1364 for the preceding SFT transition. Thus, the diagnostic reveals no strong additional suppression in RLVR relative to this reference, while the SFT–RLVR contrast remains pronounced. Whether the remaining spectral changes are functionally necessary is a separate question, which we test next.

2.2 Spectral Inheritance: Reusing the Base Model's Spectrum

We address this question through two complementary interventions.

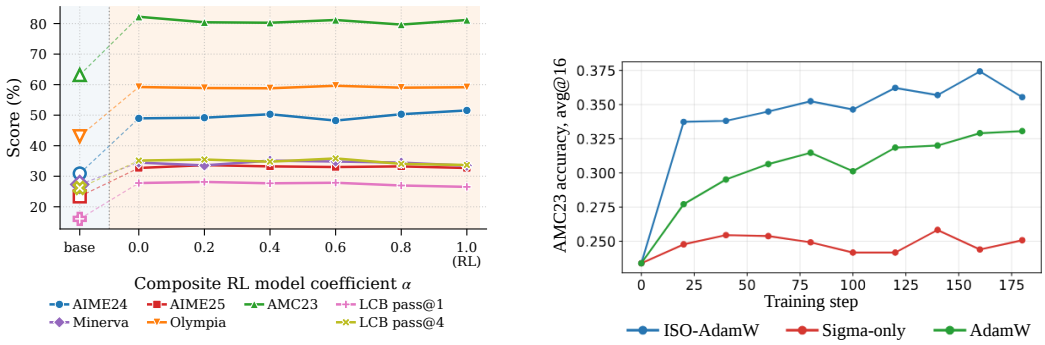
Restoring the base spectrum after training. For an RLVR endpoint (Liu et al., 2025b), $W_{\text{RL}} = U_{\text{RL}}\Sigma_{\text{RL}}V_{\text{RL}}^\top$, we retain its learned frames and interpolate only the spectrum:

$$\widetilde{W}(\alpha) = U_{\text{RL}}\big[(1-\alpha)\Sigma_0 + \alpha\Sigma_{\text{RL}}\big]V_{\text{RL}}^\top, \qquad \alpha \in [0, 1]. \tag{4}$$

At $\alpha = 0$, Proposition 2.1 identifies $\widetilde{W}(0)$ as a nearest representative in $\mathcal{F}(W_0)$; at $\alpha = 1$, the original RL checkpoint is recovered. Restoration preserves most acquired gains (Fig. 2(a)). The converse substitution, placing the RL spectrum in the base frames, does not yield comparable gains (Fig. 14).

Learning with the base spectrum fixed. For each transformed matrix, we train $W = U\Sigma_0 V^\top$ from the base checkpoint, optimizing (U, V) while maintaining their orthonormality. This parameterization acquires strong reasoning gains and outperforms AdamW in the illustrated run (Fig. 2(b)). By contrast, training only Σ with fixed base frames, a restricted control with q trainable scalars per matrix, yields limited improvement, indicating that spectral rescaling alone does not recover comparable gains.

Together, these interventions support *spectral inheritance*: the base spectra are functionally reusable for both retaining acquired gains and supporting new learning.



(a) Spectrum restoration after training.

(b) Fixed-spectrum and spectrum-only training.

Figure 2: **Functional evidence for spectral inheritance. (a)** Restoring the base spectrum while retaining RL-trained frames preserves most acquired gains. **(b)** Fixed-spectrum frame training on a 1.7B model supports strong gains, whereas the restricted spectrum-only control yields limited improvement under the tested recipe.

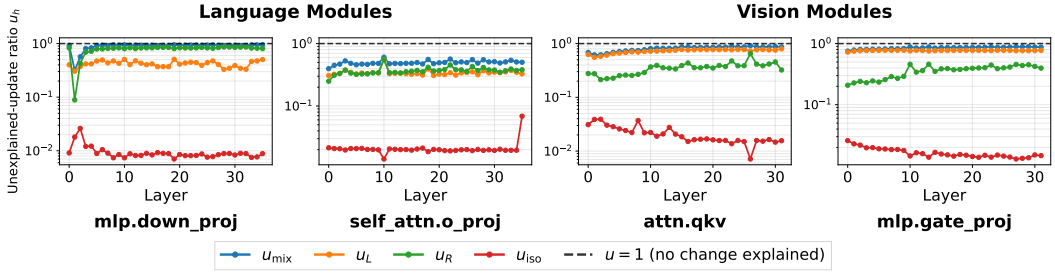


Figure 3: **Two-sided frame adaptability in endpoint reconstruction.** Unexplained-update ratios for $W_0 \to W_2$ at $r = \lfloor 0.9q \rfloor$, across layers of representative language and vision modules. Retaining one or both incoming singular subspaces leaves substantially larger residuals than retaining the incoming spectrum while adapting both frames. The dashed line $u_h = 1$ marks the error of the unchanged incoming checkpoint.

3 Frames Move: What Changes Under Spectral Inheritance?

Spectral inheritance identifies what can be reused, but not whether the associated frame changes admit a simpler description. Could the learned endpoint be explained by remixing within both incoming singular subspaces, or retaining one while allowing the other side to change?

A reconstruction test. For a transition $W_j \to W_i$, write the top-r truncated SVD as $W_t^{(r)} = U_t^{(r)} \Sigma_t^{(r)} (V_t^{(r)})^\top$, $t \in \{i, j\}$. We use $r = \lfloor 0.9q \rfloor < q$, requiring positive rank-r boundary gaps at both endpoints. Let $A = W_i^{(r)}$ and define the incoming output and input projectors $P_j = U_j^{(r)} (U_j^{(r)})^\top, Q_j = V_j^{(r)} (V_j^{(r)})^\top$. We fit the same target A under four restrictions: remixing within both incoming subspaces (mix), retaining only the output (L) or input (R) subspace, or retaining the incoming spectrum while adapting both frames (iso). The Frobenius-optimal reconstructions are

$$\widehat{W}_{\text{mix}} = P_j A Q_j, \quad \widehat{W}_L = P_j A, \quad \widehat{W}_R = A Q_j, \quad \widehat{W}_{\text{iso}} = U_i^{(r)} \Sigma_j^{(r)} (V_i^{(r)})^\top. \tag{5}$$

The first three classes freely refit their remaining mapping and spectrum. Proofs are in Appendix C.3.

For a nonzero truncated update, define the *unexplained-update ratio*

$$u_h := \frac{\|W_i^{(r)} - \widehat{W}_h\|_F}{\|W_i^{(r)} - W_j^{(r)}\|_F}, \qquad h \in \{\text{mix}, L, R, \text{iso}\}. \tag{6}$$

Since $W_j^{(r)}$ is feasible under every restriction, $0 \le u_h \le 1$: zero denotes exact reconstruction, while one matches the error of the unchanged incoming checkpoint. This retrospective test concerns endpoint representation, not the optimization trajectory or necessity for task performance.

Sequential RL and objective shift. We analyze language and vision modules in a two-stage RL pipeline initialized from `Qwen2.5-VL-3B-Instruct` (W_0) (Yuan et al., 2025). Spatial-

reasoning RL produces W_1, followed by embodied-manipulation RL producing W_2. For $W_0 \to W_2$, the median ratios $(u_{\text{mix}}, u_L, u_R, u_{\text{iso}})$ are $(87\%, 45\%, 42\%, 1.8\%)$, respectively. Fig. 11 illustrates this separation across representative modules. The same qualitative pattern holds for $W_1 \to W_2$, after re-anchoring at an already RL-trained checkpoint and changing the objective. Full results for all three transitions and rank sensitivity are reported in Appendix C.4.

Among the tested restrictions, retaining the incoming spectrum while adapting both frames yields a low-residual description of the observed rank-r checkpoint changes. Together with the functional evidence in Section 2.2, this motivates ISO's spectrum-inheriting parameterization $W = U\Sigma_0 V^\top$.

4 From Spectral Inheritance to Isospectral Optimization (ISO)

Sections 2–3 motivate a separation between reusable spectra and adaptable singular frames. We adopt this separation as an inductive bias, *inherit the spectrum, optimize the frames*, rather than as an exact update law for unconstrained RLVR. **Isospectral Optimization (ISO)** operationalizes this principle through a shared parameterization for offline expert composition and online RLVR optimization.

A shared fixed-spectrum parameterization. For a base matrix $W_0 = U_0\Sigma_0 V_0^\top$, ISO uses

$$W(U, V) = U\Sigma_0 V^\top, \qquad U \in \mathrm{St}(d_{\text{out}}, q), \quad V \in \mathrm{St}(d_{\text{in}}, q), \tag{7}$$

where $q = \min\{d_{\text{out}}, d_{\text{in}}\}$. Unlike the rank-$r$ diagnostic in Section 3, this parameterization retains all q singular modes. Composition and optimization act on (U, V), while Σ_0 is inherited from the base. Frame orthonormality ensures $W(U, V) \in \mathcal{F}(W_0)$ in exact arithmetic.

Proposition 4.1 (Local geometry of ISO). *Let $W = U\Sigma_0 V^\top$ satisfy Equation 7 with positive, simple singular values. For Stiefel tangent directions $\xi_U \in T_U\mathrm{St}(d_{\text{out}}, q)$ and $\xi_V \in T_V\mathrm{St}(d_{\text{in}}, q)$,*

$$\begin{aligned} \dot{W} &= \xi_U \Sigma_0 V^\top + U\Sigma_0 \xi_V^\top, \\ \mathrm{diag}(U^\top \dot{W} V) &= 0. \end{aligned} \tag{8}$$

Thus, feasible frame motion preserves the spectrum to first order, while orthonormal reconstruction preserves it exactly (proof in Appendix D.1). The reconstruction map also provides ISO-Merger's local effect proxies through linearization and ISO-Optimizer's factor gradients through the chain rule.

4.1 ISO-Merger: Merging RL Experts Without Rollouts

Training a single RLVR policy across heterogeneous domains can be costly and unstable. A modular alternative trains specialists from a shared base and consolidates them, often through an additional on-policy distillation stage (Kevin Lu & Thinking Machines Lab, 2025; Zeng et al., 2026). ISO-Merger instead seeks to recover their complementary capabilities directly from checkpoints, without post-merge data, rollouts, gradient updates, or distillation.

Setup and shared coordinates. For each base weight matrix $W_0 = U_0\Sigma_0 V_0^\top$, let $W_i = U_i\Sigma_i V_i^\top$, $i = 1, \ldots, K$, be the corresponding matrices from K specialists trained from the same base. All singular values are ordered nonincreasingly. Rather than combining Euclidean task vectors $W_i - W_0$, ISO-Merger represents each expert by $U_i\Sigma_0 V_i^\top \in \mathcal{F}(W_0)$ and composes its frame changes at the common anchor (U_0, V_0).

Aligned tangent directions. Singular-vector pairs admit a joint sign ambiguity. For each expert and column k, we set $s_{i,k} = \mathrm{sign}(\langle u_{0,k}, u_{i,k}\rangle)$, with $\mathrm{sign}(0) = 1$, and apply $(u_{i,k}, v_{i,k}) \leftarrow (s_{i,k}u_{i,k}, s_{i,k}v_{i,k})$. This leaves W_i unchanged while removing artificial joint-sign differences from the frame displacements. For an orthonormal anchor X, the Frobenius-orthogonal Stiefel tangent projection is

$$\Pi_X(Z) = Z - X\,\mathrm{sym}(X^\top Z), \qquad \mathrm{sym}(A) = \tfrac{1}{2}(A + A^\top). \tag{9}$$

Using the aligned frames, we obtain

$$\xi_{U,i} = \Pi_{U_0}(U_i - U_0), \qquad \xi_{V,i} = \Pi_{V_0}(V_i - V_0). \tag{10}$$

The common anchor makes these directions directly composable in shared frame coordinates. To limit interference from trailing modes, we zero all but the leading $\mathrm{round}(\rho_{\text{keep}} q)$ columns of each projected displacement, obtaining $\widetilde{\xi}_{U,i}$ and $\widetilde{\xi}_{V,i}$. We use $\rho_{\text{keep}} = 0.9$ unless otherwise stated. Masking acts on frame displacements, not on Σ_0, and need not preserve tangent feasibility; we restore it after aggregation.

Retention-aware composition. To account for overlap between expert directions, we linearize the reconstruction map at (U_0, V_0) and form local first-order effect proxies

$$g_i = \widetilde{\xi}_{U,i}\Sigma_0 V_0^\top + U_0\Sigma_0\widetilde{\xi}_{V,i}^\top. \tag{11}$$

Their Gram matrix $\Gamma_{ij} = \langle g_i, g_j\rangle_F$ measures pairwise interactions. For the merged proxy $g(c) = \sum_{i=1}^K c_i g_i$, we target unit self-retention: its projection coefficient along each nonzero g_i should satisfy $\langle g(c), g_i\rangle_F / \|g_i\|_F^2 = 1$. The ideal target is therefore $\Gamma c = \mathrm{diag}(\Gamma)$.

Let $\mathcal{I} = \{i : \|g_i\|_F > 0\}$. We set coefficients outside $\mathcal{I}$ to zero and, when $\mathcal{I} \neq \varnothing$, solve and clip the active system:

$$\begin{aligned}\big(\Gamma_{\mathcal{I}\mathcal{I}} + \lambda_{\mathrm{ridge}} I_{|\mathcal{I}|}\big)\bar{c}_{\mathcal{I}} &= \mathrm{diag}(\Gamma_{\mathcal{I}\mathcal{I}}),\\ c^\star_{\mathcal{I}} &= \mathrm{clip}(\bar{c}_{\mathcal{I}}; c_{\min}, c_{\max}).\end{aligned} \tag{12}$$

Ridge stabilization handles an ill-conditioned Gram matrix, while clipping limits extreme coefficients. Unit self-retention is a local proxy-space target, not a guarantee of capability retention.

Retraction and reconstruction. We reproject the weighted aggregate onto each base tangent space: $\xi_{U,\star} = \Pi_{U_0}(\sum_i c_i^\star \widetilde{\xi}_{U,i})$, and analogously obtain $\xi_{V,\star}$. For a thin SVD $X = P_X S_X Q_X^\top$, the polar factor $\mathrm{polar}(X) = P_X Q_X^\top$ has orthonormal columns. We retract the updated frames and reconstruct the merged weight:

$$\begin{aligned}U_\star &= \mathrm{polar}(U_0 + \xi_{U,\star}), \qquad V_\star = \mathrm{polar}(V_0 + \xi_{V,\star}),\\ W_\star &= U_\star \Sigma_0 V_\star^\top.\end{aligned} \tag{13}$$

The resulting matrix shares the base spectrum up to numerical precision. We apply this construction to 2D projection, embedding, and unembedding matrices; one-dimensional parameters use an arithmetic average across experts. Alignment, projection, masking, coefficient settings, and retraction details, together with the complete algorithm, are provided in Appendix F.

4.2 ISO-Optimizer: Online Fixed-Spectrum RLVR Training

ISO-Optimizer defines a generic transformation $\mathrm{Opt} \mapsto \mathrm{ISO}[\mathrm{Opt}]$: the base optimizer supplies the update equations and state machinery, while ISO supplies the fixed-spectrum parameterization and feasibility restoration. For each transformed matrix, we initialize $(U, V) = (U_0, V_0)$ and train $W = U\Sigma_0 V^\top$ with Σ_0 fixed. Using AdamW or Muon yields ISO-AdamW or ISO-Muon, respectively.

Factor gradients and tentative updates. Let $G_W = \nabla_W \mathcal{L}(W)$ be the gradient of the RLVR loss at $W = U\Sigma_0 V^\top$. The chain rule gives the Euclidean factor gradients

$$G_U = G_W V\Sigma_0, \qquad G_V = G_W^\top U\Sigma_0. \tag{14}$$

Let $\mathrm{Opt}(X, G, s)$ denote a base-optimizer update of variable X with gradient G and state s. We apply the update rule to both frames using separate states:

$$\begin{aligned}(\bar{U}, s_U^+) &= \mathrm{Opt}(U, G_U, s_U),\\ (\bar{V}, s_V^+) &= \mathrm{Opt}(V, G_V, s_V).\end{aligned} \tag{15}$$

For example, ISO-AdamW maintains first- and second-moment buffers for each frame rather than for the dense weight. The base optimizer receives the ambient factor gradients; the tentative factors need not remain orthonormal.

Feasibility restoration. We apply the polar map and reconstruct the weight:

$$\begin{aligned}U^+ &= \mathrm{polar}(\bar{U}), \qquad V^+ = \mathrm{polar}(\bar{V}),\\ W^+ &= U^+\Sigma_0(V^+)^\top.\end{aligned} \tag{16}$$

Thus, $W^+ \in \mathcal{F}(W_0)$ in exact arithmetic. Both the update rule and optimizer states operate in frame coordinates: ISO is not a weight-space optimizer step followed by projection onto $\mathcal{F}(W_0)$. Although it introduces two factor variables, it does not enlarge the represented weight-space class; the reconstructed weights remain constrained to the base fixed-spectrum family. The factor parameterization and optimizer-state dynamics are both part of the method.

Implementation overhead. The main additional structured operation in ISO-Optimizer is the polar retraction applied after each factor update. The additional operations also include factor-gradient construction, factor updates, and weight reconstruction. We implement it using an FP64 SVD-based polar computation for numerical stability. Following a DION-style distributed implementation (Ahn et al., 2025), we distribute the matrix retractions across multiple GPUs. In our RLVR profiling, the additional optimizer cost increases end-to-end step time by approximately 1.2%, as reported in Section 5.2.

The additional storage remains manageable in our sharded RLVR setting: on `Qwen3-4B-Base`, ISO-AdamW increases model-wide moment memory by 31%, while the measured peak-memory increase under identical 16-way FSDP is 1.8 GB per GPU, including factor storage (Appendix I).

Despite the added overhead, ISO uses $1.5\times$ fewer updates and reduces wall-clock time to matched accuracy by 34% (Section 5.2).

5 Experiments

We evaluate ISO through its two complementary instantiations: ISO-Merger for recovering specialist capabilities through checkpoint-only composition, and ISO-Optimizer for improving accuracy and accelerating RLVR training.

5.1 ISO-Merger: Composing Heterogeneous RL Experts Without Rollouts

Table 1: **Merging three RL experts on Qwen2.5-7B-Instruct.** Coding reports pass accuracy (ACC) and unit-test accuracy (UT) on LiveBench (LB) and LiveCodeBench (LCB). Tool use follows BFCL v4 with Live and Non-Live averages; memory uses RULER's recurrent setting at 32K–64K context lengths for HotpotQA and SQuAD. Avg. is the mean of the ten metric columns. Bold marks column-wise best scores separately among experts and merged models. The final row reports ISO-Merger's score as a percentage of the corresponding domain expert's score, using the reported means. Shading indicates Experts and Ours.

	Coding				Tool Use		Memory				
	LB		LCB v2		Live	Non-Live	HotpotQA		SQuAD		
Method	ACC	UT	ACC	UT	ACC	ACC	32K	64K	32K	64K	Avg
Base Qwen et al. (2025)	36.25 ± 0.86	48.27 ± 1.44	28.31 ± 0.51	43.08 ± 0.82	62.96	69.26	49.87 ± 1.45	45.54 ± 0.58	58.90 ± 1.07	56.77 ± 0.61	49.92
RLVR-Coder Wang et al. (2025c)	**37.42 ± 1.22**	49.42 ± 1.60	**30.17 ± 0.99**	**44.24 ± 0.99**	63.37	68.07	50.21 ± 0.16	45.61 ± 0.33	60.51 ± 0.70	56.84 ± 0.42	50.59
RLVR-Tool Qian et al. (2025)	36.87 ± 1.79	48.08 ± 0.82	28.20 ± 1.95	42.43 ± 1.90	**71.83**	**81.82**	50.39 ± 0.68	45.33 ± 0.26	59.73 ± 1.18	57.86 ± 0.52	52.25
RLVR-Memory Yu et al. (2025)	37.32 ± 1.37	**50.05 ± 1.61**	27.41 ± 2.99	41.70 ± 4.63	63.82	72.61	**78.60 ± 0.54**	**77.95 ± 0.29**	**79.98 ± 0.38**	**78.52 ± 0.22**	**60.80**
Task Arithmetic Ilharco et al. (2022)	38.85 ± 0.70	52.18 ± 0.40	31.55 ± 0.18	47.06 ± 0.58	**73.73**	**82.86**	72.22 ± 0.22	70.49 ± 0.10	75.03 ± 0.29	74.95 ± 0.32	61.89
TIES Yadav et al. (2023)	40.30 ± 2.18	52.04 ± 2.50	31.46 ± 0.35	47.33 ± 0.30	68.24	76.54	76.42 ± 0.36	75.24 ± 0.14	78.27 ± 0.46	77.86 ± 0.59	62.37
TSV Gargiulo et al. (2025)	36.96 ± 0.86	49.23 ± 0.59	28.92 ± 1.79	43.37 ± 3.25	73.64	82.69	75.37 ± 0.29	74.10 ± 0.30	78.30 ± 0.17	77.26 ± 0.78	61.98
RAM Yuan et al. (2026)	38.59 ± 0.79	50.38 ± 1.53	31.31 ± 0.28	46.33 ± 0.30	72.56	81.89	76.17 ± 0.20	75.29 ± 0.49	78.42 ± 0.47	77.83 ± 0.46	62.88
OrthoMerge-G-TIES Yang et al. (2026)	40.74 ± 2.00	**53.22 ± 1.90**	**31.98 ± 0.84**	**47.48 ± 0.77**	66.98	78.17	74.79 ± 0.49	74.20 ± 0.49	77.75 ± 0.12	76.32 ± 0.21	62.16
ISO-Merger	**41.81 ± 2.00**	52.73 ± 1.57	31.06 ± 1.88	45.69 ± 3.06	72.32	81.07	**79.46 ± 0.78**	**76.77 ± 0.27**	**79.10 ± 0.14**	**78.01 ± 0.32**	**63.80**
Expert-relative score (%)	111.7	106.7	102.9	103.3	100.7	99.1	101.1	98.5	98.9	99.4	–

Setup. We evaluate ISO-Merger in its intended setting: composing multiple RLVR specialists trained from the same base model, without post-merge data, rollouts, or distillation. We consider two settings: (1) three specialists for coding (Wang et al., 2025c), tool use (Qian et al., 2025), and long-context memory (Yu et al., 2025), initialized from `Qwen2.5-7B-Instruct` (Qwen et al., 2025); (2) two specialists for coding (Wang et al., 2025a) and math (He et al., 2025a), initialized from `DeepSeek-R1-Distill-Qwen-1.5B` (DS-1.5B). We compare against representative data-free merging baselines: Task Arithmetic (Ilharco et al., 2022), TIES (Yadav et al., 2023), TSV-Merge (Gargiulo et al., 2025), RAM (Yuan et al., 2026), and OrthoMerge-G-TIES (Yang et al., 2026). We evaluate immediately after merging, with no fine-tuning. Stochastic evaluations are repeated three times, with reported mean and standard deviation across runs. Unless otherwise specified, we report average@16. Full evaluation protocols, baseline configurations, best@4/worst@4 results, and ablation studies are provided in Appendix G.

Results. ISO-Merger achieves the highest aggregate scores among the compared data-free methods in both settings: 63.80 versus 62.88 on Qwen2.5-7B (Table 1), and 44.38 versus 43.52 on DS-1.5B (Table 3, Appendix G.3.1). On Qwen2.5-7B, ISO-Merger achieves 106.2%, 99.8%, and 99.5% of the corresponding specialists' domain-average scores on coding, tool use, and memory, respectively. By comparison, Task Arithmetic reaches 101.9% on tool use but only 92.9% on memory,

Table 2: **Math RLVR results.** We report average@16 accuracy.

Model	Method	AIME24	AIME25	AMC23	Minerva	Olympiad	Avg.
Qwen3-1.7B-Base	Base	3.33	2.50	23.34	18.43	19.19	13.36
	AdamW	14.38	10.21	44.58	33.02	38.80	28.20
	Muon	15.22	11.26	**45.51**	32.96	40.09	29.01
	ISO-AdamW	**16.67**	**11.67**	44.50	**33.92**	**40.30**	**29.41**
Qwen3-4B-Base	Base	5.00	3.96	26.58	20.43	21.95	15.58
	AdamW	32.92	22.08	67.77	45.29	56.02	44.82
	Muon	32.29	25.00	66.34	46.12	**57.82**	45.51
	ISO-AdamW	**33.54**	**28.13**	**68.60**	**46.97**	57.64	**46.98**

(a) Qwen3-1.7B-Base (b) Qwen3-4B-Base



Figure 4: **Aggregate math accuracy during RLVR training.** ISO-AdamW uses a single learning rate of 1×10^{-6}, whereas the weight-space AdamW baselines are tuned over the sweep described in the text. AdamW with a learning rate of 7.5×10^{-7} performs strictly worse and is omitted from the figure for clarity.

highlighting ISO-Merger's more balanced recovery of specialist performance. On the applicable metrics, ISO-Merger also improves aggregate worst@4 over the strongest baselines by 1.62 and 1.36 percentage points in the two settings, respectively, while maintaining comparable aggregate best@4 (Appendix G.3.3). Together, these results support the practical value of spectral inheritance for recovering complementary specialist capabilities in a single model without post-merge training. We further examine the contributions of spectral rebasing, retention-aware composition, and mode masking ($\rho_{\text{keep}} = 0.9$) through ablations in Appendix G.4.

5.2 ISO-Optimizer: Improving RLVR Accuracy and Training Efficiency

Setup. Motivated by spectral inheritance, ISO-Optimizer reuses the base spectra and applies conventional optimizers directly to the associated singular frames. We evaluate accuracy and training efficiency on mathematical reasoning and coding against the corresponding weight-space optimizers. For the main math experiments, we train `Qwen3-1.7B-Base` and `Qwen3-4B-Base` (Qwen Team, 2025) on DeepMath-103K (He et al., 2025b) for 400 actor updates. ISO-AdamW uses a frame learning rate of 10^{-6} at both scales. We sweep AdamW learning rates over $\{7.5 \times 10^{-7}, 10^{-6}, 2 \times 10^{-6}, 3 \times 10^{-6}\}$ and Muon learning rates over $\{2.5 \times 10^{-5}, 5 \times 10^{-5}, 7.5 \times 10^{-5}\}$, with an additional AdamW run at 5×10^{-6} on 4B. We evaluate AIME24, AIME25, AMC23, Minerva, and OlympiadBench. Full configurations and evaluation protocols are provided in Appendix I.

Math results. ISO-AdamW achieves the highest aggregate average@16 accuracy at both model scales (Tab. 2), improving over the strongest tested AdamW baselines by 1.21 and 2.16 points on Qwen3-1.7B and Qwen3-4B, respectively. It also exceeds Muon in aggregate and achieves the highest score on four of the five benchmarks at each scale. The learning curves further show that ISO-AdamW reaches the strongest AdamW run's 400-update accuracy in fewer actor updates at both scales (Fig. 4). We also ran an additional AdamW experiment at a larger learning rate of 5×10^{-6}. It peaks early and then declines on Qwen3-4B-Base (Appendix Fig. 15).

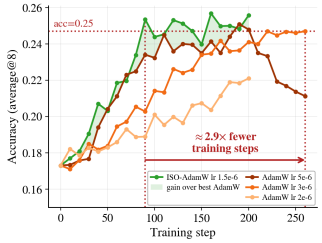


Figure 5: **Coding accuracy on DS-1.5B.**

Coding results. Beyond math, we train `DS-1.5B` following ArcherCoder (Wang et al., 2025a).

An initial AdamW sweep over $\{1, 2, 3\} \times 10^{-6}$ favors larger learning rates, so we extend the sweep to 5×10^{-6}. We also increase ISO-AdamW's frame learning rate from 10^{-6} to 1.5×10^{-6} for this setting. To limit multi-epoching on the smaller coding dataset, we use 200 actor updates for the main comparison. ISO-AdamW achieves an average@8 of 25.58 across LiveCodeBench v5 and v6, versus

24.80 for the strongest AdamW baseline tested (Tab. 12). The learning curves show ISO-AdamW ahead at most evaluation checkpoints. Extending the AdamW runs at 3×10^{-6} and 5×10^{-6} to 260 updates still does not match ISO-AdamW's peak of 25.7, reached within 200 updates (Fig. 5).

Wall-clock efficiency and overhead. We profile `Qwen3-4B-Base` using the ISO-AdamW (10^{-6}) and AdamW (2×10^{-6}) runs in Fig. 4b. Additional optimizer computation costs about 49 seconds per step, only 1.2% of the profiled end-to-end step time (Fig. 6, left). ISO-AdamW reaches AdamW's 400-update accuracy at update 260, saving 124h (34%), and finishes 2.2 points higher at update 400 (Fig. 6, right). Its roughly 20% higher average step time mainly reflects an earlier increase in response length (Fig. 6, left), consistent with faster learning rather than optimizer overhead. Thus, ISO-AdamW not only uses fewer actor updates, but also translates those updates saving in less wall-clock time. Full profiling details are provided in Appendix I.

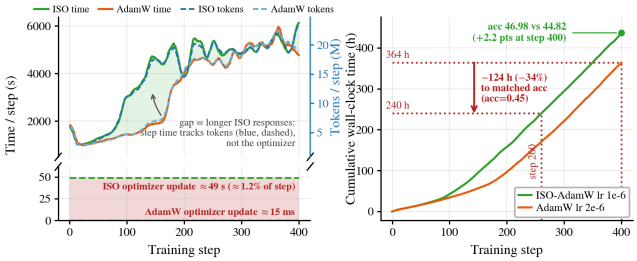


Figure 6: **Wall-clock cost of ISO vs. AdamW on `Qwen3-4B-Base`** Left, top: per-step time (solid, left axis) and tokens per step (dashed blue, right axis); Left, bottom: the optimizer update alone. Right: cumulative wall-clock time

ISO-Muon variant. The benefit also extends to Muon. On `Qwen3-4B-Base`, ISO-Muon reaches the strongest tested Muon run's final accuracy by update 300 and attains a higher final aggregate score (46.2 vs. 45.5; Appendix Fig. 17).

Comparison with Pion. Pion (Shi et al., 2026) is a concurrent spectrum-preserving optimizer that updates W through orthogonal rotations, with its own momentum and step-size rules. Its RLVR configuration alternates between left- and right-side transformations. ISO instead provides a generic optimizer transformation, applying a chosen base optimizer's update rule directly to both singular frames. We use Pion's official implementation, sweeping learning rates over $\{1, 2, 3\} \times 10^{-6}$ on Qwen3-1.7B-Base. Due to computational constraints, we use its published RLVR learning rate of 1×10^{-6} on Qwen3-4B-Base without further tuning. Among completed 400-update runs, Pion outperforms the 10^{-6} AdamW baseline but not the strongest tested AdamW configuration, while ISO-AdamW achieves the highest final accuracy at both scales. On 4B, ISO-AdamW reaches Pion's 400-update accuracy using approximately 40% as many actor updates (Appendix Fig. 16).

Ablation: Does learning the spectrum help? We compare ISO-AdamW with a variant that additionally trains the diagonal spectrum Σ, initialized at Σ_0, while retaining the same factorization $W = U\Sigma V^\top$ and orthonormality constraints on the frames. On `Qwen3-1.7B-Base`, the fixed- and trainable-spectrum variants exhibit similar learning trajectories, reaching aggregate accuracies of 29.4 and 29.6, respectively, at update 400 (Appendix Fig. 18). The learned spectrum remains close to initialization, with a relative drift of 0.0065%. The comparable observed accuracy supports reusing the base spectrum, reinforcing the rationale for ISO's frame-coordinate formulation: inherit the spectrum and directly optimize the adaptable singular frames.

Ablation: Random-seed sensitivity. We train `Qwen3-1.7B-Base` with two independent seeds per method under a supplementary 8K-context configuration to reduce the computational cost, fixing the learning rate at 10^{-6} for both AdamW and ISO-AdamW. ISO-AdamW achieves aggregate accuracies of 27.6 and 27.3, compared with 24.4 and 24.8 for AdamW. The absolute run-to-run differences are 0.30 and 0.40 percentage points, respectively, under this fixed-learning-rate 8K configuration.

6 Conclusion

We identify and functionally validate *spectral inheritance*: RLVR can reuse the base model's weight spectra while acquiring new behavior through changes in both singular frames. This separation motivates *Isospectral Optimization* (ISO), an RLVR-native framework for offline composition and online learning. ISO-Merger combines shared-base specialists without post-merge data or training, achieving the strongest aggregate performance among the compared data-free methods. ISO-Optimizer applies conventional optimizers to the frames, improving reasoning and coding accuracy and reaching matched accuracy in fewer actor updates. Together, these results support a reusable design principle for RLVR: *inherit the spectrum, optimize the frames*.

AI USE STATEMENT.

Generative AI tools were used to assist with language editing, presentation, and organization of the manuscript, and to provide feedback on clarity. All technical claims, experimental results, analyses, and final manuscript content were reviewed and verified by the authors, who take full responsibility for the work.

REPRODUCIBILITY STATEMENT.

We provide the training and evaluation configurations, computational platforms, and implementation details used in our experiments throughout the main paper and appendix. These include the settings necessary to reproduce our optimization procedures, training runs, and evaluations. We aim to report sufficient experimental and implementation details to enable faithful reproduction of the results presented in this work.

REFERENCES

Kwangjun Ahn, Byron Xu, Natalie Abreu, Ying Fan, Gagik Magakyan, Pratyusha Sharma, Zheng Zhan, and John Langford. Dion: Distributed orthonormalized updates. *arXiv preprint arXiv:2504.05295*, 2025.

Jeremy Bernstein. Modular manifolds. *Thinking Machines Lab: Connectionism*, 2025. doi: 10.64434/tml.20250926. https://thinkingmachines.ai/blog/modular-manifolds/.

Antonio Andrea Gargiulo, Donato Crisostomi, Maria Sofia Bucarelli, Simone Scardapane, Fabrizio Silvestri, and Emanuele Rodola. Task singular vectors: Reducing task interference in model merging. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 18695–18705, 2025.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Bingxiang He, Zekai Qu, Zeyuan Liu, Yinghao Chen, Yuxin Zuo, Cheng Qian, Kaiyan Zhang, Weize Chen, Chaojun Xiao, Ganqu Cui, et al. Justrl: Scaling a 1.5 b llm with a simple rl recipe. *arXiv preprint arXiv:2512.16649*, 2025a.

Zhiwei He, Tian Liang, Jiahao Xu, Qiuzhi Liu, Xingyu Chen, Yue Wang, Linfeng Song, Dian Yu, Zhenwen Liang, Wenxuan Wang, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning. 2025b. URL https://arxiv.org/abs/2504.11456.

Jian Hu, Mingjie Liu, Ximing Lu, Fang Wu, Zaid Harchaoui, Shizhe Diao, Yejin Choi, Pavlo Molchanov, Jun Yang, Jan Kautz, et al. Brorl: Scaling reinforcement learning via broadened exploration. *arXiv preprint arXiv:2510.01180*, 2025.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. *arXiv preprint arXiv:2212.04089*, 2022.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*, 2024.

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

Kevin Lu and Thinking Machines Lab. On-policy distillation. *Thinking Machines Lab: Connectionism*, 2025. doi: 10.64434/tml.20251026. https://thinkingmachines.ai/blog/on-policy-distillation.

Zhizhong Li, Sina Sajadmanesh, Jingtao Li, and Lingjuan Lyu. Stella: Subspace learning in low-rank adaptation using stiefel manifold. In *Advances in Neural Information Processing Systems*, volume 38. Curran Associates, Inc., 2025.

Jingyuan Liu, Jianlin Su, Xingcheng Yao, Zhejun Jiang, Guokun Lai, Yulun Du, Yidao Qin, Weixin Xu, Enzhe Lu, Junjie Yan, et al. Muon is scalable for llm training. *arXiv preprint arXiv:2502.16982*, 2025a.

Mingjie Liu, Shizhe Diao, Ximing Lu, Jian Hu, Xin Dong, Yejin Choi, Jan Kautz, and Yi Dong. Prorl: Prolonged reinforcement learning expands reasoning boundaries in large language models. *arXiv preprint arXiv:2505.24864*, 2025b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Sagnik Mukherjee, Lifan Yuan, Dilek Hakkani-Tur, and Hao Peng. Reinforcement learning finetunes small subnetworks in large language models. *arXiv preprint arXiv:2505.11711*, 2025.

Cheng Qian, Emre Can Acikgoz, Qi He, Hongru Wang, Xiusi Chen, Dilek Hakkani-Tür, Gokhan Tur, and Heng Ji. Toolrl: Reward is all tool learning needs. *arXiv preprint arXiv:2504.13958*, 2025.

Zeju Qiu, Simon Buchholz, Tim Xiao, Maximilian Dax, Bernhard Schölkopf, and Weiyang Liu. Reparameterized llm training via orthogonal equivalence transformation. *Advances in Neural Information Processing Systems*, 38:140775–140821, 2026a.

Zeju Qiu, Lixin Liu, Adrian Weller, Han Shi, and Weiyang Liu. Poet-x: Memory-efficient llm training by scaling orthogonal transformation. *arXiv preprint arXiv:2603.05500*, 2026b.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Scale AI. The next frontier of data training: Rl environments, February 2026. URL https://scale.com/blog/rl-environments.

Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. Rl's razor: Why online reinforcement learning forgets less. *arXiv preprint arXiv:2509.04259*, 2025.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, pp. 1279–1297, 2025.

Kexuan Shi, Hanxuan Li, Zeju Qiu, Yandong Wen, Simon Buchholz, and Weiyang Liu. Pion: A spectrum-preserving optimizer via orthogonal equivalence transformation. *arXiv preprint arXiv:2605.12492*, 2026.

Fahim Tajwar, Guanning Zeng, Yueer Zhou, Yuda Song, Daman Arora, Yiding Jiang, Jeff Schneider, Ruslan Salakhutdinov, Haiwen Feng, and Andrea Zanette. Maximum likelihood reinforcement learning. *arXiv preprint arXiv:2602.02710*, 2026.

Jiakang Wang, Runze Liu, Lei Lin, Wenping Hu, Xiu Li, Fuzheng Zhang, Guorui Zhou, and Kun Gai. Aspo: Asymmetric importance sampling policy optimization. *arXiv preprint arXiv:2510.06062*, 2025a.

Jiakang Wang, Runze Liu, Fuzheng Zhang, Xiu Li, and Guorui Zhou. Stabilizing knowledge, promoting reasoning: Dual-token constraints for rlvr. *arXiv preprint arXiv:2507.15778*, 2025b.

Yinjie Wang, Ling Yang, Ye Tian, Ke Shen, and Mengdi Wang. Co-evolving llm coder and unit tester via reinforcement learning. *arXiv preprint arXiv:2506.03136*, 2025c.

Kaiyue Wen, David Hall, Tengyu Ma, and Percy Liang. Fantastic pretraining optimizers and where to find them. *arXiv preprint arXiv:2509.02046*, 2025.

Fang Wu, Weihao Xuan, Ximing Lu, Mingjie Liu, Yi Dong, Zaid Harchaoui, and Yejin Choi. The invisible leash: Why rlvr may or may not escape its origin. *arXiv preprint arXiv:2507.14843*, 2025.

xAI. Grok: Ai assistant, 2025. URL https://x.ai/grok. Accessed: 2025-09-24, continuously updated.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. *Advances in neural information processing systems*, 36:7093–7115, 2023.

Sihan Yang, Kexuan Shi, and Weiyang Liu. Orthogonal model merging. *arXiv preprint arXiv:2602.05943*, 2026.

Hongli Yu, Tinghong Chen, Jiangtao Feng, Jiangjie Chen, Weinan Dai, Qiying Yu, Ya-Qin Zhang, Wei-Ying Ma, Jingjing Liu, Mingxuan Wang, et al. Memagent: Reshaping long-context llm with multi-conv rl-based memory agent. *arXiv preprint arXiv:2507.02259*, 2025.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *Advances in Neural Information Processing Systems*, 38:113222–113244, 2026.

Xiangchi Yuan, Dachuan Shi, Chunhui Zhang, Zheyuan Liu, Shenglong Yao, Soroush Vosoughi, and Wenke Lee. Behavior knowledge merge in reinforced agentic models, 2026. URL https://arxiv.org/abs/2601.13572.

Yifu Yuan, Haiqin Cui, Yaoting Huang, Yibin Chen, Fei Ni, Zibin Dong, Pengyi Li, Yan Zheng, and Jianye Hao. Embodied-r1: Reinforced embodied reasoning for general robotic manipulation. *arXiv preprint arXiv:2508.13998*, 2025.

Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, et al. Glm-5: from vibe coding to agentic engineering. *arXiv preprint arXiv:2602.15763*, 2026.

Fangzhao Zhang and Mert Pilanci. Spectral adapter: Fine-tuning in spectral space. *arXiv preprint arXiv:2405.13952*, 2024.

Hanqing Zhu, Zhenyu Zhang, Wenyan Cong, Xi Liu, Sem Park, Vikas Chandra, Bo Long, David Z Pan, Zhangyang Wang, and Jinwon Lee. Apollo: Sgd-like memory, adamw-level performance. *Proceedings of Machine Learning and Systems*, 7, 2025a.

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

Zilin Zhu, Chengxing Xie, Xin Lv, and slime Contributors. slime: An llm post-training framework for rl scaling. https://github.com/THUDM/slime, 2025c. GitHub repository. Corresponding author: Xin Lv.

APPENDIX OUTLINE

The appendices are organized as follows.

- **Appendix A: Spectral-Distance Metrics, Empirical Evidence, and Calibration.** Defines the spectral-distance diagnostics, reports endpoint and training-time evidence, and calibrates spectral motion against an isotropic dimensional reference.
- **Appendix B: Qualitative Index-Alignment Check.** Provides rank-index singular-direction sanity checks for the spectrum-restoration intervention.
- **Appendix C: Fixed-Spectrum Theory and Frame-Adaptability Diagnostics.** Establishes the fixed-spectrum distance and nearest representatives, derives the dimension-aware calibration, formalizes the reconstruction classes and unexplained-update ratios, and reports rank sensitivity.
- **Appendix D: First-Order Properties of the ISO Parameterization.** Proves first-order spectrum preservation for feasible frame motion and derives the ISO factor gradients.
- **Appendix E: A Same-Base SFT–RLVR Case Study.** Compares SFT and RLVR checkpoints derived from the same 14B backbone and reports an additional spectral-interpolation intervention.
- **Appendix F: ISO-Merger Details.** Specifies sign canonicalization, tangent projection, mode masking, retention coefficients, retraction, parameter scope, and the complete merging algorithm.
- **Appendix G: Data-Free Merging Experimental Details and Additional Results.** Describes the experts, baselines, hyperparameters, evaluation protocols, additional experimental results and metrics, and ablation studies for ISO-Merger.
- **Appendix H: Numerical Precision of the SVD-Based Retraction.** Documents the SVD-based retraction, evaluates the accuracy–runtime trade-off of PyTorch SVD configurations, and motivates the FP64 GPU implementation.
- **Appendix I: Online RLVR Training Details.** Gives the shared setup and the mathematical-reasoning and competitive-coding RLVR configurations, optimizer settings, training budgets, and evaluation protocols.
- **Appendix J: Related Works.** Reviews related work on RLVR optimization, spectrum-preserving methods, and geometry-aware model merging, and clarifies ISO's distinctions from the closest approaches.

A Spectral-Distance Metrics, Empirical Evidence, and Calibration

Distance metrics and aggregation. For each analyzed matrix ℓ, let $\Delta W^{(\ell)} = W_1^{(\ell)} - W_0^{(\ell)}$. We evaluate the exact distance to the base fixed-spectrum family at two scales:

$$\delta_\Sigma^{(\ell)} := \frac{\|\sigma(W_1^{(\ell)}) - \sigma(W_0^{(\ell)})\|_2}{\|W_0^{(\ell)}\|_F}, \qquad \rho_\Sigma^{(\ell)} := \frac{\|\sigma(W_1^{(\ell)}) - \sigma(W_0^{(\ell)})\|_2}{\|\Delta W^{(\ell)}\|_F}. \tag{17}$$

Here δ_Σ measures proximity relative to the base-weight scale, whereas ρ_Σ measures the fraction of the full checkpoint displacement left unexplained by a closest fixed-spectrum representative. The ratios are defined for nonzero denominators. For a collection $\mathcal{L}$ of analyzed matrices, the pooled relative residual is

$$\rho_\Sigma^{\text{pool}} := \left[\frac{\sum_{\ell\in\mathcal{L}} \|\sigma(W_1^{(\ell)}) - \sigma(W_0^{(\ell)})\|_2^2}{\sum_{\ell\in\mathcal{L}} \|\Delta W^{(\ell)}\|_F^2}\right]^{1/2}. \tag{18}$$

This pooled quantity is distinct from the arithmetic mean of the per-matrix ratios.

A.1 Endpoint and Training-Time Evidence

Long-horizon endpoint. We examine whether spectral proximity persists after prolonged RLVR using the released endpoint of a reasoning run initialized from `DeepSeek-R1-Distill-Qwen-1.5B` (DS-1.5B) and trained for over 3,000 updates (Liu et al., 2025b; Hu et al., 2025). The checkpoint sequence starts from `Qwen2.5-Math-1.5B`, proceeds through an SFT transition to `DS-1.5B`, and then through RLVR to `Nemotron-Research-Reasoning-Qwen-1.5B`. Thus, the SFT endpoint serves as the base for the subsequent RLVR stage. These are sequential transitions rather than an SFT–RLVR comparison from a shared initialization. This case complements the broader evidence across models, datasets, RL objectives, and training horizons reported by (Zhu et al., 2025b).

Fig. 7 shows that the RLVR endpoint's spectrum nearly overlaps that of its incoming base, whereas the preceding SFT transition produces substantial spectral contraction. Across RLVR layers, $\delta_\Sigma^{(\ell)}$ is approximately $10^{-2}\%$, and $\rho_\Sigma^{\text{pool}} = 0.03$. Over the analyzed matrices, the pooled distance to the base fixed-spectrum families is therefore only 3% of the full base-to-RL displacement.

Sampled training checkpoints. Endpoint proximity alone cannot exclude an intermediate spectral excursion that subsequently cancels. We therefore analyze an independent unconstrained Qwen3-4B-Base AdamW run, using checkpoints saved every 10 actor updates through update 190. At each saved checkpoint, we measure proximity to the same initial fixed-spectrum family $\mathcal{F}(W_0)$.

Fig. 8 shows that both $\delta_\Sigma(t)$ and $\rho_\Sigma(t)$ remain small over the observed sequence. The mean base-normalized distance stays below 2×10^{-5}, while the reported mean relative residual is approximately 1.6%. At the resolution of the saved checkpoints, spectral stability therefore persists during training rather than appearing only at the final endpoint.

A.2 Dimension-Aware Calibration

Why dimensional calibration is needed. Small spectral drift does not by itself establish an optimization preference for spectrum-preserving motion. In the generic full-rank, simple-spectrum case, only $q = \min\{d_{\text{out}}, d_{\text{in}}\}$ independent first-order coordinates change the singular values within the $d_{\text{out}}d_{\text{in}}$-dimensional matrix space. At $W_0 = U_0\Sigma_0 V_0^\top$, these coordinates for a displacement ΔW are $\operatorname{diag}(U_0^\top \Delta W V_0)$. Consequently, even an isotropically oriented displacement places only a small fraction of its squared norm in spectrum-changing coordinates.

Normalized spectrum-changing energy. We account for this dimensional effect using

$$\kappa_{\text{spec}}^{(\ell)} := \frac{d_{\text{out}}d_{\text{in}}}{q} \frac{\left\|\operatorname{diag}\left(U_0^{(\ell)\top} \Delta W^{(\ell)} V_0^{(\ell)}\right)\right\|_2^2}{\|\Delta W^{(\ell)}\|_F^2}, \tag{19}$$

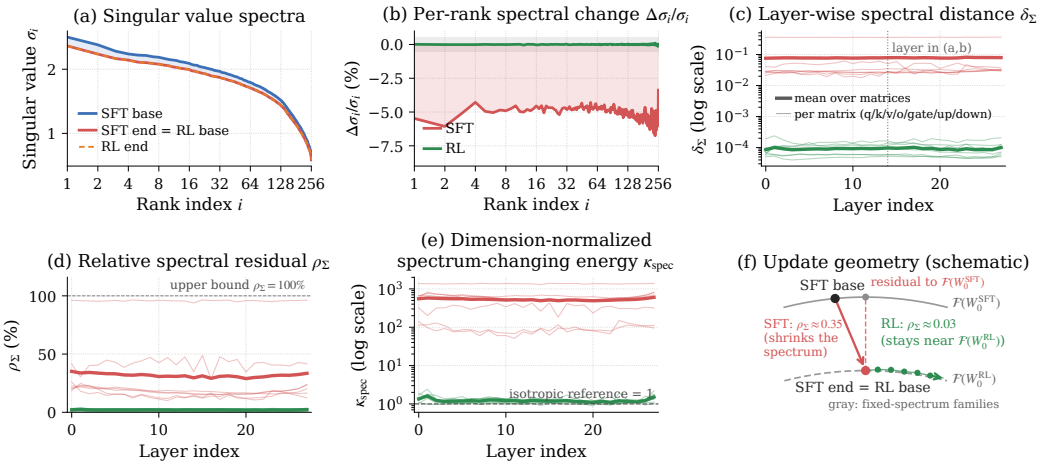


Figure 7: **Spectral dynamics under sequential SFT and RLVR. (a)** Representative singular-value profiles (`v_proj`, layer 14); the SFT endpoint is the RLVR base. **(b)** Rank-wise relative change: SFT reshapes the spectrum, whereas RLVR remains close to its incoming values. **(c)** Base-normalized spectral distance δ_Σ: the RLVR endpoint is two to three orders of magnitude closer to its base fixed-spectrum family than the illustrative SFT endpoint is to its own. **(d)** Mean relative spectral residuals are approximately 2% for RLVR and 32% for SFT. **(e)** Dimension-normalized spectrum-changing energy κ_{spec} (Equation 19): RLVR remains order-one, whereas SFT lies far above the isotropic reference. **(f)** Schematic endpoint geometry. The straight arrow denotes the SFT endpoint displacement, not a continuous optimization path.

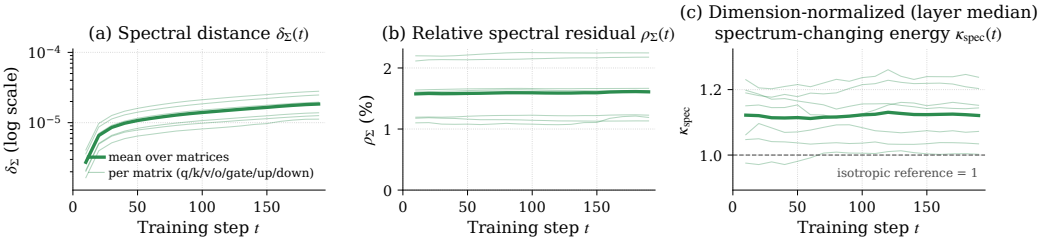


Figure 8: **Near-isospectrality at sampled training checkpoints.** AdamW on Qwen3-4B-Base, sampled every 10 actor updates through update 190. **(a)** The mean base-normalized spectral distance $\delta_\Sigma(t)$ stays below 2×10^{-5}. **(b)** The mean relative spectral residual $\bar{\rho}_\Sigma(t)$ is approximately 1.6%. **(c)** The dimension-normalized spectrum-changing energy $\kappa_{\text{spec}}(t)$ remains order-one relative to the isotropic reference.

where the dimensions and q refer to matrix ℓ. Under the full-rank, simple-spectrum assumptions, the matrices $\{u_{0,k} v_{0,k}^\top\}_{k=1}^q$ form an orthonormal basis for the first-order spectrum-changing subspace. An isotropic direction places expected squared energy in this subspace in proportion to its dimension:

$$\mathbb{E}\left[\frac{\|\operatorname{diag}(U_0^\top \Delta W V_0)\|_2^2}{\|\Delta W\|_F^2}\right] = \frac{q}{d_{\text{out}} d_{\text{in}}}. \tag{20}$$

Hence $\mathbb{E}[\kappa_{\text{spec}}] = 1$ under the isotropic reference. Values $\kappa_{\text{spec}} \ll 1$ indicate additional suppression of first-order spectrum-changing motion; order-one values do not reveal such a strong preference; and $\kappa_{\text{spec}} \gg 1$ indicates concentration in these coordinates.

Results and interpretation. Across matrix types in the public DS-1.5B reasoning run, RLVR yields mean κ_{spec} values between 1.02 and 1.35 (Fig. 7(e)). The same order-one pattern persists along the sampled Qwen3-4B-Base trajectory (Fig. 8(c)). This diagnostic therefore does not reveal strong additional suppression of spectrum-changing motion beyond the isotropic dimensional reference.

The illustrative SFT transition, by contrast, yields mean κ_{spec} values between 89 and 1364. Thus, correcting for ambient dimensionality preserves a pronounced distinction: the analyzed SFT displacement is far more concentrated in spectrum-changing coordinates than the RLVR displacement. These results distinguish the descriptive observation of near-isospectrality from an explanation based on preferential spectral suppression; neither determines the functional role of the remaining spectral changes.

B Qualitative Index-Alignment Check

Why this check is needed. The main text avoids using individual singular-vector identities as formal evidence, since SVD bases are gauge-dependent under sign flips and rotations inside near-degenerate singular-value clusters. Nevertheless, because our spectrum-rebasing intervention pairs singular values by rank index, it is useful to verify whether rank-index tracking is qualitatively stable in the RLVR runs we analyze.

Rank-index singular-vector rotations. For each layer, we compute the principal angle between rank-index paired singular directions from the base model and the post-training checkpoint, after standard sign alignment. These plots should be interpreted only as qualitative sanity checks. The formal frame-adaptability conclusions in Section 3 use projector-based reconstructions and unexplained-update ratios.

Fig. 9 and 10 provide a qualitative contrast between RLVR and SFT. Under RLVR, rank-index paired singular directions rotate mildly and smoothly, suggesting that the spectrum-restoration intervention is not dominated by large index-wise mismatch. Under SFT, many matched directions become nearly orthogonal, indicating that rank-index tracking is much less stable. We emphasize that these plots are not used as formal evidence for singular-subspace motion. The main diagnostic evidence is the gauge-invariant projector and reconstruction analysis in Section 3.

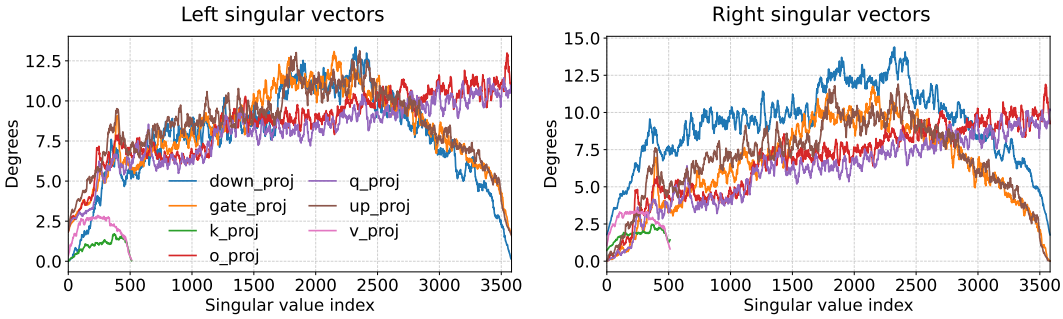


Figure 9: **Qualitative singular-direction rotations under RL post-training.** Mean principal angles between rank-index paired base and RL-trained singular directions remain small and smooth across attention and MLP layers. This suggests that rank-index tracking is qualitatively stable under RLVR in these runs. These plots are used only as a sanity check. Formal claims use projector-based diagnostics.

C Fixed-Spectrum Theory and Frame-Adaptability Diagnostics

Notation. Let $W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$ and $q = \min\{d_{\text{out}}, d_{\text{in}}\}$. We write

$$\sigma(W) = \big(\sigma_1(W), \ldots, \sigma_q(W)\big)$$

for all singular values in nonincreasing order, including zeros with multiplicity. A thin SVD is

$$W = U\Sigma V^\top, \qquad U \in \mathrm{St}(d_{\text{out}}, q), \quad V \in \mathrm{St}(d_{\text{in}}, q), \quad \Sigma = \mathrm{Diag}(\sigma(W)).$$

Here $\mathrm{Diag}(a)$ constructs a diagonal matrix from a vector, whereas $\mathrm{diag}(A)$ extracts the diagonal of a matrix.

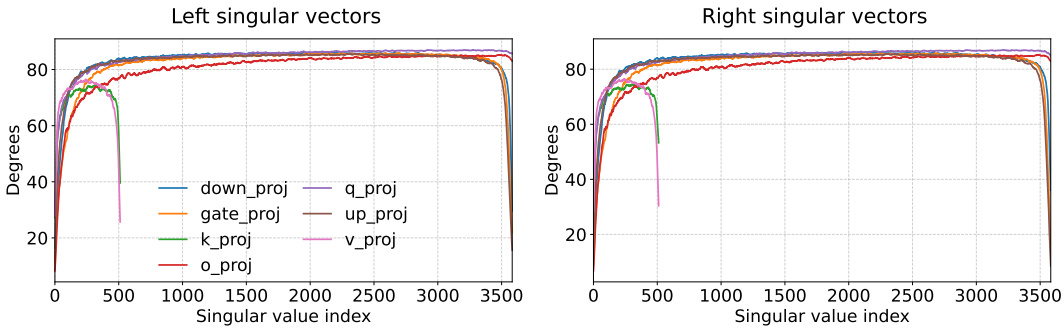


Figure 10: **Qualitative singular-direction rotations under SFT.** Mean angles between rank-index paired base and SFT singular directions increase toward near-orthogonality for many ranks. In high-dimensional spaces, unrelated directions are typically nearly orthogonal, so this indicates that simple rank-index tracking becomes much less informative under SFT.

C.1 The Fixed-Spectrum Family and Nearest Representatives

Proposition C.1 (Two-sided characterization of the fixed-spectrum family)**.** *For any source matrix* W_0*,*

$$\begin{aligned}\mathcal{F}(W_0) &:= \left\{Z \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}} : \sigma(Z) = \sigma(W_0)\right\} \\ &= \left\{Q_L W_0 Q_R^\top : Q_L \in \mathrm{O}(d_{\text{out}}),\ Q_R \in \mathrm{O}(d_{\text{in}})\right\}.\end{aligned} \tag{21}$$

Proof. Orthogonal multiplication preserves singular values, proving one inclusion. For the reverse inclusion, write

$$W_0 = U_0 \Sigma_0 V_0^\top, \qquad Z = U_Z \Sigma_0 V_Z^\top.$$

Extend U_0, U_Z and V_0, V_Z to complete orthogonal bases and choose Q_L, Q_R mapping the source bases to the target bases. Then $Q_L W_0 Q_R^\top = Z$. □

Proposition C.2 (Exact distance to the fixed-spectrum family)**.** *For* $W, W_0 \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$*,*

$$\operatorname{dist}_F\big(W, \mathcal{F}(W_0)\big) = \|\sigma(W) - \sigma(W_0)\|_2. \tag{22}$$

If $W = U\Sigma V^\top$ *and* $\Sigma_0 = \operatorname{Diag}(\sigma(W_0))$*, then*

$$U\Sigma_0 V^\top \in \operatorname*{arg\,min}_{Z \in \mathcal{F}(W_0)} \|W - Z\|_F. \tag{23}$$

Proof. Every $Z \in \mathcal{F}(W_0)$ has Frobenius norm $\|W_0\|_F$, so

$$\|W - Z\|_F^2 = \|W\|_F^2 + \|W_0\|_F^2 - 2\langle W, Z\rangle_F.$$

By von Neumann's trace inequality,

$$\langle W, Z\rangle_F \le \sum_{k=1}^{q} \sigma_k(W)\sigma_k(W_0).$$

The bound is attained by $Z = U\Sigma_0 V^\top$. Substitution gives

$$\min_{Z \in \mathcal{F}(W_0)} \|W - Z\|_F^2 = \sum_{k=1}^{q} \big(\sigma_k(W) - \sigma_k(W_0)\big)^2.$$

□

Remark C.3 (Gauge and nonuniqueness of a rebased representative). The distance to the fixed-spectrum family and the set of nearest points are basis-independent, but a particular representative $U\Sigma_0 V^\top$ need not be unique.

Suppose a block $\mathcal{C}$ of the target spectrum is exactly repeated. Replacing

$$U_{\mathcal{C}} \leftarrow U_{\mathcal{C}} R, \qquad V_{\mathcal{C}} \leftarrow V_{\mathcal{C}} R, \qquad R \in \mathrm{O}(|\mathcal{C}|)$$

leaves W unchanged but produces the rebased block

$$U_{\mathcal{C}} R \Sigma_{0,\mathcal{C}} R^\top V_{\mathcal{C}}^\top .$$

Every such representative is a nearest point in $\mathcal{F}(W_0)$. It is identical for all R when $\Sigma_{0,\mathcal{C}}$ is proportional to the identity. More generally, with $\bar{\sigma}_{\mathcal{C}}$ denoting the block mean,

$$\left\| R\Sigma_{0,\mathcal{C}} R^\top - \Sigma_{0,\mathcal{C}} \right\|_F \le 2 \left\| \Sigma_{0,\mathcal{C}} - \bar{\sigma}_{\mathcal{C}} I \right\|_F . \tag{24}$$

Thus the ambiguity is small when the source spectrum is nearly flat inside the cluster. Near-degenerate but distinct singular values do not create exact gauge freedom, but can make the numerical singular vectors ill-conditioned. Our rebasing experiment evaluates the representative selected by the numerical SVD; Appendix B reports a qualitative stability check.

Corollary C.4 (Finite-horizon spectral drift)**.** *Let $W_0, \dots, W_T$ be a matrix trajectory and $E_t = W_{t+1} - W_t$. Then*

$$\|\sigma(W_t) - \sigma(W_0)\|_2 = \operatorname{dist}_F\big(W_t, \mathcal{F}(W_0)\big) \le \|W_t - W_0\|_F \le \sum_{\tau=0}^{t-1} \|E_\tau\|_F . \tag{25}$$

C.2 Spectral-Distance Metrics and Dimension-Aware Calibration

For matrix ℓ, let $\Delta W^{(\ell)} = W_1^{(\ell)} - W_0^{(\ell)}$. We use

$$\begin{aligned} \delta_\Sigma^{(\ell)} &= \frac{\operatorname{dist}_F\left(W_1^{(\ell)}, \mathcal{F}(W_0^{(\ell)})\right)}{\|W_0^{(\ell)}\|_F}, \\ \rho_\Sigma^{(\ell)} &= \frac{\operatorname{dist}_F\left(W_1^{(\ell)}, \mathcal{F}(W_0^{(\ell)})\right)}{\|\Delta W^{(\ell)}\|_F}. \end{aligned} \tag{26}$$

The pooled residual reported in the main text is

$$\rho_\Sigma^{\text{pool}} = \left[\frac{\sum_\ell \operatorname{dist}_F^2\left(W_1^{(\ell)}, \mathcal{F}(W_0^{(\ell)})\right)}{\sum_\ell \|\Delta W^{(\ell)}\|_F^2} \right]^{1/2} . \tag{27}$$

Proposition C.5 (First-order spectrum-changing subspace and isotropic calibration)**.** *Assume W_0 is full rank with simple singular values and write $W_0 = U_0 \Sigma_0 V_0^\top$. The first-order spectrum-changing subspace at W_0, equivalently the normal space of its fixed-spectrum family, is*

$$N_{W_0}\mathcal{F}(W_0) = \left\{ U_0 \operatorname{Diag}(a) V_0^\top : a \in \mathbb{R}^q \right\}. \tag{28}$$

The orthogonal projection of a perturbation H onto this spectrum-changing subspace is

$$\Pi_{\text{spec}}(H) = U_0 \operatorname{Diag}\left(\operatorname{diag}(U_0^\top H V_0)\right) V_0^\top . \tag{29}$$

If $\operatorname{vec}(H)/\|H\|_F$ is isotropic, then

$$\mathbb{E}\left[\frac{\|\Pi_{\text{spec}}(H)\|_F^2}{\|H\|_F^2} \right] = \frac{q}{d_{\text{out}} d_{\text{in}}}. \tag{30}$$

Consequently,

$$\kappa_{\text{spec}} = \frac{d_{\text{out}} d_{\text{in}}}{q} \frac{\|\operatorname{diag}(U_0^\top H V_0)\|_2^2}{\|H\|_F^2} \tag{31}$$

satisfies $\mathbb{E}[\kappa_{\text{spec}}] = 1$. Thus, one is the isotropic dimensional reference under this normalization.

Proof. For a simple full-rank spectrum, fixing the q singular values imposes q independent first-order constraints. The matrices $\{u_{0,k} v_{0,k}^\top\}_{k=1}^q$ form an orthonormal basis for this spectrum-changing subspace, which gives Equation 29. An isotropic direction places expected squared energy in a fixed subspace in proportion to its dimension. The spectrum-changing and ambient dimensions are q and $d_{\text{out}} d_{\text{in}}$, respectively. □

C.3 Reconstruction Classes and Unexplained-Update Ratios

This subsection gives the full reconstruction test used in Section 3: its structural restrictions, optimal reconstructions, normalized residuals, and interpretation. The sequential-RL experiments and rank-sensitivity results are reported in Appendix C.4.

Incoming checkpoint and admissible truncations. For a transition $W_j \to W_i$, let W_j be the *incoming checkpoint* and W_i the *learned endpoint*. For each corresponding weight matrix, let $q = \min\{d_{\text{out}}, d_{\text{in}}\}$. At an integer rank $1 \le r < q$, write

$$\begin{aligned} W_t^{(r)} &= U_t^{(r)} \Sigma_t^{(r)} \big(V_t^{(r)}\big)^\top, \\ P_t^{(r)} &= U_t^{(r)} \big(U_t^{(r)}\big)^\top, \qquad Q_t^{(r)} = V_t^{(r)} \big(V_t^{(r)}\big)^\top, \qquad t \in \{i, j\}. \end{aligned} \tag{32}$$

Here $U_t^{(r)}$ and $V_t^{(r)}$ contain the leading r left and right singular vectors, and $P_t^{(r)}$ and $Q_t^{(r)}$ project onto the corresponding output and input subspaces. A transition is *admissible* at rank r when

$$\sigma_r(W_t) > \sigma_{r+1}(W_t), \qquad t \in \{i, j\}, \tag{33}$$

so that both retained subspaces are uniquely defined. The main-text comparison uses $r = \lfloor 0.9q \rfloor$; Appendix C.4 varies the truncation fraction.

Four structural restrictions. Let $A = W_i^{(r)}$. We compare three subspace-retaining classes,

$$\begin{aligned} \mathcal{C}_{\text{mix}} &= \left\{ U_j^{(r)} C \big(V_j^{(r)}\big)^\top : C \in \mathbb{R}^{r \times r} \right\}, \\ \mathcal{C}_L &= \left\{ U_j^{(r)} B : B \in \mathbb{R}^{r \times d_{\text{in}}} \right\}, \\ \mathcal{C}_R &= \left\{ B \big(V_j^{(r)}\big)^\top : B \in \mathbb{R}^{d_{\text{out}} \times r} \right\}, \end{aligned} \tag{34}$$

with the incoming rank-r fixed-spectrum family,

$$\mathcal{F}_j^{(r)} := \left\{ U \Sigma_j^{(r)} V^\top : U \in \text{St}(d_{\text{out}}, r),\ V \in \text{St}(d_{\text{in}}, r) \right\}. \tag{35}$$

The classes $\mathcal{C}_{\text{mix}}$, $\mathcal{C}_L$, and $\mathcal{C}_R$ retain both incoming subspaces, only the output subspace, or only the input subspace, respectively. Their remaining mapping and spectrum may refit freely. By contrast, $\mathcal{F}_j^{(r)}$ retains the incoming singular values $\Sigma_j^{(r)}$ while allowing both frames to vary.

Proposition C.6 (Optimal checkpoint reconstructions)**.** *For an admissible transition at rank r, the Frobenius-optimal reconstructions of $A = W_i^{(r)}$ under the four restrictions are*

$$\begin{aligned} \widehat{W}_{\text{mix}} &= P_j^{(r)} A Q_j^{(r)}, \quad \widehat{W}_L = P_j^{(r)} A, \\ \widehat{W}_R &= A Q_j^{(r)}, \qquad \widehat{W}_{\text{iso}} = U_i^{(r)} \Sigma_j^{(r)} \big(V_i^{(r)}\big)^\top. \end{aligned} \tag{36}$$

In particular,

$$\begin{aligned} \widehat{W}_{\text{iso}} &\in \arg\min_{Z \in \mathcal{F}_j^{(r)}} \|A - Z\|_F, \\ \min_{Z \in \mathcal{F}_j^{(r)}} \|A - Z\|_F &= \|\Sigma_i^{(r)} - \Sigma_j^{(r)}\|_F. \end{aligned} \tag{37}$$

Proof. The maps $A \mapsto P_j^{(r)} A Q_j^{(r)}$, $A \mapsto P_j^{(r)} A$, and $A \mapsto A Q_j^{(r)}$ are orthogonal projections, under the Frobenius inner product, onto $\mathcal{C}_{\text{mix}}$, $\mathcal{C}_L$, and $\mathcal{C}_R$, respectively. They therefore minimize the reconstruction error over these classes. The isospectral result follows from Proposition C.2, applied to $W_i^{(r)}$ and $W_j^{(r)}$: their remaining $q - r$ singular values are zero, so the distance is $\|\Sigma_i^{(r)} - \Sigma_j^{(r)}\|_F$, attained by $U_i^{(r)} \Sigma_j^{(r)} (V_i^{(r)})^\top$. □

The remix reconstruction can equivalently be written as

$$\begin{aligned} \widehat{W}_{\text{mix}} &= U_j^{(r)} B_{ij}^{(r)} \big(V_j^{(r)}\big)^\top, \\ B_{ij}^{(r)} &:= \big(U_j^{(r)}\big)^\top W_i^{(r)} V_j^{(r)}. \end{aligned} \tag{38}$$

The core $B_{ij}^{(r)}$ is unconstrained: it may rotate, mix, rescale, and change the represented spectrum. Thus, this class fixes the incoming spans, not the individual singular directions or mode scales.

Normalized residuals and unexplained-update ratios. We normalize each reconstruction error by the endpoint scale:

$$e_h := \frac{\|A - \widehat{W}_h\|_F}{\|A\|_F}, \qquad h \in \{\mathrm{mix}, L, R, \mathrm{iso}\}. \tag{39}$$

The unchanged incoming checkpoint provides the reference

$$e_{\mathrm{drift}} := \frac{\|W_i^{(r)} - W_j^{(r)}\|_F}{\|W_i^{(r)}\|_F}. \tag{40}$$

This is a scale reference, not an additional reconstruction class. For $W_i^{(r)} \neq W_j^{(r)}$, the corresponding *unexplained-update ratio* is

$$u_h := \frac{e_h}{e_{\mathrm{drift}}} = \frac{\|W_i^{(r)} - \widehat{W}_h\|_F}{\|W_i^{(r)} - W_j^{(r)}\|_F}, \qquad h \in \{\mathrm{mix}, L, R, \mathrm{iso}\}. \tag{41}$$

Because $W_j^{(r)}$ belongs to all four reconstruction classes, optimality gives

$$0 \le \|W_i^{(r)} - \widehat{W}_h\|_F \le \|W_i^{(r)} - W_j^{(r)}\|_F, \qquad 0 \le u_h \le 1.$$

Thus, $u_h = 0$ denotes exact reconstruction, whereas $u_h = 1$ means that the best reconstruction under restriction h is no closer to the endpoint than the unchanged incoming checkpoint. The quantities u_h are undefined for zero truncated displacement. The fixed-spectrum residuals have the closed forms

$$e_{\mathrm{iso}} = \frac{\|\Sigma_i^{(r)} - \Sigma_j^{(r)}\|_F}{\|\Sigma_i^{(r)}\|_F}, \tag{42}$$

$$u_{\mathrm{iso}} = \frac{\|\Sigma_i^{(r)} - \Sigma_j^{(r)}\|_F}{\|W_i^{(r)} - W_j^{(r)}\|_F}. \tag{43}$$

Remark C.7 (Interpretation of the reconstruction test). The classes have different dimensions, so their residual ordering is not a complexity-normalized model-selection result. The test concerns parameterization sufficiency: whether retaining one or both incoming spans still permits a low-residual description of the observed rank-r endpoint. Each subspace-retaining class contains its corresponding fixed-spectrum restriction, making its optimal residual an optimistic lower bound for that more restrictive description. Thus, a large residual cannot be attributed to unnecessarily fixing the remaining core or spectrum.

Remark C.8 (Gauge invariance and internal rotations). For fixed checkpoints, the retained projectors and reconstruction classes do not depend on the choice of bases for the retained subspaces, and the optimal residual values are basis-independent. If $\Sigma_i^{(r)}$ contains repeated singular values, $\widehat{W}_{\mathrm{iso}}$ may not be the unique nearest representative, but e_{iso} and u_{iso} remain uniquely determined whenever defined.

The remix class contains the internal fixed-spectrum slice

$$\left\{ U_j^{(r)} R_U \Sigma_j^{(r)} R_V^\top \big(V_j^{(r)}\big)^\top : R_U, R_V \in \mathrm{O}(r) \right\},$$

where $\mathrm{O}(r)$ is the orthogonal group. Consequently, a large optimal remix residual also excludes a low-residual explanation based solely on rotations within both incoming subspaces.

Remark C.9 (Why $r < q$). At full thin rank, at least one of $P_j^{(q)}$ and $Q_j^{(q)}$ is the identity. The corresponding one-sided reconstruction is then exact, and the remix reconstruction coincides with the other one-sided reconstruction. For square matrices, both projectors are the identity and all three subspace-retaining reconstructions are exact. Using $r < q$ makes both retained subspaces proper and keeps the comparison informative.

C.4 Sequential RL Evidence and Rank Sensitivity

We apply the reconstruction test to address two questions: whether simpler, subspace-retaining transformations explain learned endpoints, and whether the same structural pattern recurs after changing the RL objective.

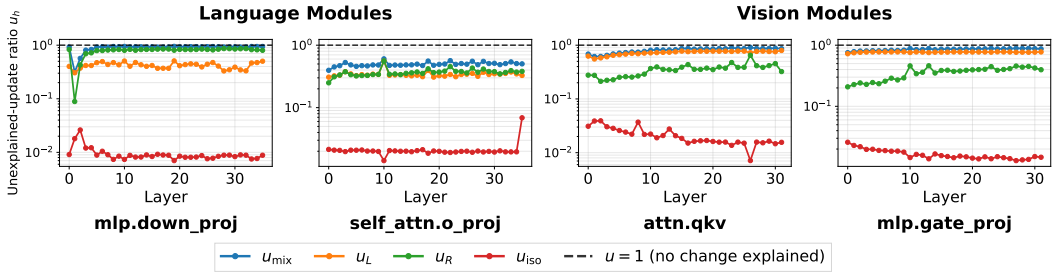


Figure 11: **Both incoming frames must remain adaptable for low-residual endpoint reconstruction.** Unexplained-update ratios u_h for $W_0 \to W_2$ at $r = \lfloor 0.9q \rfloor$, across layers of representative language and vision modules. The dashed line $u_h = 1$ marks the error of the unchanged incoming checkpoint. Retaining one or both incoming subspaces (u_{mix}, u_L, u_R) leaves substantially larger residuals than retaining the incoming spectrum while adapting both frames (u_{iso}). Fig. 12 extends the comparison across transitions and truncation ranks.

A sequential objective-shift stress test. We analyze a two-stage vision-language-action RL pipeline initialized from `Qwen2.5-VL-3B-Instruct` (Yuan et al., 2025). Stage I optimizes spatial reasoning and produces W_1 from W_0. Stage II starts from this already RL-trained checkpoint, optimizes an embodied-manipulation objective, and produces W_2. We examine corresponding weight matrices in both language and vision modules across three transitions:

$$W_0 \to W_1, \qquad W_1 \to W_2, \qquad W_0 \to W_2. \tag{44}$$

The first transition provides the initial-stage reference. The second tests recurrence after both an objective shift and re-anchoring at an RL-trained checkpoint: its incoming spectrum and subspaces are those of W_1, not W_0. The cumulative transition measures the complete two-stage change relative to W_0. Unless otherwise stated, we use $r = \lfloor 0.9q \rfloor$ and the admissibility condition in Equation 33.

Endpoint reconstruction: both frames must remain adaptable. For the cumulative transition $W_0 \to W_2$, Fig. 11 reports the unexplained-update ratios across layers of representative language and vision modules. The reported median ratios are 87% for remixing within both incoming subspaces, 45% for retaining only the output subspace, and 42% for retaining only the input subspace. These residuals are already optimistic because the corresponding classes freely refit the remaining mapping and spectrum. By contrast, retaining the incoming spectrum while adapting both frames leaves a median residual of only 1.8%. Thus, among the tested reconstruction classes, both singular frames must remain adaptable for a low-residual characterization of the observed checkpoint changes.

Recurrence under a new RL objective. The same qualitative separation holds for $W_1 \to W_2$, where the incoming checkpoint has already undergone RL and the objective changes from spatial reasoning to embodied manipulation. Thus, the two-frame reconstruction pattern is not confined to the first RL stage: it recurs after re-anchoring at W_1 and changing the objective. Fig. 12 includes this transition, the first-stage transition $W_0 \to W_1$, and the cumulative transition $W_0 \to W_2$.

Rank-sensitivity protocol. For $\alpha \in \{0.2, 0.4, 0.6, 0.8, 0.9\}$, we use

$$r_\alpha = \max\{1, \lfloor \alpha q \rfloor\}. \tag{45}$$

At each α, a matrix–transition pair is included only when $r_\alpha < q$ is admissible at both endpoints; inadmissible pairs are omitted at that value of α. For each included pair with nonzero truncated displacement, we compute $u_h = e_h / e_{\text{drift}}$ as in Equation 41.

Rank-sensitivity results. The separation between inherited-spectrum and subspace-retaining reconstructions persists across the displayed ranks in Fig. 12. For the representative matrix–transition panels shown, u_{iso} remains below 2% throughout the sweep, whereas the subspace-retaining alternatives leave tens of percent of the truncated update unexplained. At $\alpha = 0.2$, the reported median across the twelve displayed panels is $u_{\text{mix}} = 0.99$. The gap is therefore already pronounced at small truncation ranks: remixing within the dominant incoming spans explains little of the observed rank-r change, whereas adapting both frames under the inherited spectrum remains low-residual.

Together, the endpoint, objective-shift, and rank-sensitivity results support the structural conclusion of Section 3: the incoming spectrum can be retained while both frames remain adaptable for a

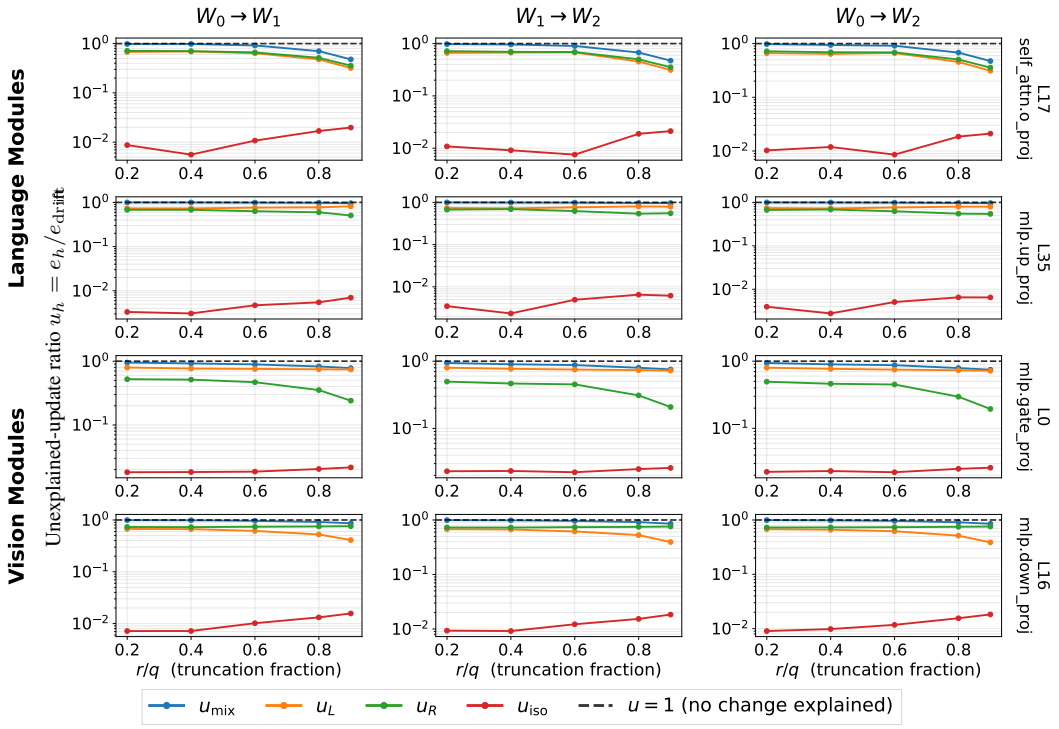


Figure 12: **Rank sensitivity across sequential RL transitions.** Unexplained-update ratios versus r/q for representative language and vision matrices and all three transitions. The dashed line $u_h = 1$ marks the unchanged-checkpoint reference. Across the displayed sweep, inherited-spectrum reconstruction remains below 2%, while the subspace-retaining alternatives leave substantially larger residuals. The reported median u_{mix} at $\alpha = 0.2$ is 0.99 across the twelve displayed matrix–transition panels.

low-residual description of the observed changes. Combined with the functional interventions in Section 2.2, this motivates ISO's parameterization $W(U, V) = U\Sigma_0 V^\top$ as an inductive bias for learning and composition.

D First-Order Properties of the ISO Parameterization

D.1 Proof of Proposition 4.1

Proof. We prove the two claims in Proposition 4.1.

First, consider a differentiable curve $W(\varepsilon) \in \mathcal{F}(W_0)$ with $W(0) = W = U\Sigma_0 V^\top$. Since all q singular values are positive and simple, we may locally choose differentiable Stiefel representatives $U(\varepsilon)$ and $V(\varepsilon)$ such that

$$W(\varepsilon) = U(\varepsilon)\Sigma_0 V(\varepsilon)^\top, \qquad U(0) = U, \quad V(0) = V. \tag{46}$$

Let

$$\dot{U} = \frac{d}{d\varepsilon} U(\varepsilon)\Big|_{\varepsilon=0}, \qquad \dot{V} = \frac{d}{d\varepsilon} V(\varepsilon)\Big|_{\varepsilon=0}, \qquad \dot{W} = \frac{d}{d\varepsilon} W(\varepsilon)\Big|_{\varepsilon=0}. \tag{47}$$

Differentiating the fixed-spectrum representation gives

$$\dot{W} = \dot{U}\Sigma_0 V^\top + U\Sigma_0 \dot{V}^\top. \tag{48}$$

Multiplying on the left by $U^\top$ and on the right by V yields

$$U^\top \dot{W} V = U^\top \dot{U}\,\Sigma_0 + \Sigma_0\,\dot{V}^\top V. \tag{49}$$

Because $U(\varepsilon)$ and $V(\varepsilon)$ remain on the Stiefel manifold, differentiating $U(\varepsilon)^\top U(\varepsilon) = I$ and $V(\varepsilon)^\top V(\varepsilon) = I$ at $\varepsilon = 0$ gives

$$U^\top \dot{U} + \dot{U}^\top U = 0, \qquad V^\top \dot{V} + \dot{V}^\top V = 0. \tag{50}$$

Thus $U^\top \dot{U}$ and $\dot{V}^\top V$ are skew-symmetric. Since Σ_0 is diagonal, both $U^\top \dot{U}\,\Sigma_0$ and $\Sigma_0 \dot{V}^\top V$ have zero diagonal. Therefore

$$\operatorname{diag}(U^\top \dot{W} V) = 0. \tag{51}$$

Second, let H be an arbitrary perturbation and define

$$W(\varepsilon) = W + \varepsilon H. \tag{52}$$

For each singular value, choose differentiable singular vectors $u_k(\varepsilon)$ and $v_k(\varepsilon)$ with singular value $\sigma_k(\varepsilon)$. Then

$$\sigma_k(\varepsilon) = u_k(\varepsilon)^\top W(\varepsilon) v_k(\varepsilon). \tag{53}$$

Differentiating at $\varepsilon = 0$ gives

$$\sigma_k'(0) = \dot{u}_k^\top W v_k + u_k^\top H v_k + u_k^\top W \dot{v}_k. \tag{54}$$

Using $W v_k = \sigma_k u_k$ and $u_k^\top W = \sigma_k v_k^\top$, this becomes

$$\sigma_k'(0) = \sigma_k \dot{u}_k^\top u_k + u_k^\top H v_k + \sigma_k v_k^\top \dot{v}_k. \tag{55}$$

Since $u_k(\varepsilon)$ and $v_k(\varepsilon)$ remain unit vectors, differentiating $u_k(\varepsilon)^\top u_k(\varepsilon) = 1$ and $v_k(\varepsilon)^\top v_k(\varepsilon) = 1$ gives

$$\dot{u}_k^\top u_k = 0, \qquad v_k^\top \dot{v}_k = 0. \tag{56}$$

Hence

$$\frac{d}{d\varepsilon} \sigma_k(W + \varepsilon H)\Big|_{\varepsilon=0} = u_k^\top H v_k. \tag{57}$$

Collecting this identity for $k = 1, \ldots, q$ shows that $\operatorname{diag}(U^\top H V)$ is exactly the vector of first-order singular-value changes. This proves the proposition. □

D.2 Factor-Gradient Derivation

Let $\mathcal{L}(W)$ be the RLVR loss, let $G_W = \nabla_W \mathcal{L}(W)$, and use the fixed-spectrum parameterization

$$W(U, V) = U \Sigma_0 V^\top \in \mathcal{F}(W_0). \tag{58}$$

For infinitesimal Stiefel-factor changes $\xi_U \in T_U \mathrm{St}(d_{\mathrm{out}}, q)$ and $\xi_V \in T_V \mathrm{St}(d_{\mathrm{in}}, q)$, the induced weight motion is

$$\dot{W} = \xi_U \Sigma_0 V^\top + U \Sigma_0 \xi_V^\top. \tag{59}$$

The first-order change in loss is

$$\begin{aligned} d\mathcal{L}[\dot{W}] &= \langle G_W, \dot{W} \rangle_F \\ &= \langle G_W V \Sigma_0, \xi_U \rangle_F + \langle G_W^\top U \Sigma_0, \xi_V \rangle_F. \end{aligned} \tag{60}$$

Therefore the raw factor gradients used by ISO are

$$G_U = G_W V \Sigma_0, \qquad G_V = G_W^\top U \Sigma_0. \tag{61}$$

These are the Euclidean gradients in the ambient factor coordinates. ISO does not explicitly construct a weight-space projected gradient. Instead, the fixed-spectrum parameterization and polar retraction restrict every represented iterate to $\mathcal{F}(W_0)$ up to numerical precision.

E A Same-Base SFT–RLVR Case Study

The main-text case study in Section 2.1 compares an SFT transition and an RLVR transition initialized from different base checkpoints. To control for this difference, we repeat the analysis using two public 14B checkpoints derived from the same Qwen2.5-14B base:

$$\texttt{Qwen2.5-14B} \longrightarrow \begin{cases} \texttt{DeepSeek-R1-Distill-Qwen-14B} & \text{(distilled SFT)}, \\ \texttt{Qwen-2.5-14B-SimpleRL-Zoo} & \text{(RLVR)}. \end{cases}$$

Both endpoints are measured relative to the shared base checkpoint. We restrict the comparison to the seven transformer projection matrices whose parameter names and shapes match the shared Qwen2.5-14B backbone, avoiding differences introduced by checkpoint-specific tokenizer or configuration changes. This comparison therefore controls for the starting backbone weights, while the training data, objectives, and optimization procedures remain intentionally different. Fig. 13 summarizes the results.

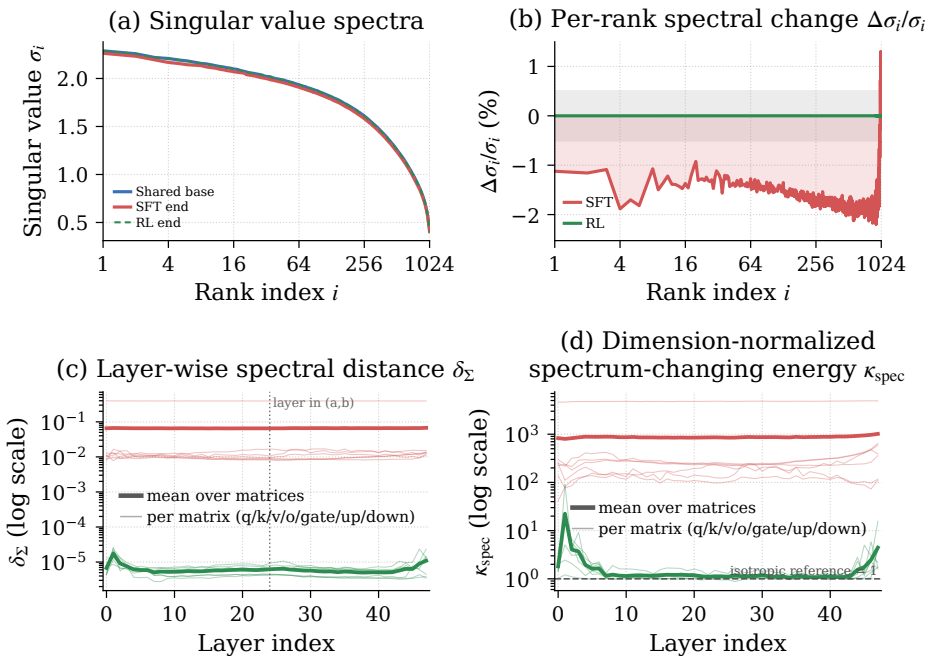


Figure 13: **Same-base spectral dynamics.** The distilled-SFT and RLVR endpoints are both compared with their shared `Qwen2.5-14B` base. **(a)** Singular-value profiles for `v_proj` in layer 24. **(b)** Rank-wise relative spectral change: the RLVR curve remains near zero at the plotted scale, whereas distilled SFT systematically reshapes the spectrum. **(c)** Layer-wise base-normalized spectral distance δ_Σ, shown per projection matrix (thin) and averaged across projection types (thick); the RLVR endpoint remains several orders of magnitude closer to the fixed-spectrum family of the shared base. **(d)** Dimension-normalized spectrum-changing energy κ_{spec}: RLVR remains order-one relative to the isotropic dimensional reference, whereas distilled SFT lies roughly two to three orders of magnitude above it.

Raw spectral proximity. The two endpoints differ sharply in their spectral displacement from the shared base. At the scale of Fig. 13b, the per-rank changes of the RLVR endpoint are visually near zero. Its base-normalized spectral distance is approximately

$$\delta_\Sigma \approx 6 \times 10^{-6}$$

across layers and projection types. By contrast, the distilled-SFT endpoint substantially changes the spectrum: δ_Σ is approximately 10^{-2} for most projection types and reaches approximately 0.4 for the attention output projection.

Dimension-aware calibration. Raw proximity alone does not establish a preferentially spectrum-preserving update direction, so we additionally report the dimension-normalized spectrum-changing energy κ_{spec}. Across projection types, RLVR yields mean values between 1.4 and 3.2, with an overall mean of 1.9. These values remain order-one relative to the isotropic dimensional reference, with the largest deviations concentrated in the earliest layers.

The distilled-SFT endpoint yields mean κ_{spec} values between 117 and 4810, with an overall mean of 873 and the largest concentration in the attention output projection. The same-base comparison therefore reproduces the main-text contrast: SFT concentrates its displacement in spectrum-changing coordinates by roughly two to three orders of magnitude, whereas RLVR exhibits no comparable concentration. This difference cannot be explained by the two transitions starting from different backbone checkpoints.

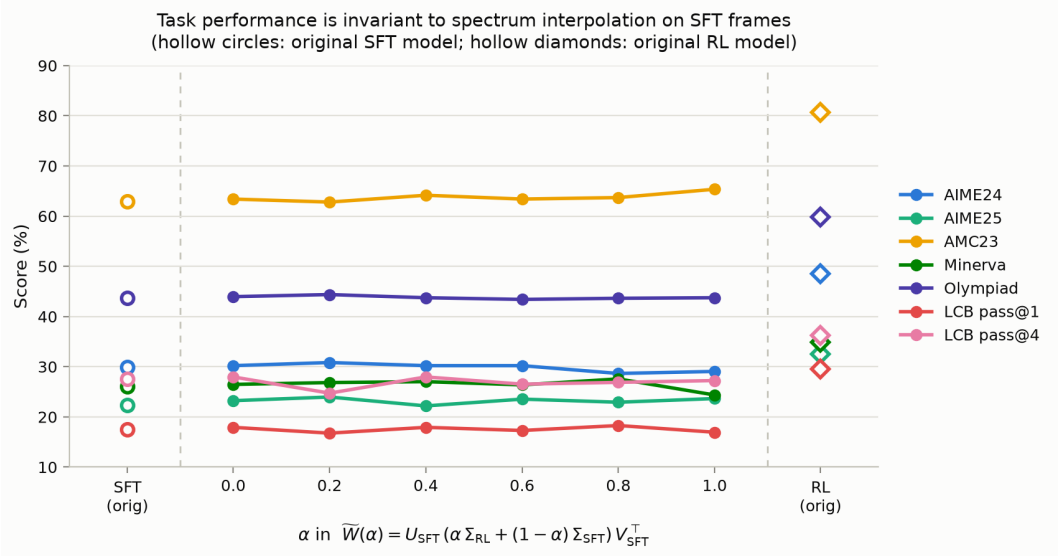


Figure 14: **Task performance is insensitive to spectral interpolation in the SFT singular frames.** We hold the SFT left and right singular frames fixed and interpolate only the spectrum, $\widetilde{W}(\alpha) = U_{\mathrm{SFT}}[(1-\alpha)\Sigma_{\mathrm{SFT}} + \alpha\Sigma_{\mathrm{RL}}]V_{\mathrm{SFT}}^{\top}$, for $\alpha \in [0,1]$. Performance remains nearly unchanged as an arbitrary proportion of the RL spectrum is substituted into the SFT frames. Hollow circles and diamonds show the original SFT and RL checkpoints, respectively. This complements Fig. 2(a), which performs the converse intervention by restoring the SFT/base spectrum in the RL-trained frames.

F ISO-Merger Details

This appendix expands the construction in Section 4.1, including the shared frame coordinates, sign alignment, tangent projection, mode masking, retention coefficients, and feasibility restoration. Algorithm 1 gives the complete procedure.

F.1 Setup and Shared Coordinates

ISO-Merger takes a shared base checkpoint θ_0 and $K \geq 1$ RL expert checkpoints $\{\theta_i\}_{i=1}^{K}$ with corresponding parameter names and shapes. We study specialists trained from the same base for distinct domains or capabilities, rather than already-generalist policies trained on broad, substantially overlapping mixtures. Composition uses only these checkpoints, with no post-merge data, rollouts, gradient updates, or distillation.

For a corresponding 2D weight matrix, let $q = \min\{d_{\mathrm{out}}, d_{\mathrm{in}}\}$ and compute the thin SVDs

$$W_0 = U_0\Sigma_0 V_0^{\top}, \qquad W_i = U_i\Sigma_i V_i^{\top}, \qquad i = 1,\dots,K. \tag{62}$$

All singular values are ordered nonincreasingly, with $U_i \in \mathrm{St}(d_{\mathrm{out}}, q)$ and $V_i \in \mathrm{St}(d_{\mathrm{in}}, q)$, including $i = 0$. ISO-Merger reuses the base spectrum Σ_0 and represents expert i through its learned frames under that spectrum:

$$U_i\Sigma_0 V_i^{\top} \in \mathcal{F}(W_0). \tag{63}$$

The expert spectra Σ_i need not equal Σ_0; the input experts need not have been trained with ISO. Rather than combining Euclidean task vectors, we compose frame displacements anchored at the same base factors (U_0, V_0). All subsequent quantities and coefficients are computed separately for each 2D weight matrix; matrix indices are omitted for readability.

F.2 Sign Canonicalization

For a positive, simple singular value, the corresponding singular-vector pair is determined only up to a joint sign: (u_k, v_k) and $(-u_k, -v_k)$ represent the same matrix. Consequently, subtracting unaligned

frames can produce displacement columns of norm approximately two even when their represented direction has barely changed. We align each expert's rank-indexed singular-vector pairs to the base:

$$s_{i,k} := \operatorname{sign}\big(\langle u_{0,k}, u_{i,k}\rangle\big), \qquad \operatorname{sign}(0) := 1, \tag{64}$$

$$u_{i,k} \leftarrow s_{i,k} u_{i,k}, \qquad v_{i,k} \leftarrow s_{i,k} v_{i,k}. \tag{65}$$

Applying the same sign to both frames leaves W_i unchanged. All later occurrences of U_i and V_i refer to these aligned frames. Exactly repeated singular values admit additional within-block rotations, such as $U_{i,C} \mapsto U_{i,C}Q$ and $V_{i,C} \mapsto V_{i,C}Q$ for an orthogonal Q. The procedure resolves joint signs but does not explicitly canonicalize this additional freedom.

F.3 Stiefel Tangent Projection

For an anchor $U_0 \in \mathrm{St}(d_{\mathrm{out}}, q)$,

$$T_{U_0}\mathrm{St}(d_{\mathrm{out}}, q) = \left\{\xi \in \mathbb{R}^{d_{\mathrm{out}} \times q} : U_0^\top \xi + \xi^\top U_0 = 0\right\}. \tag{66}$$

The orthogonal projection under the Frobenius inner product is

$$\Pi_{U_0}(Z) = Z - U_0 \operatorname{sym}(U_0^\top Z), \qquad \operatorname{sym}(A) = \tfrac{1}{2}(A + A^\top). \tag{67}$$

After sign alignment, define the finite frame displacements

$$\Delta U_i := U_i - U_0, \qquad \Delta V_i := V_i - V_0, \tag{68}$$

and their tangent projections

$$\xi_{U,i} = \Pi_{U_0}(\Delta U_i), \qquad \xi_{V,i} = \Pi_{V_0}(\Delta V_i). \tag{69}$$

The right-frame projection is defined analogously at V_0. Using a common anchor makes the expert directions directly composable in shared frame coordinates.

F.4 Singular-Mode Masking

The masking step is motivated by the empirical observation that trailing singular directions are less stable across experts and that retaining their displacements can degrade merged performance. For a keep ratio $\rho_{\mathrm{keep}} \in (0, 1]$, define

$$\begin{aligned} k_{\mathrm{keep}} &:= \mathrm{round}(\rho_{\mathrm{keep}} q), \\ D_{\mathrm{keep}} &:= \mathrm{Diag}\left((\mathbf{1}\{k \le k_{\mathrm{keep}}\})_{k=1}^{q}\right). \end{aligned} \tag{70}$$

We retain the leading columns of the projected frame displacements:

$$\widetilde{\xi}_{U,i} = \xi_{U,i} D_{\mathrm{keep}}, \qquad \widetilde{\xi}_{V,i} = \xi_{V,i} D_{\mathrm{keep}}. \tag{71}$$

Unless otherwise stated, $\rho_{\mathrm{keep}} = 0.9$. This operation selects displacement coordinates, not singular values: the full q-mode spectrum Σ_0 is retained in the final reconstruction, so masking is not a low-rank truncation of the weight.

Column masking need not preserve the Stiefel tangent constraints. We therefore use the masked displacements to construct local effect proxies and form their weighted aggregate, then restore tangent feasibility before retraction as described in Appendix F.6.

F.5 Retention Coefficients with a Unit-Retention Target

Local effect proxies. Linearizing the reconstruction map $W(U, V) = U\Sigma_0 V^\top$ at (U_0, V_0) gives the local first-order effect proxy

$$g_i := \widetilde{\xi}_{U,i}\Sigma_0 V_0^\top + U_0 \Sigma_0 \widetilde{\xi}_{V,i}^\top. \tag{72}$$

The proxies are computed from the masked displacements before aggregate tangent projection. We measure their interactions through the Gram matrix

$$\Gamma_{ij} := \langle g_i, g_j \rangle_F, \qquad \Gamma \in \mathbb{R}^{K \times K}, \tag{73}$$

where $\langle A, B\rangle_F = \mathrm{tr}(A^\top B)$. For coefficients $c \in \mathbb{R}^K$, let $g(c) = \sum_{i=1}^K c_i g_i$. For every nonzero proxy g_i, define its self-retention by

$$\mathrm{ret}_i(c) := \frac{\langle g(c), g_i\rangle_F}{\|g_i\|_F^2} = \frac{(\Gamma c)_i}{\Gamma_{ii}}. \tag{74}$$

The unit-retention target $\mathrm{ret}_i(c) = 1$ asks the merged proxy to retain a unit projection coefficient along each expert proxy. The corresponding ideal equations are

$$\Gamma c = b, \qquad b := \mathrm{diag}(\Gamma). \tag{75}$$

Zero proxies, ridge stabilization, and clipping. Let $\mathcal{I} = \{i : \|g_i\|_F > 0\}$ denote the active experts for this matrix. Zero-norm proxies impose no retention constraint; we omit their rows and columns and set their coefficients to zero. When $\mathcal{I} \neq \varnothing$, we solve

$$\big(\Gamma_{\mathcal{I}\mathcal{I}} + \lambda_{\mathrm{ridge}} I_{|\mathcal{I}|}\big)\bar{c}_{\mathcal{I}} = b_{\mathcal{I}}, \tag{76}$$

and apply entrywise clipping:

$$c_i^\star = \begin{cases} \min\{c_{\max}, \max\{c_{\min}, \bar{c}_i\}\}, & i \in \mathcal{I}, \\ 0, & i \notin \mathcal{I}. \end{cases} \tag{77}$$

We use $\lambda_{\mathrm{ridge}} = 10^{-12}$ as a small numerical stabilizer and $[c_{\min}, c_{\max}] = [0, 1.5]$ to keep the coefficients nonnegative and limit amplification. If $\mathcal{I} = \varnothing$, the Gram solve is skipped and all coefficients remain zero.

Unit self-retention is a target in the local proxy space. Ridge stabilization and clipping can change the attained proxy retention, and subsequent tangent projection and nonlinear retraction further change the resulting weight displacement. The procedure therefore does not guarantee exact retention in the reconstructed model or its task performance.

F.6 Retraction and Reconstruction

We combine the masked expert displacements using $c^\star$ and project the aggregates back onto the tangent spaces:

$$\begin{aligned} \xi_{U,\star} &= \Pi_{U_0}\left(\sum_{i=1}^K c_i^\star \widetilde{\xi}_{U,i}\right), \\ \xi_{V,\star} &= \Pi_{V_0}\left(\sum_{i=1}^K c_i^\star \widetilde{\xi}_{V,i}\right). \end{aligned} \tag{78}$$

Polar retraction. For $X \in \mathbb{R}^{d\times q}$ with $d \geq q$, write a thin SVD $X = P_X S_X Q_X^\top$ and define

$$\mathrm{polar}(X) := P_X Q_X^\top. \tag{79}$$

This has orthonormal columns and is a nearest Stiefel factor to X in Frobenius norm. When X has full column rank,

$$\mathrm{polar}(X) = X(X^\top X)^{-1/2}. \tag{80}$$

For rank-deficient X, the SVD expression selects one nearest Stiefel factor, which need not be unique.

The ISO-Merger retraction arguments have full column rank in exact arithmetic. In particular, tangent feasibility gives

$$\begin{aligned} (U_0 + \xi_{U,\star})^\top (U_0 + \xi_{U,\star}) &= I_q + \xi_{U,\star}^\top \xi_{U,\star} \succ 0, \\ (V_0 + \xi_{V,\star})^\top (V_0 + \xi_{V,\star}) &= I_q + \xi_{V,\star}^\top \xi_{V,\star} \succ 0. \end{aligned} \tag{81}$$

The final factors and merged weight are

$$\begin{aligned} U_\star &= \mathrm{polar}(U_0 + \xi_{U,\star}), \\ V_\star &= \mathrm{polar}(V_0 + \xi_{V,\star}), \\ W_\star &= U_\star \Sigma_0 V_\star^\top. \end{aligned} \tag{82}$$

Because both frames have orthonormal columns, $W_\star \in \mathcal{F}(W_0)$ in exact arithmetic; the implementation preserves this property up to floating-point error. If all coefficients are zero, this construction returns the base matrix in exact arithmetic.

Algorithm 1 ISO-Merger

Require: Base model θ_0 and K shared-base experts $\{\theta_i\}_{i=1}^K$
Require: Keep ratio $\rho_{\text{keep}} = 0.9$, ridge $\lambda_{\text{ridge}} = 10^{-12}$, clip bounds $[c_{\min}, c_{\max}] = [0, 1.5]$
Ensure: Merged model $\theta_\star$

for each 2D projection, embedding, or unembedding parameter **do**
  Read corresponding matrices W_0 and $\{W_i\}_{i=1}^K$
  Compute the thin SVD $W_0 = U_0 \Sigma_0 V_0^\top$
  $q \leftarrow \min\{d_{\text{out}}, d_{\text{in}}\}$; $k_{\text{keep}} \leftarrow \text{round}(\rho_{\text{keep}} q)$
  $D_{\text{keep}} \leftarrow \text{Diag}((\mathbf{1}\{k \le k_{\text{keep}}\})_{k=1}^q)$
  for $i = 1, \dots, K$ **do**
    Compute the thin SVD $W_i = U_i \Sigma_i V_i^\top$
    for $k = 1, \dots, q$ **do**
      $s_{i,k} \leftarrow \text{sign}(\langle u_{0,k}, u_{i,k} \rangle)$ ▷ $\text{sign}(0) = 1$
      $u_{i,k} \leftarrow s_{i,k} u_{i,k}$; $v_{i,k} \leftarrow s_{i,k} v_{i,k}$
    end for
    $\xi_{U,i} \leftarrow \Pi_{U_0}(U_i - U_0)$; $\xi_{V,i} \leftarrow \Pi_{V_0}(V_i - V_0)$
    $\widetilde{\xi}_{U,i} \leftarrow \xi_{U,i} D_{\text{keep}}$; $\widetilde{\xi}_{V,i} \leftarrow \xi_{V,i} D_{\text{keep}}$
    $g_i \leftarrow \widetilde{\xi}_{U,i} \Sigma_0 V_0^\top + U_0 \Sigma_0 \widetilde{\xi}_{V,i}^\top$
  end for
  $c^\star \leftarrow 0 \in \mathbb{R}^K$; $\mathcal{I} \leftarrow \{i : \|g_i\|_F > 0\}$
  if $\mathcal{I} \neq \varnothing$ **then**
    Form $\Gamma_{\mathcal{I}\mathcal{I}}$ with entries $\Gamma_{ij} = \langle g_i, g_j \rangle_F$
    $b_{\mathcal{I}} \leftarrow \text{diag}(\Gamma_{\mathcal{I}\mathcal{I}})$
    Solve $(\Gamma_{\mathcal{I}\mathcal{I}} + \lambda_{\text{ridge}} I_{|\mathcal{I}|}) \bar{c}_{\mathcal{I}} = b_{\mathcal{I}}$
    $c^\star_{\mathcal{I}} \leftarrow \text{clip}(\bar{c}_{\mathcal{I}}; c_{\min}, c_{\max})$
  end if
  $\xi_{U,\star} \leftarrow \Pi_{U_0}(\sum_{i=1}^K c_i^\star \widetilde{\xi}_{U,i})$
  $\xi_{V,\star} \leftarrow \Pi_{V_0}(\sum_{i=1}^K c_i^\star \widetilde{\xi}_{V,i})$
  $U_\star \leftarrow \text{polar}(U_0 + \xi_{U,\star})$; $V_\star \leftarrow \text{polar}(V_0 + \xi_{V,\star})$
  Store $W_\star = U_\star \Sigma_0 V_\star^\top$ in the corresponding parameter of $\theta_\star$
end for
for each 1D parameter **do**
  Store $w_\star = w_0 + \frac{1}{K} \sum_{i=1}^K (w_i - w_0)$ in $\theta_\star$
end for
return $\theta_\star$

F.7 Merged Parameter Scope

All per-layer 2D projection matrices, together with embedding and unembedding matrices, are merged using the construction above. One-dimensional parameters, including normalization scales and attention biases, use a task-vector average:

$$w_\star = w_0 + \frac{1}{K} \sum_{i=1}^K (w_i - w_0) = \frac{1}{K} \sum_{i=1}^K w_i. \tag{83}$$

These parameters do not use the matrix-specific retention coefficients.

F.8 Full ISO-Merger Algorithm

Algorithm 1 follows the same order as the preceding construction: sign alignment, tangent projection, mode masking, proxy-based coefficient computation, aggregate projection, and polar reconstruction.

G Data-Free Merging Experimental details and Additional Results

G.1 Baselines

We compare against five data-free baselines: Task Arithmetic (Ilharco et al., 2022), TIES (Yadav et al., 2023), TSV-Merge (Gargiulo et al., 2025), RAM (Yuan et al., 2026), and OrthoMerge-G-TIES (Yang et al., 2026).

The first four methods operate on Euclidean task-vector representations, while OrthoMerge-G-TIES additionally introduces a geometry-aware orthogonal component and is described separately below. Let θ_0 denote the shared base model and let $\{\theta_i\}_{i=1}^K$ be K expert models fine-tuned from this same initialization. The task vector of expert i is

$$\tau_i := \theta_i - \theta_0. \tag{84}$$

A data-free merging method constructs a merged update τ_{merge} without using training data and outputs

$$\theta_{\text{merge}} = \theta_0 + \tau_{\text{merge}}. \tag{85}$$

For a matrix-shaped parameter in layer ℓ, we use the layer-wise notation

$$\Delta W_i^{(\ell)} := W_i^{(\ell)} - W_0^{(\ell)}, \qquad W^{(\ell)} \in \mathbb{R}^{m_\ell \times n_\ell}. \tag{86}$$

Task Arithmetic merges experts by linearly combining their task vectors and adding the resulting update back to the base model:

$$\tau_{\text{TA}} = \lambda \sum_{i=1}^{K} \tau_i, \qquad \theta_{\text{TA}} = \theta_0 + \tau_{\text{TA}}. \tag{87}$$

Task Arithmetic is computationally cheap, requiring only vector additions and a global scaling coefficient. However, it does not explicitly address sign conflicts, redundant updates, layer-wise geometry, or sparse task-specific signals. As a result, interference and signal dilution can occur when expert updates overlap or when important updates are distributed over disjoint parameter regions.

TIES-Merging follows three steps: *Trim*, *Elect Sign*, and *Merge*. It first removes small-magnitude entries from each task vector, then elects a consensus sign for every coordinate, and finally averages only the updates whose signs agree with the elected sign.

Let $\rho_{\text{TIES}} \in (0, 1]$ be the retained density. The trimmed update for expert i is

$$m_i = \text{TopKMask}(|\tau_i|, \rho_{\text{TIES}}), \qquad \widetilde{\tau}_i = m_i \odot \tau_i, \tag{88}$$

where $m_i^p = 1$ indicates that coordinate p is among the largest ρ_{TIES} fraction of entries of $|\tau_i|$. The elected sign at coordinate p is

$$\gamma^p = \text{sign}\left(\sum_{i=1}^{K} \widetilde{\tau}_i^p\right). \tag{89}$$

TIES then keeps only the experts aligned with this sign:

$$A_p = \{i : \widetilde{\tau}_i^p \neq 0,\ \text{sign}(\widetilde{\tau}_i^p) = \gamma^p\}. \tag{90}$$

The merged coordinate is

$$\tau_{\text{TIES}}^p = \begin{cases} \dfrac{1}{|A_p|} \sum_{i \in A_p} \widetilde{\tau}_i^p, & |A_p| > 0, \\ 0, & |A_p| = 0, \end{cases} \tag{91}$$

and the final model is

$$\theta_{\text{TIES}} = \theta_0 + \lambda \tau_{\text{TIES}}. \tag{92}$$

TIES remains an element-wise method, but it is more robust than naive averaging when many small updates are irrelevant or when experts update the same coordinate in opposite directions. Its main limitation is that it does not explicitly model matrix-level subspace geometry.

TSV-Merge, short for Task Singular Vectors Merge, operates on layer-wise task matrices rather than fully flattened vectors. For each 2D weight matrix, TSV decomposes each expert update with SVD, keeps the leading singular components, orthogonalizes the concatenated singular-vector bases, and reconstructs the merged matrix.

For layer ℓ, compute

$$\Delta W_i^{(\ell)} = U_i \Sigma_i V_i^\top. \tag{93}$$

After truncating to rank k_ℓ,

$$\Delta W_i^{(\ell)} \approx U_i^{(k)} \Sigma_i^{(k)} {V_i^{(k)}}^\top. \tag{94}$$

The truncated bases and spectra are concatenated as

$$U = [U_1^{(k)} | \cdots | U_K^{(k)}], \qquad V = [V_1^{(k)} | \cdots | V_K^{(k)}], \qquad \Sigma = \mathrm{blockdiag}(\Sigma_1^{(k)}, \ldots, \Sigma_K^{(k)}). \tag{95}$$

TSV reduces singular-vector interference by orthogonalizing U and V. In the whitening form,

$$U_\perp = U(U^\top U)^{-1/2}, \qquad V_\perp = V(V^\top V)^{-1/2}. \tag{96}$$

A numerically stable implementation can use the orthogonal Procrustes form. If $U = P_U D_U Q_U^\top$ and $V = P_V D_V Q_V^\top$, then

$$U_\perp = P_U Q_U^\top, \qquad V_\perp = P_V Q_V^\top. \tag{97}$$

The merged task matrix is reconstructed as

$$\widehat{\Delta}^{(\ell)} = U_\perp \Sigma V_\perp^\top, \qquad W_{\mathrm{TSV}}^{(\ell)} = W_0^{(\ell)} + \alpha \widehat{\Delta}^{(\ell)}. \tag{98}$$

For non-matrix parameters, such as biases or normalization vectors, TSV typically falls back to Task Arithmetic.

TSV is a spectral, subspace-level method. By decorrelating singular-vector bases, it can reduce interference that is invisible to element-wise rules. This comes at a higher computational cost because SVDs are required for matrix-shaped layers. The rank k_ℓ is either treated as a hyperparameter or chosen through an automatic rank-reduction rule.

RAM is designed for sparse RL updates, where naive averaging can dilute coordinates that are important for only one expert. For each coordinate p, RAM identifies the experts with non-negligible updates:

$$A_p = \{ i \in \{1, \ldots, K\} : |\tau_i^p| > \epsilon_{\mathrm{act}} \}, \qquad c^p = |A_p|, \tag{99}$$

where ϵ_{act} is an active-update threshold. The merged coordinate is

$$\tau_{\mathrm{RAM}}^p = \begin{cases} 0, & c^p = 0, \\ \tau_j^p, & c^p = 1 \text{ and } A_p = \{j\}, \\ \dfrac{1}{c^p} \displaystyle\sum_{i \in A_p} \tau_i^p, & c^p \geq 2. \end{cases} \tag{100}$$

Thus, inactive coordinates are discarded, task-unique coordinates are preserved without being divided by K, and shared coordinates are averaged only over the experts that actively update them. The final model is

$$\theta_{\mathrm{RAM}} = \theta_0 + \tau_{\mathrm{RAM}}. \tag{101}$$

RAM is element-wise like TIES, but its selection criterion is different. Instead of trimming by global magnitude density and resolving signs, RAM separates inactive, unique, and shared update regions. This makes it suitable when RL-trained experts store task-specific behavior in sparse and partially non-overlapping parameter subsets.

OrthoMerge (Yang et al., 2026) is a geometry-preserving merging framework that factors each expert update into an *implicit orthogonal transformation* of the base weights and a Euclidean residual, merges the orthogonal components on the rotation manifold, and merges the residuals with a standard element-wise rule. The released implementation provides six variants: two Procrustes-target constructions (C, which fits the rotation only to rows whose update conflicts with the mean task vector, and G, which fits it to the full expert weights) combined with three residual backends (Task Arithmetic, TIES, TSV). We report the G-TIES variant for two reasons. First, TIES is a strong

residual backend in both of our merging settings: among the element-wise baselines it attains the best or near-best overall average on both backbones (Tab. 1 and 3), so we pair OrthoMerge with the classical method that is already competitive in our comparison. Second, the G construction fits the Procrustes rotation to the *full* expert weights, so the extracted transformation represents the entire expert update. The C construction instead fits it only to the rows that conflict with the mean task vector. Since ISO-Merger likewise composes each expert's full update—represented as motion of its singular frames under the fixed base spectrum—G-TIES is the directly comparable variant.

For each 2D weight matrix (excluding layer norms), OrthoMerge first solves a one-sided orthogonal Procrustes problem per expert,

$$R_i^{(\ell)} = \underset{R \in \mathrm{O}(n_\ell)}{\arg\min} \left\| W_0^{(\ell)} R - W_i^{(\ell)} \right\|_F = \bar{U}\bar{V}^\top, \qquad {W_0^{(\ell)}}^\top W_i^{(\ell)} = \bar{U}\bar{S}\bar{V}^\top, \tag{102}$$

so that the rotation acts only on the input (column) space of the layer. Each rotation is mapped to a skew-symmetric matrix by the inverse Cayley transform,

$$A_i^{(\ell)} = \left(R_i^{(\ell)} + I\right)^{-1}\left(R_i^{(\ell)} - I\right), \qquad A_i^{(\ell)} = -{A_i^{(\ell)}}^\top, \tag{103}$$

and the K skew matrices are aggregated with a magnitude-corrected mean that averages direction and intensity separately,

$$A_{\mathrm{mrg}}^{(\ell)} = \left(\tfrac{1}{K}\textstyle\sum_{i=1}^{K} \left\| A_i^{(\ell)} \right\|_F\right) \cdot \frac{\sum_{i=1}^{K} A_i^{(\ell)}}{\left\| \sum_{i=1}^{K} A_i^{(\ell)} \right\|_F}, \tag{104}$$

which prevents the norm shrinkage that a plain average of nearly orthogonal directions would induce. The merged rotation is recovered with the forward Cayley map and applied to the base weights,

$$R_{\mathrm{mrg}}^{(\ell)} = \left(I - A_{\mathrm{mrg}}^{(\ell)}\right)^{-1}\left(I + A_{\mathrm{mrg}}^{(\ell)}\right), \qquad W_{\mathrm{rot}}^{(\ell)} = W_0^{(\ell)} R_{\mathrm{mrg}}^{(\ell)}. \tag{105}$$

The part of each expert update not captured by its own rotation is treated as a Euclidean residual,

$$\delta_i^{(\ell)} = W_i^{(\ell)} - W_0^{(\ell)} R_i^{(\ell)}, \tag{106}$$

(for non-matrix parameters $\delta_i = \tau_i$), and the residuals are merged with the TIES rule of trimming, sign election, and sign-consistent averaging described above. The final model is

$$W_{\mathrm{OM}}^{(\ell)} = W_{\mathrm{rot}}^{(\ell)} + \lambda\, \tau_{\mathrm{TIES}}\left(\{\delta_i\}\right). \tag{107}$$

OrthoMerge is the geometry-aware baseline closest to ISO-Merger, but the two methods act in different coordinates. OrthoMerge applies a one-sided orthogonal transformation on the input side of each matrix: $W_0^{(\ell)} R$ preserves the left singular directions and singular values of $W_0^{(\ell)}$, so any output-side frame change must be represented through its Euclidean residual path.

ISO-Merger instead composes two-sided frame changes under the shared base spectrum, adapting both U and V in fixed-spectrum Stiefel coordinates. OrthoMerge's rotation class is contained in the more permissive output-subspace-retaining class $\mathcal{C}_L$ analyzed in Section 3. Even this larger class leaves a substantial unexplained-update ratio, whereas retaining the incoming spectrum while adapting both frames gives a low-residual description. This motivates ISO-Merger's two-sided frame parameterization.

G.2 Experimental deatils

Backbones and experts. For the Qwen2.5 setting, we use `Qwen2.5-7B-Instruct` as the shared base and merge three RLVR experts: CURE for coding (Wang et al., 2025c), ToolRL for tool use (Qian et al., 2025), and MemAgent for long-context memory (Yu et al., 2025). For the DeepSeek-R1-Distill setting, we use `DeepSeek-R1-Distill-Qwen-1.5B` as the shared base and merge Archer2.0 for coding (Wang et al., 2025a) and JustRL for math (He et al., 2025a).

Baseline hyperparameters. We follow the official implementations for all baselines. For TIES, we use $\lambda = 1.0$ and $\rho_{\mathrm{TIES}} = 0.2$. For TSV-Merge, we use $\alpha = 1.0$ with automatic rank reduction. For RAM, we use $\epsilon = 10^{-5}$. For ISO-Merger, all SVD operations are performed in FP64. For OrthoMerge-G-TIES, we use the official implementation with its default hyperparameters: FP32 SVD-based Procrustes, 20% retained density and $\lambda = 1.0$ for the TIES residual stage. An additional FP64-SVD comparison for OrthoMerge-G-TIES is reported in Appendix G.3.2.

Evaluation protocol. For the Qwen2.5 setting, we evaluate coding on LiveBench and LiveCodeBench, tool use on BFCL, and long-context memory on RULER HotpotQA and SQuAD at 32K–64K context lengths. For the DeepSeek-R1-Distill setting, we evaluate coding on LiveBench and LiveCodeBench, and math on AIME24, AIME25, AMC23, Minerva, and OlympiadBench. Unless otherwise specified, we generate 16 rollouts per prompt and report means and standard deviations across three evaluation seeds (42, 123, and 1234). BFCL scores are obtained from a single evaluation per merged model.

Decoding settings. For DeepSeek-R1-Distill math and coding evaluations, we use temperature 0.6 and top-$p = 0.95$. For Qwen2.5 LiveBench and LiveCodeBench, we use temperature 1.0. For BFCL, we use single-sample near-greedy decoding with temperature 0.001, following the official evaluation protocol. For RULER long-context evaluation, we use temperature 0.7. Inference is conducted with vLLM on NVIDIA H100 GPUs.

G.3 Additional results

G.3.1 Results on DeepSeek-R1-Distill-Qwen-1.5B

Tab. 3 reports the data-free composition of Archer2.0 for coding and JustRL for math. ISO-Merger achieves the highest aggregate score of 44.38, compared with 43.52 for TIES, the strongest baseline by overall average. Relative to the corresponding domain specialists, ISO-Merger matches or exceeds their reported mean scores on $7/9$ metrics—both LiveCodeBench measures and all five math benchmarks—compared with at most $3/9$ for any baseline.

At the domain level, ISO-Merger achieves 101.4% and 101.7% of the corresponding specialists' domain-averaged scores on coding and math, respectively, compared with 100.3% and 99.4% for TIES. Although ISO-Merger remains below the coding specialist on the two LiveBench metrics, it exceeds the specialist averages in both domains. Together, these results support balanced recovery and enhancement of complementary specialist performance in a single fixed-spectrum model, without post-merge training.

Table 3: **Main results on DeepSeek-R1-Distill-Qwen-1.5B with 2 RL experts**. Coding: pass accuracy (ACC) and unit-test accuracy (UT) on LiveBench (LB) and LiveCodeBench (LCB). Math: ACC on AIME 2024, AIME 2025, AMC 2023, Minerva, and OlympiadBench. Avg is the average over all columns. Best expert/merged result per column in **bold**. Shading indicates Experts and Ours. The final row reports ISO-Merger's score as a percentage of the corresponding domain expert's score. Ratios are computed from the reported means, with 100% indicating an equal score.

	Coding				Math					
	LB		LCB v5		AIME 2024	AIME 2025	AMC 2023	Minerva	OlympiadBench	Avg
Method	ACC	UT	ACC	UT	Avg@32	Avg@32	Avg@8	Avg@4	Avg@4	
Base Guo et al. (2025)	17.99 ± 0.29	26.24 ± 0.53	16.82 ± 0.28	20.85 ± 0.18	30.90 ± 1.15	23.40 ± 0.86	63.15 ± 0.38	27.33 ± 0.43	43.11 ± 0.26	29.98
Archer2.0 Wang et al. (2025a)	**26.66 ± 0.45**	**37.26 ± 0.55**	**26.90 ± 0.21**	**38.99 ± 0.37**	42.05 ± 0.43	28.02 ± 0.44	73.44 ± 0.50	30.09 ± 0.23	49.99 ± 0.65	39.27
JustRL He et al. (2025a)	22.23 ± 0.23	31.85 ± 0.74	23.05 ± 0.21	30.18 ± 0.31	**53.16 ± 0.44**	**36.84 ± 0.73**	**82.78 ± 0.51**	**34.71 ± 0.50**	**55.67 ± 0.42**	**41.16**
Task Arithmetic Ilharco et al. (2022)	26.22 ± 0.22	35.82 ± 0.11	26.70 ± 0.21	36.58 ± 0.32	50.83 ± 0.39	33.09 ± 1.16	80.82 ± 0.51	33.00 ± 0.34	54.31 ± 0.15	41.93
TIES Yadav et al. (2023)	26.37 ± 0.73	36.75 ± 0.90	27.62 ± 0.13	39.45 ± 0.15	54.51 ± 0.72	35.76 ± 0.30	82.13 ± 0.14	33.76 ± 0.48	55.36 ± 0.15	43.52
TSV Gargiulo et al. (2025)	26.40 ± 0.40	36.83 ± 0.15	27.61 ± 0.17	40.01 ± 0.26	53.16 ± 0.57	35.17 ± 0.72	80.97 ± 0.92	33.88 ± 0.46	54.68 ± 0.27	43.19
RAM Yuan et al. (2026)	**26.71 ± 0.29**	**36.99 ± 0.47**	27.18 ± 0.63	38.83 ± 0.67	54.20 ± 0.55	35.80 ± 0.71	82.28 ± 0.38	34.10 ± 0.34	55.54 ± 0.25	43.51
OrthoMerge-G-TIES Yang et al. (2026)	26.46 ± 0.62	36.73 ± 0.37	27.74 ± 0.30	39.86 ± 0.64	54.55 ± 1.44	35.14 ± 0.87	81.88 ± 0.63	33.55 ± 0.42	55.47 ± 0.24	43.49
ISO-Merger	25.52 ± 0.08	36.42 ± 0.55	**27.84 ± 0.13**	**41.84 ± 0.31**	**55.00 ± 0.52**	**37.81 ± 0.31**	**83.53 ± 1.02**	**34.77 ± 0.23**	**56.64 ± 0.05**	**44.38**
Expert-relative score (%)	95.7	97.7	103.5	107.3	103.5	102.6	100.9	100.2	101.7	–

G.3.2 Effect of SVD Precision in OrthoMerge

To test whether higher SVD precision closes the gap of OrthoMerge (Yang et al., 2026) to ISO-Merger, we additionally evaluate OrthoMerge-G-TIES with FP64 SVDs, keeping all other merging and evaluation settings unchanged. On `Qwen2.5-7B-Instruct` with three RL experts (Tab. 4), OrthoMerge achieves aggregate scores of 62.16 with its official FP32 configuration and 61.36 with FP64, compared with 63.80 for ISO-Merger. Thus, ISO-Merger retains its aggregate advantage under both configurations; matching SVD precision does not close the observed performance gap in this setting.

Table 4: **Effect of SVD precision on OrthoMerge-G-TIES.** We compare its official FP32 configuration with an FP64-SVD variant on `Qwen2.5-7B-Instruct` with three RL experts, keeping all other merging settings fixed. ISO-Merger uses FP64 SVDs and is included for reference. Metrics and evaluation protocols follow Table 1. The highest mean score in each column is shown in **bold**.

		Coding				Tool Use		Memory				
		LB		LCB v2		Live	Non-Live	HotpotQA		SQuAD		
Method	SVD	ACC	UT	ACC	UT	ACC	ACC	32K	64K	32K	64K	Avg
OrthoMerge-G-TIES	FP32	40.74 ± 2.00	**53.22 ± 1.90**	**31.98 ± 0.84**	**47.48 ± 0.77**	66.98	78.17	74.79 ± 0.49	74.20 ± 0.49	77.75 ± 0.12	76.32 ± 0.21	62.16
OrthoMerge-G-TIES	FP64	37.91 ± 1.70	49.37 ± 1.50	31.24 ± 1.03	46.42 ± 0.97	66.59	78.00	75.33 ± 0.56	73.92 ± 0.35	77.99 ± 0.47	76.84 ± 0.17	61.36
ISO-Merger	FP64	**41.81 ± 2.00**	52.73 ± 1.57	31.06 ± 1.88	45.69 ± 3.06	**72.32**	**81.07**	**79.46 ± 0.78**	**76.77 ± 0.27**	**79.10 ± 0.14**	**78.01 ± 0.32**	**63.80**

G.3.3 METRICS: BEST@4 AND WORST@4

Among n sampled responses, let c denote the number with binary correctness equal to one. For $n \geq k$, we report

$$\widehat{\text{best@}k} = 1 - \frac{\binom{n-c}{k}}{\binom{n}{k}}, \qquad \widehat{\text{worst@}k} = \frac{\binom{c}{k}}{\binom{n}{k}}.$$

In addition to average@16, we report best@4 and worst@4 to characterize the upper and lower tails of stochastic decoding performance.

- **best@k**=pass@k, the probability that at least one of k randomly drawn samples solves the problem (best-case behaviour over k draws).
- **worst@k**= $\binom{c}{k}/\binom{n}{k}$, the probability that all k randomly drawn samples solve the problem (consistency / worst-case reading).

For the Qwen2.5 setting, unit-test accuracy of LB/LCB coding benchmark is the fraction of unit tests passed, a real value in $[0, 1]$ rather than a binary outcome. BFCL is evaluated with single-sample, near-greedy decoding with temperature 0.001, following the official leaderboard protocol. So we only report best@4 and worst@4 for rest benchmark and metrics.

Table 5: **Main results on Qwen2.5-7B-Instruct with 3 RL experts under Best@4.** Coding: pass accuracy (ACC) on LiveBench (LB) and LiveCodeBench (LCB). Memory: RULER HotpotQA and SQuAD at 32K–64K context lengths under the Recurrent setting. Best expert/merged result per column in **bold**. Shading indicates Experts and Ours.

	Coding		Memory				
	LB	LCB v2	HotpotQA		SQuAD		
Method	ACC	ACC	32K	64K	32K	64K	Avg
Base Qwen et al. (2025)	48.35 ± 1.46	39.93 ± 0.64	68.98 ± 1.44	66.54 ± 1.51	79.62 ± 1.23	77.20 ± 0.30	63.44
RLVR-Coder Wang et al. (2025c)	48.97 ± 1.20	**41.13 ± 1.50**	70.03 ± 0.33	66.33 ± 0.79	81.90 ± 0.59	77.60 ± 0.42	64.33
RLVR-Tool Qian et al. (2025)	**51.54 ± 2.17**	39.91 ± 1.66	69.31 ± 0.61	67.71 ± 1.53	81.12 ± 0.80	78.74 ± 0.65	64.72
RLVR-Memory Yu et al. (2025)	49.74 ± 2.61	37.49 ± 3.49	**86.21 ± 0.26**	**86.38 ± 0.90**	**87.26 ± 0.32**	**88.54 ± 0.39**	72.60
Task Arithmetic Ilharco et al. (2022)	51.99 ± 1.78	42.46 ± 0.28	81.65 ± 0.60	81.37 ± 0.37	84.13 ± 0.44	87.58 ± 0.31	71.53
TIES Yadav et al. (2023)	50.04 ± 2.18	42.34 ± 1.42	85.80 ± 0.18	84.32 ± 0.04	85.84 ± 0.36	88.05 ± 0.37	72.73
TSV Gargiulo et al. (2025)	48.82 ± 0.29	39.04 ± 2.44	84.75 ± 0.37	83.54 ± 0.37	86.39 ± 0.35	87.72 ± 0.44	71.71
RAM Yuan et al. (2026)	48.73 ± 0.54	41.06 ± 0.25	85.91 ± 0.52	84.17 ± 0.61	86.62 ± 0.43	**88.39 ± 0.57**	72.48
OrthoMerge-G-TIES Yang et al. (2026)	**52.20 ± 2.00**	**42.52± 1.77**	84.38 ± 0.37	83.47 ± 0.40	86.06 ± 0.26	87.66 ± 0.41	72.72
Ours	50.39 ± 1.98	40.68 ± 1.99	**86.05 ± 0.78**	**84.92 ± 0.37**	**86.98 ± 0.21**	88.23 ± 0.38	**72.88**

Tab. 5, 6, 7, and 8 show that ISO-Merger achieves the highest upper and lower tails under both backbones. In particular, for worst@4, which is substantially more challenging, ISO-Merger reaches an overall average of 54.85, compared with 53.23 for the strongest training-free baseline. On `DeepSeek-R1-Distill-Qwen-1.5B`, ISO-Merger reaches a total average of 34.98, improving over the best baseline at 33.62. For best@4, ISO-Merger remains competitive with the original specialists while composing multiple skills into a single model. These results further demonstrate that different RL experts initialized from the same base model can be combined in fixed-spectrum Stiefel coordinates.

Table 6: **Main results on Qwen2.5-7B-Instruct with 3 RL experts under Worst@4.** Coding: pass accuracy (ACC) on LiveBench (LB) and LiveCodeBench (LCB). Memory: RULER HotpotQA and SQuAD at 32K–64K context lengths under the Recurrent setting. Best expert/merged result per column in **bold**. Shading indicates Experts and Ours.

	Coding		Memory				
	LB	LCB v2	HotpotQA		SQuAD		
Method	ACC	ACC	32K	64K	32K	64K	Avg
Base Qwen et al. (2025)	24.44 ± 0.38	16.99 ± 0.11	30.76 ± 0.95	24.34 ± 0.24	37.71 ± 1.31	35.55 ± 0.92	28.30
RLVR-Coder Wang et al. (2025c)	**25.84 ± 1.57**	**19.96 ± 0.41**	30.37 ± 0.84	24.72 ± 0.80	38.49 ± 1.38	35.89 ± 0.65	29.21
RLVR-Tool Qian et al. (2025)	21.87 ± 0.84	16.78 ± 1.66	30.61 ± 1.23	23.48 ± 1.03	37.83 ± 1.69	35.89 ± 1.27	27.74
RLVR-Memory Yu et al. (2025)	24.26 ± 0.76	17.28 ± 2.64	**70.22 ± 0.73**	**68.40 ± 0.14**	**69.66 ± 0.80**	**67.09 ± 0.51**	**52.82**
Task Arithmetic Ilharco et al. (2022)	26.24 ± 2.29	21.46 ± 0.52	61.86 ± 0.16	58.54 ± 0.90	64.89 ± 0.89	61.44 ± 0.56	49.07
TIES Yadav et al. (2023)	30.48 ± 2.58	21.41 ± 0.66	65.91 ± 0.74	65.12 ± 0.20	**69.79 ± 0.59**	**66.66 ± 0.46**	53.23
TSV Gargiulo et al. (2025)	26.12 ± 1.78	19.39 ± 1.36	65.33 ± 0.99	63.43 ± 0.55	69.36 ± 0.90	65.32 ± 0.98	51.49
RAM Yuan et al. (2026)	27.41 ± 1.87	21.72 ± 0.57	65.30 ± 0.34	65.82 ± 0.90	69.57 ± 0.43	65.75 ± 0.61	52.59
OrthoMerge-G-TIES Yang et al. (2026)	28.71 ± 2.11	21.65 ± 0.66	64.88 ± 1.15	64.80 ± 0.59	68.31 ± 0.38	63.62 ± 0.54	52.00
Ours	**33.50 ± 2.28**	**21.80 ± 1.61**	**71.78 ± 0.96**	**66.12 ± 0.31**	69.75 ± 0.29	66.13 ± 0.52	**54.85**

Table 7: **Main results on DeepSeek-R1-Distill-Qwen-1.5B with 2 RL experts under Best@4**. Coding ACC on LB and LCB v5. Math: ACC on AIME24, AIME25, AMC23, Minerva, and OlympiadBench. Best expert/merged result per column in **bold**. Shading indicates Experts and Ours.

	Coding		Math					
	LB	LCB v5	AIME24	AIME25	AMC23	Minerva	Olympiad	Total
Method	ACC	ACC	ACC	ACC	ACC	ACC	ACC	Avg
Base Guo et al. (2025)	29.68 ± 0.17	25.96 ± 0.42	54.49 ± 1.34	33.34 ± 0.99	80.49 ± 0.80	41.67 ± 2.00	56.05 ± 0.56	45.95
Archer2.0 Wang et al. (2025a)	**36.11 ± 0.45**	**36.75 ± 0.27**	64.74 ± 0.72	39.88 ± 1.12	86.49 ± 0.12	43.38 ± 0.52	61.19 ± 0.32	52.65
JustRL He et al. (2025a)	31.67 ± 0.42	32.63 ± 0.62	**71.62 ± 0.16**	**49.25 ± 1.32**	**90.83 ± 0.65**	**45.22 ± 1.08**	**64.49 ± 0.91**	55.10
Task Arithmetic Ilharco et al. (2022)	**36.10 ± 0.45**	37.18 ± 0.35	72.34 ± 0.54	47.12 ± 2.32	90.51 ± 0.17	44.61 ± 0.46	64.20 ± 0.56	56.01
TIES Yadav et al. (2023)	34.31 ± 0.69	37.66 ± 0.31	**73.40 ± 0.42**	47.37 ± 0.77	89.62 ± 0.38	45.10 ± 0.87	64.35 ± 0.35	55.97
TSV Gargiulo et al. (2025)	35.84 ± 0.34	**37.67 ± 0.39**	72.41 ± 1.07	46.94 ± 1.18	89.93 ± 1.10	**45.10 ± 0.92**	64.94 ± 0.42	56.12
RAM Yuan et al. (2026)	35.34 ± 0.87	36.96 ± 1.04	73.06 ± 0.62	49.50 ± 1.81	90.36 ± 0.63	44.98 ± 0.76	64.59 ± 0.55	56.40
OrthoMerge-G-TIES Yang et al. (2026)	35.18 ± 0.72	37.47 ± 0.50	73.95 ± 1.16	47.13 ± 0.96	89.82 ± 0.51	44.36 ± 0.56	64.74 ± 0.68	56.09
Ours	34.04 ± 0.56	36.87 ± 0.09	72.69 ± 0.34	**50.43 ± 0.71**	**91.31 ± 0.61**	44.73 ± 0.87	**64.99 ± 0.56**	**56.44**

G.4 Ablations of ISO-Merger Design Choices

Setup. We isolate three design choices in ISO-Merger: spectrum re-basing, retention-aware composition, and mode masking. First, **TIES + spec-rebase** applies the strongest Euclidean merging baseline, TIES (Yadav et al., 2023), and subsequently re-bases the merged checkpoint to the base spectrum, testing whether spectrum preservation alone accounts for ISO-Merger's gains. Second, **w/o retention-aware composition** replaces the learned retention coefficients with uniform coefficients, $c_1 = c_2 = 1$. Third, **w/o mode masking** retains all frame modes by setting $\rho_{\text{keep}} = 1$, instead of $\rho_{\text{keep}} = 0.9$ used by the full method. All other ISO-Merger components and evaluation settings are held fixed. Results on `DeepSeek-R1-Distill-Qwen-1.5B` are reported in Tab. 9.

Results. The ablations separate the contributions of the three design choices. Spectrum re-basing alone is insufficient: TIES + spec-rebase obtains an aggregate score of 43.09, below full ISO-Merger at 44.38 and also below vanilla TIES at 43.52 (Tab. 3). Mode masking has the largest effect among the tested components: removing it reduces the aggregate score from 44.38 to 42.02, with degradation in both coding and math. Retention-aware composition provides a smaller but complementary improvement: uniform coefficients reach 44.14. Interestingly, removing retention-aware composition slightly improves several math metrics, but reduces the coding-domain average from 32.91 to 32.06; the full method therefore achieves a better cross-domain balance and the highest overall score. Together, these results show that ISO-Merger's gain is not explained by spectrum re-basing alone:

Table 8: **Main results on DeepSeek-R1-Distill-Qwen-1.5B with 2 RL experts under Worst@4**. Coding ACC on LB and LCB v5. Math: ACC on AIME24, AIME25, AMC23, Minerva, and OlympiadBench. Best expert/merged result per column in **bold**. Shading indicates Experts and Ours.

	Coding		Math					
	LB	LCB v5	AIME24	AIME25	AMC23	Minerva	Olympiad	Total
Method	ACC	ACC	ACC	ACC	ACC	ACC	ACC	Avg
Base Guo et al. (2025)	8.38 ± 0.38	8.95 ± 0.47	12.42 ± 0.92	12.96 ± 0.56	46.02 ± 0.64	14.95 ± 0.76	30.17 ± 0.73	19.12
Archer2.0 Wang et al. (2025a)	**15.95 ± 0.80**	**17.39 ± 0.24**	19.80 ± 0.63	17.36 ± 0.36	57.83 ± 1.03	18.75 ± 1.08	38.22 ± 0.60	26.47
JustRL He et al. (2025a)	14.02 ± 0.22	14.42 ± 0.14	**33.61 ± 0.93**	**27.25 ± 1.15**	**72.89 ± 0.65**	**23.90 ± 0.60**	**46.32 ± 0.14**	**33.20**
Task Arithmetic Ilharco et al. (2022)	15.82 ± 0.26	16.67 ± 0.50	28.49 ± 0.56	20.99 ± 0.47	68.35 ± 0.85	21.94 ± 0.92	43.41 ± 0.64	30.81
TIES Yadav et al. (2023)	**17.76 ± 0.87**	18.22 ± 0.27	33.40 ± 1.45	25.21 ± 0.22	71.85 ± 0.97	23.28 ± 1.25	45.23 ± 0.07	33.57
TSV Gargiulo et al. (2025)	15.91 ± 0.52	17.46 ± 0.16	32.33 ± 0.38	24.69 ± 1.48	68.36 ± 1.44	24.02 ± 0.87	42.86 ± 0.81	32.23
RAM Yuan et al. (2026)	17.41 ± 1.11	17.84 ± 0.25	33.90 ± 0.86	24.10 ± 0.14	71.46 ± 0.29	25.12 ± 0.35	45.48 ± 0.12	33.62
OrthoMerge-G-TIES Yang et al. (2026)	16.45 ± 0.67	18.27 ± 0.19	33.21 ± 1.99	24.86 ± 0.60	70.85 ± 0.80	23.77 ± 0.21	45.28 ± 0.60	33.24
Ours	16.18 ± 0.18	**18.95 ± 0.45**	**36.10 ± 1.01**	**27.41 ± 0.83**	**73.23 ± 1.81**	**25.12 ± 1.21**	**47.85 ± 0.12**	**34.98**

mode selection provides the largest benefit among the tested choices, while retention-aware weighting further improves balanced specialist composition.

Table 9: **ISO-Merger ablations on DeepSeek-R1-Distill-Qwen-1.5B with two RL experts.** All variants merge the same Archer2.0 and JustRL checkpoints and follow the evaluation protocol of Tab. 3. Values are mean ± standard deviation; Avg is the average over all columns. Best expert/merged result per column in **bold**. Shading indicates Experts and Ours.

Method	Coding				Math					Avg
	LB		LCB v5		AIME 2024	AIME 2025	AMC 2023	Minerva	OlympiadBench	
	ACC	UT	ACC	UT	Avg@32	Avg@32	Avg@8	Avg@4	Avg@4	
Base Guo et al. (2025)	17.99 ± 0.29	26.24 ± 0.53	16.82 ± 0.28	20.85 ± 0.18	30.90 ± 1.15	23.40 ± 0.86	63.15 ± 0.38	27.33 ± 0.43	43.11 ± 0.26	29.98
Archer2.0 Wang et al. (2025a)	**26.66 ± 0.45**	**37.26 ± 0.55**	**26.90 ± 0.21**	**38.99 ± 0.37**	42.05 ± 0.43	28.02 ± 0.44	73.44 ± 0.50	30.09 ± 0.23	49.99 ± 0.65	39.27
JustRL He et al. (2025a)	22.23 ± 0.23	31.85 ± 0.74	23.05 ± 0.21	30.18 ± 0.31	**53.16 ± 0.44**	**36.84 ± 0.73**	**82.78 ± 0.51**	**34.71 ± 0.50**	**55.67 ± 0.42**	**41.16**
ISO-Merger (full)	25.52 ± 0.08	36.42 ± 0.55	**27.84 ± 0.13**	**41.84 ± 0.31**	55.00 ± 0.52	**37.81 ± 0.31**	83.53 ± 1.02	34.77 ± 0.23	56.64 ± 0.05	**44.38**
w/o retention-aware composition	24.38 ± 0.12	35.62 ± 0.50	27.02 ± 0.33	41.22 ± 0.27	**55.62 ± 1.05**	37.29 ± 0.23	**84.29 ± 1.31**	**34.90 ± 0.23**	**56.91 ± 0.12**	44.14
w/o mode masking	21.48 ± 0.08	33.76 ± 0.04	24.34 ± 0.46	39.36 ± 0.30	52.78 ± 0.51	35.42 ± 0.68	82.18 ± 0.70	33.33 ± 0.09	55.54 ± 0.54	42.02
TIES + spec-rebase	**26.43 ± 0.45**	**36.50 ± 0.56**	27.19 ± 0.06	38.09 ± 0.45	52.88 ± 0.93	36.01 ± 0.32	81.88 ± 0.94	33.70 ± 0.50	55.09 ± 0.17	43.09

H Numerical Precision of the SVD-Based Retraction

In exact arithmetic, the ISO reconstruction

$$W^+ = U^+ \Sigma_0 (V^+)^\top$$

has singular values $\mathrm{diag}(\Sigma_0)$ whenever U^+ and V^+ have orthonormal columns. In practice, however, the polar retraction is computed using a finite-precision SVD, so numerical errors in the decomposition can weaken this invariant.

We therefore compare four PyTorch SVD configurations: FP32 and FP64 on CPU and GPU. As a representative sanity check, we extract the `q_proj` matrix from layer 10 of `Qwen3-1.7B-Base`,

$$W_0 \in \mathbb{R}^{2048\times 2048},$$

compute

$$W_0 = U\Sigma V^\top,$$

and reconstruct

$$W_{\mathrm{rec}} = U\Sigma V^\top.$$

We report the mean squared spectral reconstruction error

$$\epsilon_\sigma := \frac{1}{q}\,\|\sigma(W_{\mathrm{rec}}) - \sigma(W_0)\|_2^2, \tag{108}$$

together with the wall-clock decomposition time.

Tab. 10 shows that FP32 GPU SVD introduces substantially larger numerical error than FP64 GPU SVD: 7.7899×10^{-4} versus 2.8924×10^{-8} in this representative matrix. FP64 reduces the error by approximately 2.7×10^{4} while increasing the measured runtime only from 0.894 to 0.923 seconds. We therefore use FP64 GPU SVD for the polar retraction throughout ISO.

Table 10: Runtime and mean squared spectral reconstruction error under different PyTorch SVD precision and device configurations.

Precision	Device	Mean squared spectral error	Time (s)
FP32	CPU	3.6633e-7	2.115
FP64	CPU	1.1126e-8	3.699
FP32	GPU	**7.7899e-4**	0.894
FP64	GPU	2.8924e-8	0.923

I Online RLVR Training Details

The ISO parameterization, factor gradients, and retraction are described in Section 4.2.

I.1 Training and Evaluation Configurations

Common setup. All experiments are implemented using verl (Sheng et al., 2025) and run on NVIDIA H100 80 GB GPUs. We use the DAPO algorithm (Yu et al., 2026) and hold all non-optimizer settings fixed between the weight-space and ISO variants.

We use asymmetric policy-ratio clipping with

$$(\epsilon_{\text{low}}, \epsilon_{\text{high}}) = (0.2, 0.28). \tag{109}$$

The KL coefficient is set to

$$\beta_{\text{KL}} = 10^{-3},$$

with the KL term applied as a loss-shaping term rather than as part of the rollout reward. We enable online dynamic filtering by removing prompt groups whose sampled rollouts are either all correct or all incorrect. Unless stated otherwise, training rollouts use temperature 1.0 and top-$p = 1.0$.

We set weight decay to zero for both the weight-space and ISO variants. At the learning rates used here, the standard coefficient $\lambda = 10^{-2}$ falls below the BF16-visible update scale and provides no measurable benefit in our RLVR runs (Zhu et al., 2025b).

Mathematical reasoning. We train `Qwen3-1.7B-Base` and `Qwen3-4B-Base` (Qwen Team, 2025) on DeepMath-103K (He et al., 2025b), following prior RLVR post-training recipes (John Schulman & Thinking Machines Lab, 2025).

For `Qwen3-1.7B-Base` and `Qwen3-4B-Base`, we use a global prompt batch size of 256 and train for 400 actor updates, corresponding to approximately one nominal pass over the prompt set before online filtering. The 1.7B runs sample 16 rollouts per prompt, whereas the 4B runs sample 12. Both use a maximum prompt length of 1,024 tokens and a maximum response length of 16,384 tokens.

ISO-AdamW uses a learning rate of 1×10^{-6} for the frame variables (U, V), followed by polar retraction after every tentative factor update.[1] For weight-space AdamW, we sweep

$$\left\{ 7.5 \times 10^{-7},\ 1 \times 10^{-6},\ 2 \times 10^{-6},\ 3 \times 10^{-6} \right\}.$$

For the weight-space Muon baselines, we sweep

$$\left\{ 2.5 \times 10^{-5},\ 5 \times 10^{-5},\ 7.5 \times 10^{-5} \right\}.$$

The ISO-Muon experiment is conducted on `Qwen3-4B-Base` using a learning rate of 2.5×10^{-5}.

[1] On `Qwen3-1.7B-Base`, we compare learning rates 7.5×10^{-7} and 1×10^{-6}. The latter performs better, so we use it throughout the math experiments.

We evaluate mathematical reasoning on AIME 2024, AIME 2025, AMC 2023, Minerva, and OlympiadBench. For each problem, we sample 16 responses using temperature 1.0 and top-$p = 0.8$. The maximum evaluation length is 16,384 tokens for the 1.7B and 4B models. For Tab. 2, we select the final checkpoint at step 400 for evaluation and report all benchmark scores from that checkpoint.

Competitive coding. For competitive coding, we train `DeepSeek-R1-Distill-Qwen-1.5B` (Guo et al., 2025) on the ARCHERCODER training split (Wang et al., 2025b), which contains 6,753 problems. We use a global prompt batch size of 128 and sample 16 rollouts per problem. The maximum prompt length is 2,048 tokens, and the maximum response length is 32,768 tokens.

ISO-AdamW uses a learning rate of 1.5×10^{-6} for (U, V), followed by polar retraction after each factor update. For the weight-space AdamW baseline, we sweep

$$\left\{1 \times 10^{-6},\ 2 \times 10^{-6},\ 3 \times 10^{-6},\ 5 \times 10^{-6}\right\},$$

where 1×10^{-6} is the learning rate used by the original Archer-style recipe. AdamW with a learning rate of 1×10^{-6} performs strictly worse and is omitted from the figure for clarity.

We train the primary runs for 200 actor updates. At a prompt batch size of 128, this budget corresponds to approximately

$$\frac{200 \times 128}{6753} \approx 3.79$$

nominal passes over the prompt set before accounting for online filtering. Because the coding dataset is relatively small, longer training can enter a multi-pass overfitting regime. To test whether the primary budget truncates the weight-space baseline prematurely, we continue the two strongest AdamW runs, with learning rates 3×10^{-6} and 5×10^{-6}, to 260 actor updates.

For evaluation, we report results on LIVECODEBENCH (Jain et al., 2024) v5 and v6. We sample 8 responses per problem using temperature 0.8 and top-$p = 1.0$, and report average accuracy under the same evaluation protocol for all methods.

I.2 MEMORY OVERHEAD OF ISO-ADAMW

For a weight matrix $W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, ISO-AdamW uses the fixed-spectrum parameterization $W = U\Sigma_0 V^\top$, where $U \in \mathbb{R}^{d_{\text{out}} \times q}$, $V \in \mathbb{R}^{d_{\text{in}} \times q}$, $\Sigma_0 \in \mathbb{R}^{q \times q}$, and $q = \min(d_{\text{out}}, d_{\text{in}})$. The factors satisfy $U^\top U = V^\top V = I_q$. ISO-AdamW maintains AdamW's first- and second-moment buffers for U and V rather than for W; the fixed spectrum Σ_0 has no associated optimizer state. We first quantify the intrinsic overhead of these moment buffers, independently of how they are distributed across devices.

Intrinsic overhead. For each transformed matrix, AdamW stores $2d_{\text{out}}d_{\text{in}}$ moment elements, whereas ISO-AdamW stores $2q(d_{\text{out}} + d_{\text{in}})$. When both optimizers use the same precision for their moment buffers, the moment-memory ratio is

$$R_{\text{moment}} = \frac{2q(d_{\text{out}} + d_{\text{in}})}{2d_{\text{out}}d_{\text{in}}} = 1 + \frac{\min(d_{\text{out}}, d_{\text{in}})}{\max(d_{\text{out}}, d_{\text{in}})} \in (1, 2]. \tag{110}$$

The worst-case ratio of $2\times$ is attained only for square matrices. For `Qwen3-4B-Base`, the per-matrix ratios are approximately 1.63 for q/o projections, 1.40 for k/v projections, and 1.26 for gate/up/down projections. Aggregating over the 252 ISO-transformed matrices and the remaining parameters—normalization parameters and the tied embedding, which use plain AdamW in both configurations—the model-wide FP32 moment memory increases from 32.2 GB to 42.2 GB, an increase of 10.0 GB or approximately 31%. For this parameter scope and moment precision, the model-wide ratio is independent of sharding and offloading, and we report it as our primary moment-memory metric.

Per-device cost under the training configuration. All optimizer states are sharded by fully sharded data parallelism, so the intrinsic overhead maps to a much smaller per-accelerator increment. Under the identical 16-way FSDP configuration (2 nodes $\times$ 8 H100 80 GB) used for both AdamW and ISO-AdamW in Fig. 6, the +10.0 GB model-wide overhead corresponds to +0.63 GB per device (Tab. 11). The two quantities answer complementary questions: the model-wide ratio measures the algorithmic storage cost, the per-device increment determines practical deployability; we report both.

Our implementation additionally holds U and V in FP32 as optimizer state ($q(d_{\text{out}} + d_{\text{in}})$ elements per matrix; 19.6 GB model-wide for Qwen3-4B, i.e. 1.2 GB per device at 16-way sharding), and writes

Table 11: Optimizer moment memory for Qwen3-4B-Base under AdamW and ISO-AdamW (FP32). The 16-way row is the configuration of the runs in Fig. 6.

Metric	AdamW	ISO-AdamW	ISO overhead
Model-wide FP32 moment buffers	32.2 GB	42.2 GB	+10.0 GB (+31%)
Per device, 16-way FSDP (our runs)	2.01 GB	2.64 GB	+0.63 GB (0.8% of 80 GB)

the reconstructed $W = U\Sigma_0 V^\top$ back into the model parameters after every update. This storage is an implementation convenience rather than an intrinsic requirement, for two reasons. (1) *The factors are a deterministic function of the weights.* Since Σ_0 is fixed, U and V can be recovered from W by an SVD at the start of each update, exactly as they are obtained once at initialization. The cost of this SVD is small relative to an RL step: computing it for all 252 matrices of Qwen3-4B takes 26 s on the 16-GPU layout of our runs, compared to a mean RL step of roughly one hour; the paired sign ambiguity of singular vectors is resolved by a fixed sign convention. (2) *They are ordinary optimizer state and can be offloaded.* Like the moment buffers, U and V live in the optimizer's state dictionary and are handled by the standard optimizer-state offloading of the training framework, which moves all optimizer tensors to host memory between updates. With offloading enabled, no ISO-specific memory is resident on the accelerator outside the optimizer update itself.

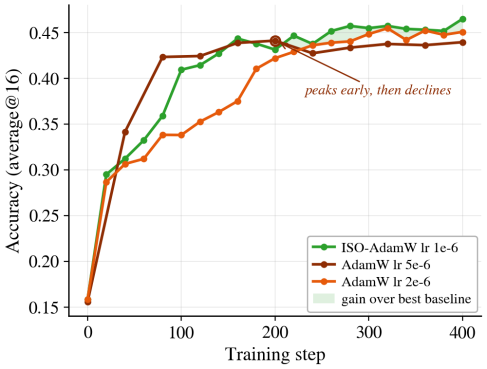


Figure 15: **AdamW with learning rate of 5e-6 on `Qwen3-4B-Base`.**

Measured peak memory. In the two runs of Fig. 6 (no offloading), the per-GPU peak allocated memory reported by the actor workers is 80.8 GB for AdamW and 82.6 GB for ISO-AdamW (+1.8 GB), consistent with the +0.63 GB of moment buffers plus the 1.2 GB of factors above.

Trade-off. The practical memory increment should be read alongside time-to-target. Under the same distributed configuration, ISO-AdamW reaches the final accuracy of the AdamW run (0.448 at 400 updates, 364 h) at update 260 after 240 h, a reduction of 124 wall-clock hours (34%). The realized trade-off is therefore +0.63 GB of moment memory per device (plus 1.2 GB of factor storage that can be recomputed or offloaded) for 124 fewer hours to matched accuracy. This efficiency gain does not erase the intrinsic 31% overhead of the moment buffers; it provides the context needed to evaluate the full resource trade-off.

I.3 More RLVR Results and Ablations

Math Results. As in Fig. 15, ISO-AdamW at lr 1×10^{-6} is compared with AdamW at its best-performing lr (2×10^{-6}) and at a larger lr (5×10^{-6}). The larger lr accelerates AdamW early but peaks around update 200 and then declines, so a higher learning rate does not close the gap to ISO-AdamW. Therefore, we omit the curve of AdamW 5×10^{-6} in Fig. 4 for clarity.

Table 12: **Coding results on DS-1.5B.** We report average@8 LiveCodeBench accuracy after 200 actor updates.

Method	LCB v5	LCB v6	Avg.
DS-1.5B	16.31	18.32	17.31
AdamW	24.60	25.00	24.80
ISO-AdamW	**24.64**	**26.53**	**25.58**

Coding Results. In Tab. 12, we report the average@8 accuracy across LiveCodeBench v5 and v6 with the same 200 actor updates. The AdamW number is from the best AdamW run with a learning rate sweep over $\{1, 2, 3, 5\} \times 10^{-6}$. ISO-AdamW achieves an average@8 of 25.58, versus 24.80 for the strongest AdamW baseline.

ISO-Muon variant. To test whether ISO's benefits extend beyond AdamW, we instantiate ISO-Muon by applying the Muon update in fixed-spectrum Stiefel coordinates. On `Qwen3-4B-Base`, we use a learning rate of 2.5×10^{-5} for ISO-Muon and sweep weight-space Muon over $\{2.5 \times 10^{-5}, 5 \times 10^{-5}, 7.5 \times 10^{-5}\}$ (Fig. 17). Although Muon exhibits a late-stage accuracy decline at

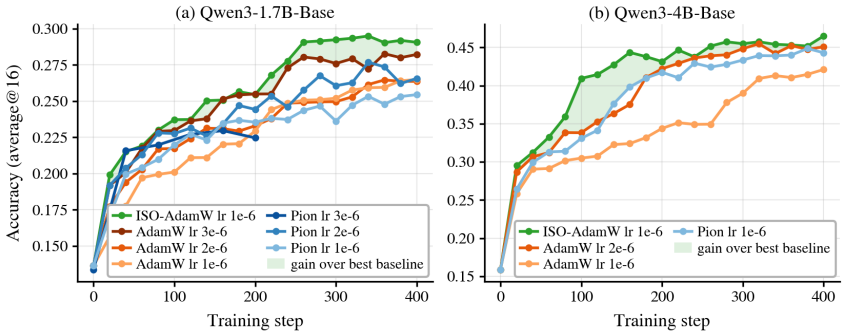


Figure 16: **ISO-AdamW vs. Pion vs. AdamW on `Qwen3-1.7B-Base` (left) and `Qwen3-4B-Base` (right).**

the largest learning rate 7.5×10^{-5}, ISO-Muon's advantage also holds against the strongest tested Muon run: it reaches that run's 400-update accuracy by update 300 and achieves a higher final aggregate score at update 400 (46.2 versus 45.5). These results extend the improvements observed with ISO-AdamW to a second base optimizer.

Comparison with recent optimizer Pion (Shi et al., 2026). Pion (Shi et al., 2026) is a concurrent spectrum-preserving optimizer. To compare Pion and ISO fairly, we port its official implementation into our pipeline (bit-exact to the release), matching training data, rollout settings, and the training recipe. On Qwen3-1.7B-Base, we sweep Pion's learning rate over $\{1, 2, 3\} \times 10^{-6}$, stopping the 3×10^{-6} run early following an accuracy decline beginning at step 160. On Qwen3-4B-Base, we use its published RLVR learning rate of 1×10^{-6} without additional tuning. Fig. 16 shows that, among completed 400-step runs, Pion underperforms the best tested AdamW configuration in both settings, while ISO achieves the highest final accuracy. On 4B, ISO also matches Pion's final accuracy with $\approx 2.5\times$ fewer actor updates (160 vs. 400) and finishes 2.2 percentage points higher.

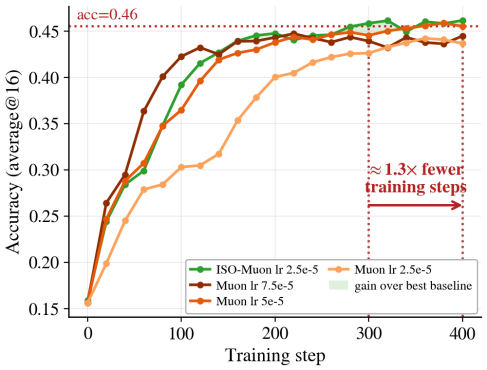


Figure 17: **ISO-Muon versus Muon on `Qwen3-4B-Base`.**

Does learning the spectrum help? As shown in Fig. 18, we compare the fixed- and trainable-spectrum ISO-AdamW variants on `Qwen3-1.7B-Base`. Despite the additional freedom for trainable-spectrum, the two variants exhibit nearly identical learning curves, reaching aggregate accuracies of 29.4 and 29.6 at update 400 for fixed and trainable Σ, respectively.

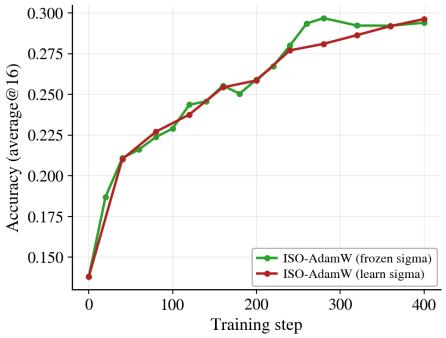


Figure 18: **ISO with Frozen vs. trainable Σ on `Qwen3-1.7B-Base`.**

Wall-clock cost and optimizer overhead details. In Fig. 6 (left), we measure the complete `optimizer.step()` call for `Qwen3-4B-Base`. For ISO-AdamW, the measured call includes all-to-all communication to assemble full matrices, factor-gradient construction, AdamW updates to both frames, polar retraction, weight reconstruction, all-to-all redistribution of the updated shards, and plain AdamW updates to normalization and embedding parameters. It excludes forward/backward computation, gradient clipping, and gradient clearing, which occur outside `optimizer.step()` within the actor-update stage. The one-time initialization SVD takes approximately 26 s and is reported separately. We average ten measured optimizer steps, obtaining 49.2 s for ISO-AdamW and 15 ms for AdamW. For the training-step comparison in Fig. 6

(left), ISO-AdamW's mean logged training-step time over 400 indices is $3{,}958\,\mathrm{s}$; normalizing the optimizer-step latency by this mean gives $49.2/3{,}958 \approx 1.2\%$.

In Fig. 6 (right), the cumulative-time curves sum the actual runs' `timing_s/step` measurements. The trainer's step timer covers rollout generation (including repeated DAPO sampling rounds), reward computation, old-policy and reference-policy log-probability computation, advantage computation, and the actor update.

J Related Work

RLVR dynamics and optimization. Most advances in RLVR have focused on data and environments , learning objectives, or systems infrastructure. By contrast, the optimization layer that translates reward feedback into parameter-space motion remains comparatively underexplored and is typically inherited from pre-training, including the choice of optimizer and parameterization. Recent analyses have begun to distinguish RLVR from pre-training and supervised fine-tuning at both the policy and parameter levels. At the policy level, RLVR has been characterized through KL-proximal and conservative policy-improvement views; at the parameter level, its updates have been reported to be sparse, off-principal, and strongly shared across runs initialized from the same source model (Wu et al., 2025; Shenfeld et al., 2025; Mukherjee et al., 2025; Zhu et al., 2025b). Building on the prior observation of limited spectral drift in RLVR (Zhu et al., 2025b), we formalize and test *spectral inheritance*: the base model's singular-value spectra can remain functionally reusable while the associated input and output frames adapt. We then use this principle to guide RLVR optimization algorithm design.

Matrix-geometric optimization and constrained parameterizations. A growing line of work treats matrix-valued parameters and updates as structured objects rather than as flattened Euclidean vectors. These methods act on different objects and impose different forms of structure. Muon orthogonalizes matrix-valued momentum updates, while DION develops scalable distributed approximations to such orthonormalized updates (Liu et al., 2025a; Ahn et al., 2025). Hyperball constrains the Frobenius norms of both weight matrices and optimizer updates, making the relative angular step size explicit rather than controlling it indirectly through weight decay (Wen et al., 2025). The Modular Manifolds perspective advocates co-designing module-specific constraints and update norms, including Stiefel-constrained weights and a manifold version of Muon (Bernstein, 2025). POET and POET-X preserve weight spectra through two-sided orthogonal-equivalence reparameterizations, with an emphasis on stable and efficient LLM training (Qiu et al., 2026a;b). Related structure has also been used for parameter-efficient adaptation: Spectral Adapter modifies a leading spectral subspace, while StelLA learns orthonormal input and output subspaces for a low-rank adapter (Zhang & Pilanci, 2024; Li et al., 2025).

Comparison with concurrent spectrum-preserving optimization. An independent concurrent work, Pion (Shi et al., 2026), also preserves weight spectra through left- and right-orthogonal transformations. Pion updates each weight matrix directly through a dedicated Lie-algebra-based rule with its own momentum, scale control, and approximate exponential-map machinery, whereas ISO exposes the singular frames as optimization variables and applies an existing base optimizer, such as AdamW or Muon, to those variables, followed by retraction. Thus, although both methods operate within spectrum-preserving weight families, they induce substantially different optimization dynamics.

The distinction is also empirical. We port Pion into the same RLVR pipeline and match the training data, rollout configuration, and evaluation protocol (Section 5.2). Across the completed Qwen3-1.7B and Qwen3-4B runs, the tested Pion configurations underperform the strongest tuned AdamW baseline, whereas ISO-AdamW achieves the highest final aggregate accuracy. On Qwen3-4B, ISO-AdamW reaches Pion's final accuracy with approximately $2.5\times$ fewer actor updates and finishes 2.2 percentage points higher. These results indicate that spectrum preservation alone does not determine optimization quality; how the fixed-spectrum geometry is parameterized and optimized is consequential in RLVR.

More broadly, Pion starts from spectrum preservation as a general LLM optimizer design principle, motivated by orthogonal-equivalence geometry and training stability. ISO instead starts from an RLVR-specific empirical and functional characterization: we identify *spectral inheritance*, calibrate

near-isospectrality against ambient dimensionality, and test reuse of the source spectrum through functional interventions and sequential RL stages. ISO then promotes this evidence into a broader RLVR post-training framework, with both an online optimizer transformation and an offline checkpoint-only expert-composition method.

From matrix-geometric priors to an RLVR-native post-training stack. Our contribution is not the first use of orthogonality, manifolds, or spectral constraints. Rather, ISO follows an evidence-to-design route tailored to RLVR. We first establish spectral inheritance through calibrated checkpoint analysis and functional interventions, and then use it as an inductive bias for two complementary stages of RLVR post-training. Online, ISO transforms conventional base optimizers to operate in fixed-spectrum frame coordinates; offline, the same geometry provides a shared coordinate system for checkpoint-only composition of RL experts. Thus, ISO-Optimizer and ISO-Merger are two realizations of the same spectral-inheritance principle rather than independent algorithms.

In summary, ISO contributes an evidence-to-stack design: it identifies spectral inheritance from RLVR's own dynamics, validates its functional relevance, and turns it into a common optimization principle spanning online policy learning and offline expert consolidation.

Geometry-aware model merging. Most training-free merging methods combine Euclidean task vectors, with techniques such as Task Arithmetic, TIES, and TSV-Merge reducing interference through sign resolution or low-rank structure (Ilharco et al., 2022; Yadav et al., 2023; Gargiulo et al., 2025). OrthoMerge instead merges orthogonal fine-tuning transformations in a Lie algebra and extends to generally fine-tuned models by separating orthogonal and residual components (Yang et al., 2026). ISO-Merger is specialized to shared-base RL experts: it projects each expert to the source-spectrum geometry, represents its frame displacement in a common base-anchored Stiefel tangent space, and performs checkpoint-only composition without post-merge prompts, rollouts, teacher queries, or optimization.