PASP: POSITIVE-ANCHORED SPECTRAL PROJECTION FOR RLVR

Anonymous authors
Paper under double-blind review

ABSTRACT

Muon has improved large language model pretraining, and Pion extends its spectral approach to reinforcement learning with verifiable rewards (RLVR). We identify a spectral asymmetry: positive-advantage gradients concentrate in a low-dimensional subspace and have lower effective rank than negative-advantage gradients. Muon and Pion transform these gradients after combining them into momentum, without separately controlling the two channels. This motivates PASP (Positive-Anchored Spectral Projection), which uses the current positive gradient to define an anchor subspace at each step. PASP jointly transforms the full positive gradient and the part of the negative gradient within this subspace. It separately transforms a low-rank approximation of the remaining negative gradient and adds the two outputs to form the update. We evaluate PASP on Qwen3 models against AdamW, Muon (hidden-only), and Pion. PASP achieves the highest mean accuracy across four reasoning benchmarks on both Qwen3 models. On Qwen3-8B, it reaches 0.489 ± 0.006, compared with 0.470 ± 0.015 for Pion, where $\pm$ denotes the sample standard deviation over three training seeds. On Qwen3-1.7B, the 8B recipes are transferred without retuning. All three PASP runs complete training, averaging 0.463, compared with 0.398 for Muon and 0.184 for Pion. Under DAPO, PASP reaches 0.485, compared with 0.466 for Muon and 0.468 for Pion. It also remains competitive on GPQA-Diamond and AMC23-Chain4, although the gains vary across settings. These results support using advantage sign to guide spectral optimization in the tested RLVR settings.

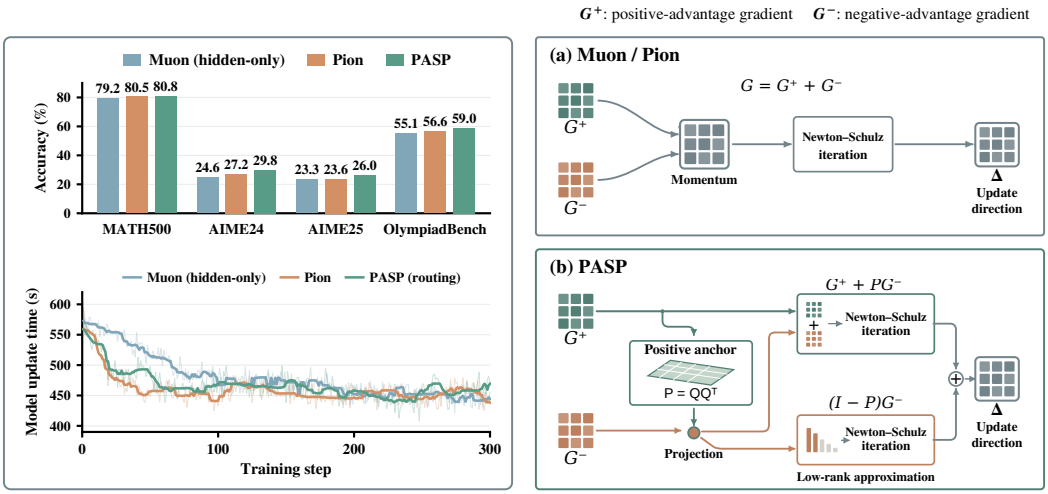


Figure 1: **Accuracy, update time, and positive anchoring. Left:** Qwen3-8B/GRPO accuracy at step 300 (means over three seeds, AIME24/25 avg@32) and model-update time (backward passes and optimizer step, excluding rollout). Thin curves show per-step timings, thick curves 11-step rolling medians. PASP uses three-pass for accuracy and sign routing for timing (Appendix C). **Right:** (a) Muon/Pion transform combined momentum. (b) PASP constructs the positive anchor $P = QQ^\top$ from the leading left-singular directions of G^+, jointly transforms $G^+ + PG^-$, and separately transforms a low-rank approximation of $(I-P)G^-$. The branch outputs sum to Δ (Section 5).

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.

1 INTRODUCTION

Muon uses the singular-value structure of momentum matrices to construct updates and has improved large language model pretraining (Jordan et al., 2024; Liu et al., 2025). Pion adapts this approach to post-training with a high-pass transformation that preserves dominant singular directions and suppresses weaker ones (Fan et al., 2026). Reinforcement learning with verifiable rewards (RLVR) trains language models using automatically checked rewards (Shao et al., 2024; DeepSeek-AI et al., 2025; Yang et al., 2025). Its policy-gradient contributions are weighted by advantages, which measure performance relative to a baseline. Positive advantages encourage an output, whereas negative advantages discourage it. However, Muon and Pion apply their spectral transformations after these contributions have been combined into momentum (Fig. 1(a)), so the transformations do not explicitly distinguish positive- and negative-advantage contributions. This motivates examining the spectral structure of the two contributions before they are combined.

We identify a spectral asymmetry between these contributions: the positive-advantage gradient G^+ concentrates in a low-dimensional subspace, while the negative-advantage gradient G^- has higher effective rank. Effective rank measures spectral concentration, with lower values indicating that fewer directions carry most of the gradient's energy. In the main Qwen3-8B experiment, the negative-to-positive ratio of mean effective ranks ranges from 21.2 to 41.3 across ten checkpoints. The same ordering appears across five advantage estimators and on Llama-3.1-8B and Qwen3-1.7B (Fig. 2). At Qwen3-8B step 50, the eight leading left singular vectors of the combined gradient retain 99.3% of the positive-gradient energy but only 19.0% of the negative-gradient energy, averaged over the probed layers (Sec. 4). After aggregation, a spectral transformation acts on the combined spectrum without separately controlling the two channels. Their different measured spectra motivate using the positive gradient to define an anchor for separating the negative contribution before spectral shaping.

We introduce PASP (Positive-Anchored Spectral Projection), which uses the concentrated positive gradient to define an anchor subspace at each step (Fig. 1(b)). With projector P onto this subspace, the negative gradient separates into PG^- within the subspace and $(I-P)G^-$ outside it. It jointly transforms the full G^+ and PG^-, and separately transforms a low-rank approximation of $(I-P)G^-$. This approximation retains a number of leading residual directions set by the anchor rank and a ratio α. The two outputs are added to form the update. Component ablations remove the sign split and residual truncation, and an anchor-source ablation compares the positive subspace with random, negative, and combined-gradient alternatives under a fixed rank rule (Sec. 6.3).

We evaluate PASP with 300 steps of mathematical RLVR training on models up to 8B parameters, using GRPO on Qwen3, Llama-3.1 (Grattafiori et al., 2024), and DeepSeekMath, and DAPO on Qwen3-8B. Evaluation covers four primary reasoning benchmarks plus GPQA-Diamond (Rein et al., 2024) and AMC23-Chain4. On Qwen3-8B, PASP reaches 0.489 ± 0.006 mean accuracy across the four primary benchmarks, compared with 0.470 ± 0.015 for Pion. Transferring the 8B recipes to Qwen3-1.7B without retuning gives 0.463 for PASP, compared with 0.398 for the stable Muon (hidden-only) baseline and 0.184 for Pion. PASP also achieves higher four-benchmark means than Muon and Pion on Llama-3.1-8B, DeepSeekMath-7B, and Qwen3-8B with DAPO (Sec. 6.2).

Our contributions are:

- **Spectral asymmetry between advantage channels.** We identify and characterize a persistent asymmetry in effective rank and spectral concentration between positive- and negative-advantage gradients in RLVR.
- **Positive-anchored spectral optimization.** We introduce PASP, which defines an adaptive anchor from the current positive-advantage gradient, jointly shapes the full positive gradient and the anchor-aligned negative component, and separately shapes a low-rank approximation of the orthogonal negative residual.
- **Experimental validation and sign routing.** PASP achieves the highest mean accuracy across the four primary benchmarks in both Qwen3/GRPO settings, with higher means than Muon and Pion on Llama-3.1-8B, DeepSeekMath-7B, and Qwen3-8B/DAPO. In a separate Qwen3-8B implementation study, sign routing reduces median step time by 38% relative to three-pass splitting, with full-run mean accuracy lower by 0.0088 (Sec. 6.3).

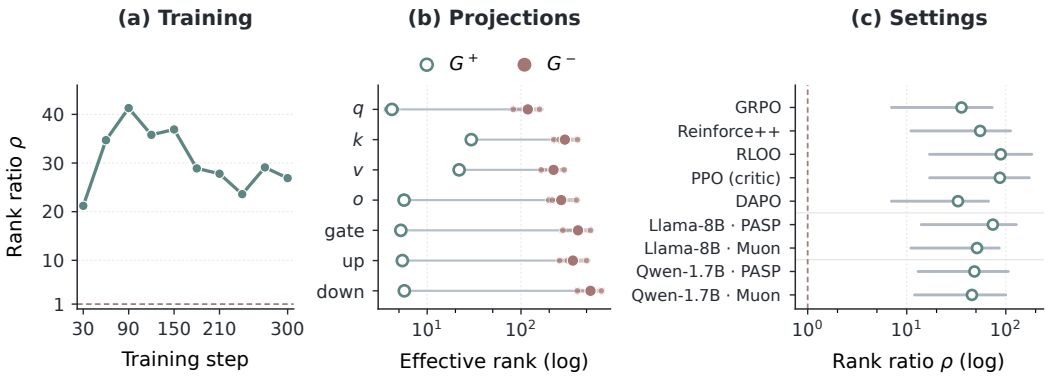


Figure 2: **Signed rank asymmetry across training, projections, and settings.** (a) The ratio of mean negative to mean positive effective rank ranges from 21.2 to 41.3 across ten checkpoints. Each channel is averaged over layers within each projection type, then over the seven types. (b) Teal and rose denote positive and negative channels. For each projection type, small points show layer-mean effective ranks at steps 30, 150, and 300. Large points show their three-checkpoint mean, and colored spans show the minimum and maximum. (c) Circles and gray bars show cross-layer means and observed layer ranges for five estimators and for PASP and Muon on Llama-3.1-8B and Qwen3-1.7B. Dashed reference lines mark equal effective rank ($\rho = 1$).

2 Related Work

Matrix-aware and spectral optimization. Shampoo and SOAP use structured preconditioners based on gradient statistics (Gupta et al., 2018; Vyas et al., 2024). Muon uses Newton–Schulz iterations to approximate the polar factor of a momentum matrix, bringing its nonzero singular values toward a common scale (Jordan et al., 2024; Liu et al., 2025). Pion addresses spectral failures in post-training through a high-pass transformation that preserves dominant singular directions and suppresses weaker ones (Fan et al., 2026). Both methods transform momentum after the gradient contributions have been combined. PASP instead uses advantage sign to construct the transformation inputs, with a positive-gradient anchor separating the negative contribution into aligned and residual components.

Low-rank and subspace optimization. GaLore reduces optimizer-state memory by projecting gradients into a low-rank subspace and computing updates there (Zhao et al., 2024). PASP uses a subspace to separate the negative-gradient contribution before spectral shaping. Its joint branch retains the full positive gradient, while only the orthogonal negative residual is approximated at low rank. Randomized range finding provides an efficient way to construct these subspaces without a full singular value decomposition (Halko et al., 2011).

RLVR gradients and optimization. RLVR methods improve advantage estimation, policy objectives, and experience reuse (Shao et al., 2024; Yu et al., 2025; Bai et al., 2026). NTHR reduces penalties on negative tokens that interfere with learning correct responses (Deng et al., 2025). Negative Sample Reinforcement (NSR) shows that learning from negative responses can improve reasoning while preserving generation diversity (Zhu et al., 2025b). Other studies examine parameter sparsity and optimization bias in RL updates (Mukherjee et al., 2026; Zhu et al., 2025a). These properties differ from spectral concentration within gradients separated by advantage sign. ResRL constructs a low-rank subspace from positive-token representations and uses negative-token projection residuals to reweight advantages (Lin et al., 2026). PASP acts on parameter-gradient matrices: it uses the positive gradient to split the negative gradient and applies separate spectral transformations without changing the RL objective.

3 Preliminaries

Signed RLVR gradients. For a hidden-layer weight matrix $W_t \in \mathbb{R}^{m\times n}$ at step t, split the policy-gradient loss into positive- and negative-advantage token contributions, $\mathcal{L}_t^{\mathrm{PG}} = \mathcal{L}_t^{+} + \mathcal{L}_t^{-}$, using the same loss normalization. The corresponding gradients are

$$G_t^{+} = \nabla_{W_t}\mathcal{L}_t^{+}, \qquad G_t^{-} = \nabla_{W_t}\mathcal{L}_t^{-}, \qquad G_t = \nabla_{W_t}\mathcal{L}_t^{\mathrm{PG}} = G_t^{+} + G_t^{-}. \tag{1}$$

The superscripts indicate advantage sign, not the signs of individual matrix entries. With a regularizer $\mathcal{R}_t$, the full loss gradient is $H_t = G_t + \nabla_{W_t}\mathcal{R}_t$.

Spectral optimization. For a compact singular value decomposition $X = U\Sigma V^\top$, a spectral transformation changes the singular values in Σ while preserving the associated directions in U and V. We write the five-step Newton–Schulz map with coefficients $\vartheta = (a, b, c)$ as

$$\begin{aligned} X^{(0)} &= A/(\|A\|_F + \epsilon), \\ X^{(k+1)} &= aX^{(k)} + b\big(X^{(k)}X^{(k)\top}\big)X^{(k)} + c\big(X^{(k)}X^{(k)\top}\big)^2X^{(k)}, \quad k = 0, \dots, 4, \\ \Phi_\vartheta(A) &= X^{(5)}. \end{aligned} \tag{2}$$

Here $\|\cdot\|_F$ is the Frobenius norm and $\epsilon > 0$ stabilizes normalization. Each iteration maps a singular value σ to $a\sigma + b\sigma^3 + c\sigma^5$ while preserving its associated singular vectors.

Muon applies this map to momentum, which accumulates gradients over training steps. With $\vartheta_{\mathrm{M}} = (3.4445, -4.7750, 2.0315)$, the map approximates the polar factor $UV^\top$, which replaces nonzero singular values by one (Jordan et al., 2024; Liu et al., 2025). Pion combines promotion steps, which move singular values toward one, with suppression steps, which attenuate small singular values. Together, they form a high-pass map that preserves dominant singular directions and suppresses weaker ones (Fan et al., 2026). Both methods transform momentum after the positive- and negative-advantage contributions have been combined.

4 Signed Rank Asymmetry in RLVR

Spectral concentration. We examine the signed gradients from Section 3 before aggregation. For a nonzero matrix A, we measure spectral concentration by the effective rank (Davis & Drusvyatskiy, 2025) and compare the channels through

$$r_{\mathrm{eff}}(A) = \frac{\|A\|_*^2}{\|A\|_F^2}, \qquad \rho = \frac{r_{\mathrm{eff}}(G^-)}{r_{\mathrm{eff}}(G^+)}. \tag{3}$$

Here $\|A\|_*$ is the sum of singular values. Lower effective rank indicates greater spectral concentration. The ratio ρ is unchanged by rescaling either channel, so it compares spectral shape independently of gradient magnitude. All measurements use policy gradients before the optimizer transformation.

The asymmetry persists across training and settings. On Qwen3-8B/GRPO, the ratio of mean effective ranks ranges from 21.2 to 41.3 across ten checkpoints from step 30 to step 300 (Fig. 2a). We average each channel's effective ranks over layers and then over the seven projection types before taking the ratio. At steps 30, 150, and 300, the cross-layer mean effective rank is higher for the negative channel in all 21 comparisons (Fig. 2b). Across five advantage estimators, including PPO with a learned critic, and the Llama-3.1-8B and Qwen3-1.7B probes, cross-layer mean ratios range from approximately 33 to 89. Every reported layer range lies above one (Fig. 2c, Appendix E.2).

Energy within the combined gradient's leading subspace. To complement the rank comparison, we measure how both channels align with the same subspace of their sum. Let U_8 contain the eight leading left singular vectors of $G = G^+ + G^-$ and set $P_8 = U_8U_8^\top$. The retained energy fractions are

$$E_\pm^{(8)} = \frac{\|P_8G^\pm\|_F^2}{\|G^\pm\|_F^2}.$$

Across the probed layers at Qwen3-8B step 50, the means are 99.3% for the positive channel and 19.0% for the negative channel. Thus, this combined-gradient subspace retains nearly all positive energy, while most negative energy lies outside it.

Implication for optimizer design. Muon and Pion apply their spectral maps to combined momentum, without separately controlling the two advantage channels. The measured spectral differences motivate using the concentrated positive gradient as a reference for organizing the negative contribution before spectral shaping.

5 POSITIVE-ANCHORED SPECTRAL PROJECTION

PASP uses the current positive gradient to define an anchor subspace (Fig. 1(b)). This subspace determines which negative components are combined with the full positive gradient and which are treated separately.[1]

Positive anchor and negative split. Let $Q_t = \mathcal{Q}_{r_t^+}(G_t^+)$ be an orthonormal basis approximating the leading r_t^+-dimensional left-singular subspace of G_t^+, computed by randomized range finding (Halko et al., 2011). The projector $P_t = Q_t Q_t^\top$ separates the negative gradient into

$$G_t^{\parallel} = P_t G_t^-, \qquad G_t^{\perp} = (I - P_t) G_t^-, \qquad G_t^- = G_t^{\parallel} + G_t^{\perp}. \tag{4}$$

The full positive gradient and $G_t^{\parallel}$ are added before transformation, allowing their contributions within the anchor to reinforce or cancel. A low-rank approximation retains the leading directions of $G_t^{\perp}$, preserving negative contributions outside the anchor while limiting their number.

Two-branch update. Write $\Pi_r(A) = \mathcal{Q}_r(A)\mathcal{Q}_r(A)^\top A$ for the randomized low-rank approximation. The update direction is

$$\Delta_t = \overline{\Phi}\Big(G_t^+ + G_t^{\parallel}\Big) + \overline{\Phi}\Big(\Pi_{r_t^-}(G_t^{\perp})\Big). \tag{5}$$

Both branches use the five-step map $\Phi \equiv \Phi_{\vartheta_{\mathrm{P}}}$ from Section 3, with Pion's promotion coefficients $\vartheta_{\mathrm{P}} = (\frac{15}{8}, -\frac{10}{8}, \frac{3}{8})$ (Fan et al., 2026). The guarded map $\overline{\Phi}$ returns the input if its transformed norm falls below a numerical threshold (Appendix A).

Adaptive ranks. We set the anchor rank from the rounded stable rank, $s(A) = \|A\|_F^2 / \|A\|_2^2$ (Rudelson & Vershynin, 2007). The rank is clipped to $[8, 64]$ and capped by the smaller matrix dimension. Estimating the largest singular value $\|A\|_2$ by power iteration avoids the full spectrum needed for effective rank. The residual rank scales with the anchor rank through α, which defaults to one. Both ranks and the anchor are recomputed at every step (Appendix A.3). Section 6.3 tests the positive anchor against alternative subspaces under the same rank rule.

Geometric properties. The negative components satisfy $\langle G_t^{\parallel}, G_t^{\perp} \rangle_F = 0$. Because the full G_t^+ enters the joint branch, the two transformed branches need not be orthogonal. In exact arithmetic, the transformed residual $\Phi(\Pi_{r_t^-}(G_t^{\perp}))$ has rank at most r_t^- and singular values at most one. If the anchor fully contains G_t^+ and the tiny-gradient fallback is inactive, $-\Delta_t$ is a descent direction for a nonzero update (Proposition A.2). For an approximate anchor, additional cross terms prevent an unconditional descent guarantee (Appendix A).

Measured spectral response. Figure 3 shows the spectra before and after shaping for Muon's combined momentum (a) and the two PASP branches in Eq. (5) (b). Shaping flattens the joint spectrum. Truncation limits residual shaping to eight directions outside the anchor, producing the sharp cutoff. At this frozen Qwen3-8B Muon checkpoint, the rank-8 positive anchor retains 75.8% of positive energy and 13.6% of negative energy. This anchor differs from the combined-gradient subspace measured in Section 4.

Algorithm 1 and Appendix A.6 specify the computation, matrix scaling, global RMS clipping, and Adagrad fallback.

[1] The equations use policy-gradient notation. With regularization, both the anchor and update use the regularized inputs $\widetilde{G}_t^+, \widetilde{G}_t^-$ defined in Appendix B.

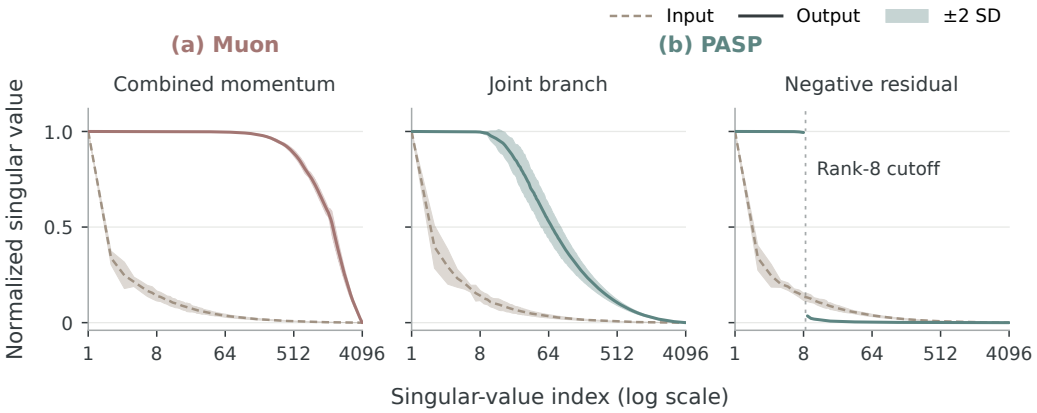


Figure 3: **Spectral responses at a frozen Qwen3-8B checkpoint.** (a) Muon's combined Nesterov momentum before and after its five-step map. (b) PASP's joint and residual branches, evaluated offline with $r_t^+ = r_t^- = 8$. The probe uses four rollout batches at step 260, layer-18 query projection. Each spectrum is normalized by its leading singular value. Curves are four-batch means, with ± 2 standard-deviation bands clipped at zero. The rank-8 approximation precedes residual shaping. The output retains the BF16 numerical tail.

6 Experiments

6.1 Experimental Setup

We compare SGD, AdamW (Loshchilov & Hutter, 2019), Muon, Pion (Fan et al., 2026), and PASP under GRPO (Shao et al., 2024) on Qwen3-8B, Qwen3-1.7B (Yang et al., 2025), Llama-3.1-8B, and DeepSeekMath-7B, and under DAPO (Yu et al., 2025) on Qwen3-8B. Runs start from math-SFT checkpoints and train on DAPO-Math-17k. Within each setting they share data, initialization, horizon, global batch, hardware, and evaluation; each optimizer keeps one disclosed recipe rather than a common absolute learning rate (Appendix D). Appendix E.5 applies the same $\{0.5\times, 1\times, 2\times\}$ grid and extends a baseline by one point when its best result is at the boundary. The primary suite is MATH500 (Lightman et al., 2023; Hendrycks et al., 2021), AIME24, AIME25, and OlympiadBench (He et al., 2024); AMC23-Chain4 and GPQA-Diamond are secondary and did not enter selection. Unless stated otherwise, $\pm$ denotes sample standard deviation across training seeds.

In the performance and cost comparisons, Muon denotes the hidden-only variant, which excludes embedding and output matrices from its spectral update. All-layer variants are labeled separately.

6.2 Main Results

PASP has the highest observed means in both Qwen3/GRPO comparisons. On Qwen3-8B, most of the gain appears on AIME24, AIME25, and OlympiadBench, where PASP improves over the strongest non-PASP result by 0.026, 0.022, and 0.024, respectively. On MATH500, Pion already reaches 0.805, while PASP reaches 0.808; the 0.003 difference lies within the reported descriptive range of decoding variability. This smaller margin is consistent with limited headroom at this accuracy level.

Qwen3-1.7B tests recipe transfer. Every optimizer keeps its Qwen3-8B recipe without retuning; for Pion, this recipe lies within 0.004 of its best tested Qwen3-8B learning rate (Appendix E.5). After the transfer, all three PASP runs complete training and PASP has the highest value on each primary benchmark. Hidden-matrix Muon also remains stable, whereas whole-matrix Pion completes but performs poorly on AIME24 and AIME25, both tested per-head Pion variants collapse, and all-layer Muon diverges. This comparison measures recipe transfer across scale, not a retuned optimum for each optimizer.

The observed advantage is directionally consistent across runs. Across 15 same-seed comparisons in five settings, PASP is higher than the strongest baseline 13 times, ties once, and is lower once. Because seeds are nested within settings and each run is evaluated once, these counts are descriptive and mix training with decoding variation; Appendix G.1 reports per-setting intervals.

Table 1: **Main results across experimental settings.** Entries are means over available training seeds. Both Qwen3-8B per-head Pion settings use three seeds; each Qwen3-1.7B per-head setting uses one run. Appendix G reports run-level summaries; *diverges* and *collapses* are defined in Appendix D. Bold marks the best result in each setting, and Improvement is against the strongest non-PASP entry. †Diagnostic; Muon is conventionally not applied to embeddings and the output layer.

Model / RL algorithm	Optimizer	MATH500	AIME24	AIME25	OlympiadBench
Qwen3-8B / GRPO	SGD	0.752	0.178	0.198	0.455
	AdamW	0.768	0.197	0.215	0.478
	Muon (hidden-only)	0.792	0.246	0.233	0.551
	Muon (all layers)†	*diverges*			
	Pion (whole-matrix)	0.805	0.272	0.236	0.566
	Pion (per-head, k_p=0)	*collapses*			
	Pion (per-head, k_p=2)	0.797	0.265	0.238	0.550
	PASP (ours)	**0.808**	**0.298**	**0.260**	**0.590**
	Improvement	+0.003	+0.026	+0.022	+0.024
Qwen3-1.7B / GRPO	SGD	0.630	0.155	0.170	0.390
	AdamW	0.643	0.165	0.180	0.401
	Muon (hidden-only)	0.741	0.189	0.192	0.472
	Muon (all layers)†	*diverges*			
	Pion (whole-matrix)	0.489	0.006	0.026	0.217
	Pion (per-head, k_p=0)	*collapses*			
	Pion (per-head, k_p=2)	*collapses*			
	PASP (ours)	**0.789**	**0.281**	**0.233**	**0.547**
	Improvement	+0.048	+0.092	+0.041	+0.075
Llama-3.1-8B / GRPO	SGD	0.474	0.016	0.005	0.193
	AdamW	0.488	0.013	0.011	0.188
	Muon (hidden-only)	**0.524**	0.020	0.022	0.250
	Pion	0.480	0.024	0.023	0.236
	PASP (ours)	0.514	**0.045**	**0.031**	**0.266**
	Improvement	−0.010	+0.021	+0.008	+0.016
DeepSeekMath-7B / GRPO	SGD	0.424	0.011	0.003	0.160
	AdamW	0.398	0.016	0.003	0.164
	Muon (hidden-only)	0.446	0.022	0.004	0.172
	Pion	0.446	0.025	0.006	0.188
	PASP (ours)	**0.464**	**0.030**	**0.011**	**0.200**
	Improvement	+0.018	+0.005	+0.005	+0.012
Qwen3-8B / DAPO	SGD	0.755	0.190	0.210	0.460
	AdamW	0.796	0.254	0.245	0.554
	Muon (hidden-only)	**0.804**	0.263	0.240	0.559
	Pion	0.798	0.269	0.235	0.570
	PASP (ours)	0.802	**0.283**	**0.281**	**0.573**
	Improvement	−0.002	+0.014	+0.036	+0.003

Secondary evaluations. On the same step-300 checkpoints, AMC23-Chain4 requires four dependent problems in one response (Tables 11 and 10). PASP reaches 0.086 versus 0.050 for Pion at 8B, and 0.049 versus 0.021 for hidden-only Muon at 1.7B. On GPQA-Diamond it leads at 8B (0.480 ± 0.008 versus 0.449 ± 0.021 for Pion) and ties Pion at 1.7B (0.253 versus 0.251). Neither evaluation entered selection.

The accuracy gap persists in the three-seed trajectory mean. From step 20 onward, PASP leads every baseline at each sampled checkpoint (Fig. 4).

Over the final 50 steps, PASP has the highest mean group reward, 0.106 versus 0.005 for Pion, without the largest actor KL or the longest responses (Fig. 5). These traces are descriptive, not a causal account of the accuracy ordering.

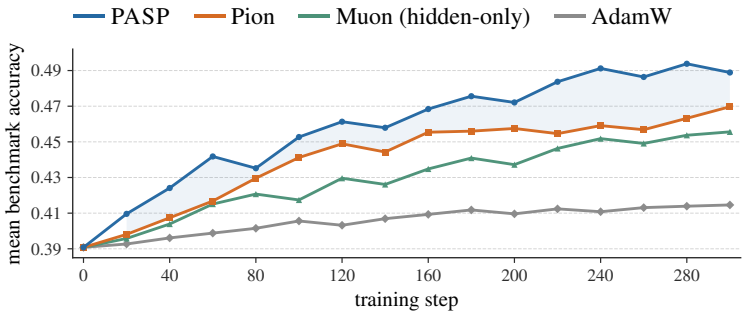
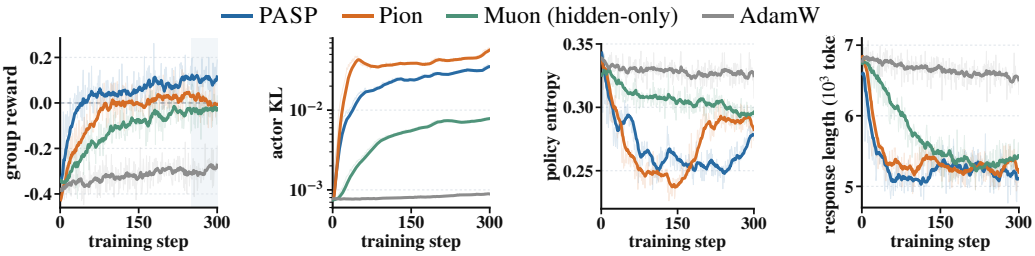


Figure 4: **Three-seed accuracy trajectories under the main-table evaluation protocol.** At each 20-step checkpoint, each benchmark is averaged over training seeds 42–44 ($n = 3$); the four benchmark means are then averaged with equal weight. All optimizers start from the same checkpoint. From step 20 onward, PASP leads at every sampled checkpoint and finishes at 0.489, compared with 0.470 for Pion, 0.456 for Muon, and 0.415 for AdamW.



(a) Group reward. (b) Actor KL. (c) Policy entropy. (d) Response length.

Figure 5: **Training dynamics across four optimizers at shared seed 43.** The panels report group reward, actor KL, policy entropy, and mean response length. Faint traces show raw logs and solid curves show exponential moving averages over the complete RLVR trajectory.

6.3 Ablations and Efficiency

The sign split provides the larger gain. The no-split control keeps PASP's polynomial coefficients, global RMS clipping, Adagrad fallback, actor learning rate, and training settings fixed. It only replaces the two-branch update with one spectral map applied to the combined gradient:

$$\Delta_t^{\text{no-split}} = \overline{\Phi}(G_t^+ + G_t^-).$$

This control reaches a four-benchmark mean of 0.459, close to hidden-only Muon at 0.456 and below whole-matrix Pion at 0.470 (Table 1). Restoring the signed two-branch construction raises the mean to 0.489. This 0.030 gap measures the complete signed construction, not advantage sign independently of the number of spectral maps. Removing residual truncation yields 0.466; both ablations are lower than full PASP on all four benchmarks.

Table 2: **PASP component ablation on Qwen3-8B.** Each benchmark entry averages training seeds 42–44 ($n = 3$). Values in parentheses are changes relative to full PASP; Mean gives the equal-weight average over the four benchmarks.

PASP variant	MATH500	AIME24	AIME25	OlympiadBench	Mean
PASP (full)	0.808	0.298	0.260	0.590	0.489
PASP w/o residual truncation	0.802 (−0.006)	0.268 (−0.030)	0.234 (−0.026)	0.562 (−0.028)	0.466
PASP w/o sign split	0.794 (−0.014)	0.250 (−0.048)	0.236 (−0.024)	0.554 (−0.036)	0.459

The positive anchor ranks first in both tested seeds. Table 3 compares four independently trained variants that differ only in the anchor basis. The positive anchor ranks first for both seeds and exceeds the random, negative, and aggregated means by 0.0177, 0.0187, and 0.0281. All variants pass the full G_t^+ into the joint branch; only the subspace partitioning G_t^- changes. The result supports positive anchoring under the fixed rank rule for these seeds; source-specific rank tuning and broader seed coverage remain open.

Table 3: **Anchor-source ablation on Qwen3-8B/GRPO.** The table reports independently trained matched-control runs. All variants use the same SFT initialization, training settings, 300-step budget, and rank rule set by G^+; only the source of the anchor basis changes. Mean averages seeds 42 and 43; bold marks the best result in each column.

Anchor source	Seed 42	Seed 43	Mean
Positive (G^+)	0.4948	0.4837	0.4893
Random, matched rank	0.4794	0.4637	0.4716
Negative (G^-)	0.4768	0.4643	0.4706
Aggregated ($G^+ + G^-$)	0.4674	0.4549	0.4612

Residual-rank sensitivity. Settings $\alpha \in \{1, 2, 5\}$ yield similar means (0.4917–0.4956), whereas $\alpha = 0.5$ collapses at step 90 and $\alpha = 10$ degrades to 0.4259 (Appendix E). Thus the retained negative structure requires control.

Sign routing and cost. Routing reduces median step time by 38%, trails three-pass accuracy by 0.0088, and matches Muon's actor-update time at higher memory cost (Appendix C).

7 Conclusion

Across the RLVR settings we probe, positive-advantage gradients have lower effective rank than negative-advantage gradients, and at the probed checkpoint the leading subspace of their sum retains 99.3% of the positive-channel energy but only 19.0% of the negative-channel energy. PASP anchors on the positive gradient, splits the negative gradient relative to that anchor, and shapes the two branches separately. Across models and RL algorithms, PASP leads in 13 of 15 same-seed comparisons, supporting advantage-sign-aware inputs to matrix-level spectral transformations.

AI use statement

We used generative AI tools to assist with implementing and debugging the optimizer and training code, checking the correspondence between the code and method definition, reviewing mathematical derivations and proofs, interpreting experimental results, editing the manuscript, and formatting figures and tables. We reviewed the AI-assisted work and take responsibility for all text, claims, code, and other artifacts. Appendix B documents the scope of the implementation checks, including a single-process total-gradient conservation test for routing.

Ethics statement

By improving spectral-optimization stability in the tested RLVR settings, PASP may reduce compute lost to failed runs, while single-backward routing lowers the cost of explicit sign decomposition. PASP is not uniformly cheaper than Muon because live range projections, a second spectral transformation, and channel accumulators retain runtime and memory overhead.

Like other methods that improve reasoning models, PASP may indirectly increase standard dual-use risks. It changes training optimization rather than model-behavior safety, so deployments still require appropriate evaluation and safeguards.

Reproducibility statement

Section 6.1 and Appendix D specify the shared training controls, optimizer-specific hyperparameters and distributed implementations, gradient-splitting assignments, and evaluation protocols. Appendix A describes rank estimation, randomized range finding, and numerical fallbacks. Appendix B states the conditions for the two-pass identity and one-pass routing, and distinguishes policy-gradient channel equivalence and total-gradient conservation from equality of the final nonlinear PASP direction. The reported checks include a single-process routing conservation test and an optional sampled two-pass diagnostic. Distributed routing attribution and equality of the final nonlinear update require separate validation. Appendices E and G report the configuration sweeps and independent runs, and

Appendix F states the scope of the comparison. The training code, verl patch, and paper-aligned recipes will be released upon publication.

REFERENCES

Fei Bai, Zhipeng Chen, Chuan Hao, Ming Yang, Ran Tao, Bryan Dai, Wayne Xin Zhao, Jian Yang, and Hongteng Xu. Towards effective experiential learning: Dual guidance for utilization and internalization, 2026. URL https://arxiv.org/abs/2603.24093.

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

Damek Davis and Dmitriy Drusvyatskiy. When do spectral gradient updates help in deep learning?, 2025. URL https://arxiv.org/abs/2512.04299.

DeepSeek-AI, Daya Guo, et al. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Wenlong Deng, Yi Ren, Muchen Li, Danica J. Sutherland, Xiaoxiao Li, and Christos Thrampoulidis. On the effect of negative gradient in group relative deep reinforcement optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/6adaf0cbeba11705d4ea67a62044f63d-Abstract-Conference.html.

Chongyu Fan, Gaowen Liu, Mingyi Hong, Ramana Rao Kompella, and Sijia Liu. Rethinking Muon beyond pretraining: Spectral failures and high-pass remedies for VLA and RLVR, 2026. URL https://arxiv.org/abs/2605.19282.

Aaron Grattafiori et al. The Llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Vineet Gupta, Tomer Koren, and Yoram Singer. Shampoo: Preconditioned stochastic tensor optimization. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pp. 1842–1850, 2018. URL https://proceedings.mlr.press/v80/gupta18a.html.

Nathan Halko, Per-Gunnar Martinsson, and Joel A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2):217–288, 2011. doi: 10.1137/090771806. URL https://arxiv.org/abs/0909.4061.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. OlympiadBench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 3828–3850, 2024. doi: 10.18653/v1/2024.acl-long.211. URL https://aclanthology.org/2024.acl-long.211/.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2021. URL https://arxiv.org/abs/2103.03874.

Keller Jordan, Yuchen Jin, Vlado Boza, Jiacheng You, Franz Cesista, Laker Newhouse, and Jeremy Bernstein. Muon: An optimizer for hidden layers in neural networks. https://kellerjordan.github.io/posts/muon/, 2024. URL https://kellerjordan.github.io/posts/muon/.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 3843–3857, 2022. doi: 10.52202/068431-0278. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/18abbeef8cfe9203fdf9053c9c4fe191-Abstract.html.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step, 2023. URL https://arxiv.org/abs/2305.20050.

Zihan Lin, Xiaohan Wang, Jie Cao, Jiajun Chai, Li Wang, Xiaodong Lu, Wei Lin, Ran He, and Guojun Yin. ResRL: Boosting LLM reasoning via negative sample projection residual reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2026. URL https://arxiv.org/abs/2605.00380.

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, 2025. URL https://arxiv.org/abs/2502.16982.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019. URL https://arxiv.org/abs/1711.05101.

Sagnik Mukherjee, Lifan Yuan, Pavan Jayasinha, Dilek Hakkani-Tür, and Hao Peng. Do we need Adam? surprisingly strong and sparse reinforcement learning with SGD in LLMs. In *International Conference on Machine Learning (ICML)*, 2026. URL https://arxiv.org/abs/2602.07729.

PyTorch Foundation. PyTorch 2.4 Release Blog. https://pytorch.org/blog/pytorch2-4/, 2024. URL https://pytorch.org/blog/pytorch2-4/. FSDP2: DTensor-based per-parameter-sharding FSDP.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof Q&A benchmark. In *Conference on Language Modeling (COLM)*, 2024. URL https://openreview.net/forum?id=Ti67584b98.

Mark Rudelson and Roman Vershynin. Sampling from large matrices: An approach through geometric functional analysis. *Journal of the ACM*, 54(4):21, 2007. doi: 10.1145/1255443.1255449. URL https://arxiv.org/abs/math/0503442.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

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 (EuroSys)*, pp. 1279–1297, 2025. doi: 10.1145/3689031.3696075. URL https://doi.org/10.1145/3689031.3696075.

Nikhil Vyas, Depen Morwani, Rosie Zhao, Mujin Kwun, Itai Shapira, David Brandfonbrener, Lucas Janson, and Sham Kakade. SOAP: Improving and stabilizing shampoo using Adam, 2024. URL https://arxiv.org/abs/2409.11321.

An Yang et al. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, et al. DAPO: An open-source LLM reinforcement learning system at scale, 2025. URL https://arxiv.org/abs/2503.14476.

Weihao Zeng, Yuzhen Huang, Qian Liu, Wei Liu, Keqing He, Zejun Ma, and Junxian He. SimpleRL-Zoo: Investigating and taming zero reinforcement learning for open base models in the wild. In *Conference on Language Modeling (COLM)*, 2025. URL https://arxiv.org/abs/2503.18892.

Jiawei Zhao, Zhenyu Zhang, Beidi Chen, Zhangyang Wang, Anima Anandkumar, and Yuandong Tian. GaLore: Memory-efficient LLM training by gradient low-rank projection. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235, pp. 61121–61143, 2024. URL https://proceedings.mlr.press/v235/zhao24s.html.

Hanqing Zhu, Zhenyu Zhang, Hanxian Huang, DiJia Su, Zechun Liu, Jiawei Zhao, Igor Fedorov, Hamed Pirsiavash, Jinwon Lee, David Z. Pan, Zhangyang Wang, Yuandong Tian, and Kai Sheng Tai. Why RL updates look sparse: An implicit compass drives optimization bias. In *NeurIPS 2025 Workshop on Efficient Reasoning*, 2025a. URL https://neurips.cc/virtual/2025/126674.

Xinyu Zhu, Mengzhou Xia, Zhepei Wei, Wei-Lin Chen, Danqi Chen, and Yu Meng. The surprising effectiveness of negative reinforcement in LLM reasoning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025b. URL https://arxiv.org/abs/2506.01347.

Appendix

Table of Contents

A Geometric Structure of the Positive-Anchored Update . . . 13
B Three Implementations of the Signed Channels . . . 16
C Computational Complexity . . . 17
D Experimental Details . . . 18
E Additional Results . . . 20
F Limitations . . . 23
G Per-Run Results . . . 24

A Geometric Structure of the Positive-Anchored Update

The geometric statements apply to the two matrices supplied to the optimizer. For regularized training, substitute $\widetilde{G}_t^+, \widetilde{G}_t^-$ for G_t^+, G_t^- throughout this appendix. Their sum, denoted by G_t in the statements below, is then the full loss gradient H_t.

A.1 Geometric Properties

Proposition A.1 (Orthogonal decomposition and full-rank recovery)**.** *For the orthogonal projector $P_t = Q_t Q_t^\top$, the components $G_t^{\parallel} = P_t G_t^-$ and $G_t^{\perp} = (I - P_t)G_t^-$ are orthogonal. If $P_t = I$, Eq. (5) reduces to $\overline{\Phi}(G_t)$.*

Proof. Because $P_t(I - P_t) = 0$,

$$\langle G_t^{\parallel}, G_t^{\perp} \rangle_F = \mathrm{Tr}\big((G_t^-)^\top P_t (I - P_t) G_t^-\big) = 0.$$

Hence $\|G_t^-\|_F^2 = \|G_t^{\parallel}\|_F^2 + \|G_t^{\perp}\|_F^2$. The positive channel enters Eq. (5) unprojected, so the decomposition discards no positive-channel energy. If $P_t = I$, then $G_t^{\perp} = 0$ and $G_t^{\parallel} = G_t^-$, and Eq. (5) becomes $\overline{\Phi}(G_t^+ + G_t^-) = \overline{\Phi}(G_t)$. □

A.2 Descent and Bounded Residual Contribution

Proposition A.2 (Descent and bounded residual contribution)**.** *Let $G_t = G_t^+ + G_t^-$, $A_t = G_t^+ + P_t G_t^-$, and $B_t = \Pi_{r_t^-}(G_t^{\perp})$, so that $\Delta_t = \overline{\Phi}(A_t) + \overline{\Phi}(B_t)$, and consider exact arithmetic with the tiny-gradient fallback inactive. The randomized range-finder statements hold with probability one over the Gaussian sketch. If the anchor contains the positive channel, $P_t G_t^+ = G_t^+$, then*

(i) $\langle G_t, \Delta_t \rangle_F = \langle A_t, \overline{\Phi}(A_t) \rangle_F + \langle B_t, \overline{\Phi}(B_t) \rangle_F$, and this quantity is positive unless $A_t = B_t = 0$;

(ii) whenever $\overline{\Phi}(B_t) = \Phi(B_t)$, every nonzero singular value of $\Phi(B_t)$ lies in $(0, 1]$, so $\|\Phi(B_t)\|_F^2 \le \mathrm{rank}(B_t) \le r_t^-$.

Proof. Let $p(x) = \frac{15}{8}x - \frac{10}{8}x^3 + \frac{3}{8}x^5$ be the scalar polynomial of ϑ_{P}. Since $p'(x) = \frac{15}{8}(1-x^2)^2 \ge 0$, $p(0) = 0$, and $p(1) = 1$, p maps $(0, 1]$ into $(0, 1]$. For $X \neq 0$ with compact singular value decomposition $X = U\Sigma V^\top$, the normalization in Eq. (2) places the singular values in $(0, 1)$, and each iteration keeps U and V while applying p to the singular values. Hence $\Phi(X) = U\psi(\Sigma)V^\top$ with $\psi(\sigma) = p^{\circ 5}\big(\sigma/(\|X\|_F + \epsilon)\big) \in (0, 1]$ for $\sigma > 0$. Therefore $\mathrm{range}\,\Phi(X) = \mathrm{range}\,X$ and $\langle X, \Phi(X) \rangle_F = \sum_i \sigma_i \psi(\sigma_i) > 0$. The guarded map returns either $\Phi(X)$ or X, so both properties also hold for $\overline{\Phi}$.

If $P_t G_t^+ = G_t^+$, then $A_t = P_t G_t$ and $G_t^{\perp} = (I - P_t)G_t$, so $G_t = A_t + G_t^{\perp}$ and $\mathrm{range}\,\overline{\Phi}(A_t) \subseteq \mathrm{range}\,P_t$. Write $R = G_t^{\perp}$ and $B_t = Q_t' Q_t'^\top R$. If $\mathrm{rank}(R) \ge r_t^-$, the Gaussian range sketch has rank r_t^- with probability one, so $\mathrm{range}\,Q_t' \subseteq \mathrm{range}\,R$. If $\mathrm{rank}(R) < r_t^-$, the sketch spans

Algorithm 1: Positive-Anchored Spectral Projection

Input: signed gradients G_t^+, G_t^-; rank ratio α; $r_{\min} = 8, r_{\max} = 64$; guarded map $\overline{\Phi}$
Output: spectral direction Δ_t
$r_t^+ \leftarrow \min\{d_t, \max\{r_{\min}, \min\{r_{\max}, \mathrm{round}(\mathrm{stable_rank}(G_t^+))\}\}\}$;
$r_t^- \leftarrow \min\{d_t, \max\{1, \mathrm{round}(\alpha r_t^+)\}\}$;
$Q_t \leftarrow \mathcal{Q}_{r_t^+}(G_t^+)$; $P_t \leftarrow Q_t Q_t^\top$; `// recomputed every step`
$G_t^{\parallel} \leftarrow P_t G_t^-$; $G_t^{\perp} \leftarrow (I - P_t) G_t^-$;
$\Delta_t \leftarrow \overline{\Phi}(G_t^+ + G_t^{\parallel}) + \overline{\Phi}(\Pi_{r_t^-}(G_t^{\perp}))$;
compute the global RMS clip c_t from Eq. (8) and apply the matrix-shape-scaled update $c_t \Delta_t$;

range R with probability one, and any orthonormal QR completion gives $B_t = R$. This also covers $R = 0$. Thus $\mathrm{range}\, B_t \subseteq \mathrm{range}(I - P_t)$ in both cases, even when the completed basis itself extends outside that space. Since the spectral map preserves column spaces, $\overline{\Phi}(B_t) = Q_t' Q_t'^\top \overline{\Phi}(B_t)$ and $\mathrm{range}\, \overline{\Phi}(B_t) \subseteq \mathrm{range}(I - P_t)$. The cross terms vanish, and

$$\langle G_t, \overline{\Phi}(A_t) \rangle_F = \langle A_t, \overline{\Phi}(A_t) \rangle_F, \qquad \langle G_t, \overline{\Phi}(B_t) \rangle_F = \langle Q_t' Q_t'^\top G_t^{\perp}, \overline{\Phi}(B_t) \rangle_F = \langle B_t, \overline{\Phi}(B_t) \rangle_F.$$

Adding the two identities gives (i), with each term positive when its argument is nonzero. Part (ii) follows because $\Phi(B_t)$ has $\mathrm{rank}(B_t) \le r_t^-$ nonzero singular values, each in $(0, 1]$. □

The hidden-matrix update in Eq. (9) multiplies $-\Delta_t$ by the positive factor $2\eta_t \sqrt{\max(m, n)}\, c_t$. The signed Adagrad direction in Eq. (11) satisfies $\langle G_t, U_t \rangle_F \ge 0$. Excluding decoupled weight decay and the tiny-gradient fallback, the combined parameter displacement therefore has a nonpositive inner product with the current loss gradient under exact containment. It is strictly negative whenever at least one shaped direction or fallback-parameter gradient is nonzero. This is a first-order statement and does not guarantee loss reduction at an arbitrary finite learning rate.

Inexact anchors. In practice, P_t retains the leading r_t^+ directions of G_t^+, and $E_t = (I - P_t) G_t^+$ need not vanish. The same calculation adds two cross terms to (i), $\langle G_t^{\perp}, (I - P_t) \overline{\Phi}(A_t) \rangle_F$ and $\langle E_t, \overline{\Phi}(B_t) \rangle_F$. When the residual guard returns $\Phi(B_t)$, the magnitude of the second term is at most $\sqrt{r_t^-}\, \|E_t\|_F$ by part (ii) and Cauchy–Schwarz. Both cross terms vanish when $E_t = 0$. Part (ii) does not require exact containment. For an inexact anchor, descent requires an additional bound on the sum of the cross terms.

A.3 Live Rank and Range Computation

$$\begin{aligned} d_t &= \min(m, n), \qquad \widehat{s}_t^+ = \frac{\|G_t^+\|_F^2}{\widehat{\sigma}_1(G_t^+)^2}, \\ r_t^+ &= \min\big\{d_t, \max\big\{r_{\min}, \min\big\{r_{\max}, \mathrm{round}(\widehat{s}_t^+)\big\}\big\}\big\}, \\ r_t^- &= \min\big\{d_t, \max\big\{1, \mathrm{round}(\alpha r_t^+)\big\}\big\}, \qquad Q_t = \mathcal{Q}_{r_t^+}(G_t^+), \qquad P_t = Q_t Q_t^\top. \end{aligned} \tag{6}$$

For each matrix, we estimate $\sigma_{\max}$ with three power iterations, round $\|G_t^+\|_F^2 / \sigma_{\max}^2$, and apply the bounds in Eq. (6) with $r_{\min} = 8$ and $r_{\max} = 64$. The range finder uses a Gaussian test matrix with no oversampling, one power iteration, and reduced QR. Both the positive basis and the residual basis are regenerated from the training RNG state at every optimizer step.

A.4 Numerical Boundary Conditions

The implementation makes two fallback cases explicit.

First, if $\|G_t^+\|_F < 10^{-9}$, we bypass rank estimation, both range projections, and both spectral maps, and return the raw direction $G_t^+ + 0.1 G_t^-$. The stable-rank ratio is undefined at zero and can be numerically unreliable near zero. Proposition A.2 does not cover this fallback. Second, for any

Table 4: **Spectral update design.** The optimizers differ not only in their spectral maps, but also in which gradient representation determines the shaping space.

Optimizer	Input	Geometry source	Uses advantage sign	Spectral treatment
Muon	combined momentum	combined spectrum	no	approximate whitening
Pion	combined momentum	combined spectrum	no	suppress spectral tail
PASP	signed gradients	current G_t^+ subspace	yes	truncate negative residual

input A, let $B = \Phi(A)$. The guarded map used in Eq. (5) is

$$\overline{\Phi}(A) = \begin{cases} A, & \|B\|_F \le 10^{-6}(\|A\|_F + 10^{-9}), \\ B, & \text{otherwise.} \end{cases} \tag{7}$$

Thus a numerically collapsed Newton–Schulz iteration cannot erase a nonzero channel. After these per-channel safeguards, the joint update is scaled by the default adaptive RMS clip in Eq. (8).

A.5 Spectral Update Design

Figure 3 reports the measured spectral responses in Section 5.

Table 4 compares the shaping inputs, the source of spectral geometry, and the treatment of spectral tails or residuals.

A.6 Update Scaling and Clipping

Before parameter-specific scaling, the implementation applies a single global RMS clip to all raw optimizer directions. For raw directions $\{U_{t,j}\}_{j=1}^{J}$ across the updated tensors, define

$$\begin{aligned} u_t &= \left(\frac{\sum_j \|U_{t,j}\|_F^2}{\sum_j \operatorname{numel}(U_{t,j})} \right)^{1/2}, \\ c_t &= \min\left(1, \frac{\max(\kappa \bar{u}_{t-1}, 10^{-12})}{u_t}\right), \qquad \bar{u}_t = \beta \bar{u}_{t-1} + (1-\beta) u_t \quad (u_t > 0). \end{aligned} \tag{8}$$

At $u_t = 0$, set $c_t = 1$ and leave the EMA unchanged. The EMA remains uninitialized until the first nonzero update, at which point $c_t = 1$ and $\bar{u}_t = u_t$. Equation (8) applies to subsequent nonzero updates. We use $\kappa = 3$ and $\beta = 0.95$ by default. This clipping is shared by the hidden-matrix PASP directions and the Adagrad directions for excluded parameters.

For a two-dimensional hidden weight, the implementation applies

$$W_{t+1} = (1 - \eta_t \lambda) W_t - \eta_t \, 2\sqrt{\max(m, n)} \, c_t \Delta_t. \tag{9}$$

Other parameters, including embeddings and output weights, use the signed Adagrad fallback specified in Appendix D.

A.7 Signed-Channel Implementations

Algorithm 1 summarizes the PASP update, with numerical boundary cases specified in Appendix A. We provide three implementations for constructing the signed gradient channels (Table 5). The three-pass reference implementation computes the positive and negative contributions through separate masked backward passes. The two-pass implementation uses one additional backward pass with absolute advantages. It recovers the reference channels when the policy gradient is linear in the signed advantage components, as specified in Appendix B.

Sign routing assigns each gradient increment to a channel without an additional backward pass. Exact policy-gradient attribution requires each routed increment to contain only one advantage sign, including contributions from other distributed ranks. This holds when routing precedes any reduction that mixes signs, or when all contributors to a reduced increment share the same sign. A locally

single-signed micro-batch alone is insufficient after a reduction that mixes signs. Shared regularizer gradients follow the routed increment's sign, whereas the reference implementation assigns them to the positive channel.

B Three Implementations of the Signed Channels

B.1 Setting and Assumption

Let the actor loss over a step's micro-batches $\{b_i\}$ decompose as $\mathcal{L} = \sum_i \ell_i(\theta; A) + \lambda\mathcal{R}$, where ℓ_i is the policy-gradient term and $\mathcal{R}$ is the shared KL/entropy regularizer. Write $A^+ = A\mathbb{1}[A > 0]$ and $A^- = A\mathbb{1}[A < 0]$, so $A = A^+ + A^-$ and $|A| = A^+ - A^-$. Let $\mathcal{G}_i(B) = \nabla\ell_i(B)$, with advantage masks and loss-normalization denominators fixed across passes. We distinguish the policy-gradient channels $G^+ = \sum_i \mathcal{G}_i(A^+)$ and $G^- = \sum_i \mathcal{G}_i(A^-)$ from the regularized optimizer inputs $\widetilde{G}^+, \widetilde{G}^-$. In regularized training, the optimizer applies Algorithm 1 to these latter inputs.

Assumption B.1 (Signed linearity in the advantage)**.** *For every micro-batch i, on the signed advantage components,*

$$\mathcal{G}_i(c_+A^+ + c_-A^-) = c_+\mathcal{G}_i(A^+) + c_-\mathcal{G}_i(A^-), \qquad c_+, c_- \in \{-1, 0, 1\}.$$

The unclipped policy gradient satisfies Assumption B.1. A clipped PPO/DAPO surrogate can violate it because changing an advantage's sign can change the active clipping branch. One optimization epoch alone does not ensure an unclipped gradient. At a policy ratio of one, with the old log probability held constant during differentiation, the clipping is locally inactive and the assumption holds. The implementation constructs this ratio when there is one minibatch, one optimization epoch, and old log probabilities are taken from the detached current forward pass rather than rollout log probabilities. These conditions must be checked for the particular run. The signed channels use the token-level definition in Section 3. Response-mean grouping is equivalent only for response-constant advantages, so PPO with GAE retains the token-level split.

B.2 Relations Among the Three Implementations

Proposition B.2 (Reference implementation)**.** *The three-pass implementation forms $\widetilde{G}^+ = G^+ + \lambda\nabla\mathcal{R}$ and $\widetilde{G}^- = G^-$ with separate masked backward passes. If $\mathcal{G}_i(A) = \mathcal{G}_i(A^+) + \mathcal{G}_i(A^-)$ for every i, then $\widetilde{G}^+ + \widetilde{G}^- = \nabla\mathcal{L}$. This condition is weaker than Assumption B.1 and holds for tokenwise clipped surrogates with the same normalization denominator in every pass. The shared regularizer is assigned to the positive channel.*

Proof. Sum the identity under masking over micro-batches and add $\lambda\nabla\mathcal{R}$. No advantage sign is reversed in these masked passes. □

Proposition B.3 (Two-pass implementation agrees with the reference)**.** *Let $G = \nabla\mathcal{L}$ and $G^{|A|} = \sum_i \nabla\ell_i(|A|) + \lambda\nabla\mathcal{R}$. Under Assumption B.1,*

$$\tfrac{1}{2}\big(G + G^{|A|}\big) = \widetilde{G}^+, \qquad \tfrac{1}{2}\big(G - G^{|A|}\big) = \widetilde{G}^-, \tag{10}$$

with $\widetilde{G}^+, \widetilde{G}^-$ as in Proposition B.2, including the regularizer attribution.

Proof. Substituting $|A| = A^+ - A^-$ into Assumption B.1 gives $\nabla\ell_i(|A|) = \nabla\ell_i(A^+) - \nabla\ell_i(A^-)$; adding and subtracting $\nabla\ell_i(A) = \nabla\ell_i(A^+) + \nabla\ell_i(A^-)$ and summing over i yields the two identities. □

The training script includes an optional diagnostic that compares the two-pass positive channel with an additional positive-masked backward pass on at most two micro-batches per step during the first three steps. This sampled check can detect a violated identity on the inspected micro-batches. It does not establish the clipping regime or numerical agreement throughout a full run.

Table 5: **Three implementations of the regularized optimizer channels.** Backward passes are counted per micro-batch.

Implementation	Passes	Requirement	Used in
Reference (three-pass)	3	additivity under masking	GRPO, cross-model
Two-pass ($\|A\|$ identity)	2	Assumption B.1	DAPO
Single-backward routing	1	single sign per increment, including distributed contributions	Section 6.3

Proposition B.4 (Routing: policy-gradient agreement and total conservation)**.** *Suppose each routed gradient increment $\Delta_i = \nabla \ell_i + \lambda \nabla \mathcal{R}_i$ contains contributions of only one advantage sign $s_i \in \{+, -\}$, and $\mathcal{R} = \sum_i \mathcal{R}_i$. In distributed training, route before any reduction that mixes signs, or require all ranks contributing to a reduced increment to have the same sign. Then routing Δ_i to channel s_i preserves the total gradient and agrees with the reference policy-gradient channels, up to floating-point summation error. Shared regularizer gradients follow the sign of their increment.*

Proof. Snapshotting the gradient accumulator around a backward pass isolates its accumulated increment. Under the stated condition, every policy-gradient contribution in that increment has the routed sign and enters the same channel as in Proposition B.2. Summing the two channels recovers $\sum_i \Delta_i = \nabla \mathcal{L}$. A reduction that mixes signs before routing can preserve this sum while invalidating the individual channel assignments. □

The available routing unit test checks total-gradient conservation in a single-process float64 calculation. It does not validate channel attribution after distributed gradient reduction. Even under the proposition's conditions, routing can differ from the reference in regularizer attribution, and the nonlinear map in Eq. (5) can therefore produce a different direction. Routing adds no persistent optimizer state.

B.3 Choice of Implementation

The reference construction requires additivity under masking with fixed loss normalization. The two-pass construction additionally requires Assumption B.1, under which it reproduces both regularized reference channels. Routing requires a single sign per routed increment, including all distributed contributions. Under that condition it reproduces the policy-gradient split and conserves the total gradient, while regularizer attribution can differ. Table 5 summarizes these requirements. Appendix C reports cost, and Appendix D records the implementation assigned to each experiment.

C Computational Complexity

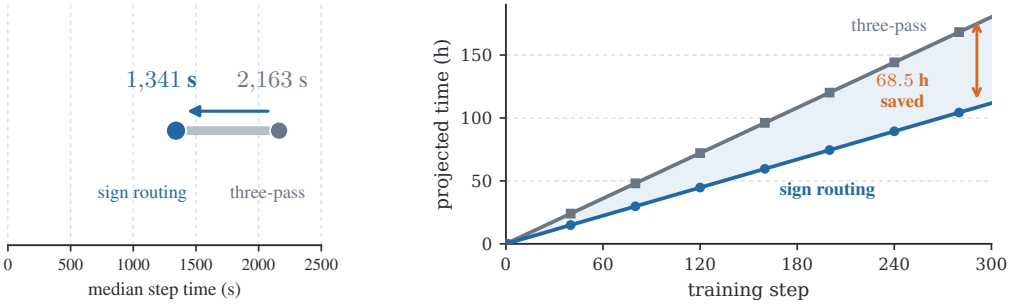
Sign routing removes the extra backward passes of the reference split. Relative to one sign-agnostic spectral transformation, the remaining optimizer-side overhead is a second Newton–Schulz pass, two randomized range projections, and two channel accumulators. For an $m \times n$ gradient with $\bar{r}_t = \max(r_t^+, r_t^-)$, the signed split and range projections cost $O(mn\bar{r}_t)$ per step, followed by two Newton–Schulz passes at the cost of Muon's whitening each. In the measured RLVR runs, these optimizer-side terms remain secondary to rollout generation.

C.1 Sign Routing and Practical Cost

Sign routing removes the extra backward passes used by the reference split. The headline accuracy results use three backward passes to obtain the full, positive, and negative gradients.

In our GRPO implementation, each local micro-batch has one advantage sign. Direct accumulation into the corresponding channel recovers the reference policy-gradient split only when the routed increment also satisfies the distributed condition in Proposition B.4. Local sign homogeneity and total-gradient conservation alone do not establish channel equivalence. Shared regularizer attribution can differ as described in Appendix B. In the designated single-run implementation A/B, routing

reduces median step time from 2163 to 1341 seconds, a 38% reduction and a 1.61× throughput increase (Fig. 6a). Projecting these rates through step 300 gives 180.3 versus 111.8 hours, a difference of 68.5 hours (Fig. 6b). Table 7 instead pools all logged steps across the available seed-42–44 runs.



(a) Per-step cost.

(b) Projected cumulative time.

Figure 6: **Timing of one-pass sign routing.** The A/B values are medians from one designated run per implementation; Table 7 reports the separate all-logged-step aggregation across available seeds. **(a)** Routing reduces median step time from 2163 to 1341 seconds, a 38% reduction and 1.61× throughput increase. **(b)** Projecting these single-run rates through step 300 gives 180.3 hours for the three-pass split and 111.8 hours for routing, saving 68.5 hours.

Sign routing also completes the full training horizon. Table 6 reports two 300-step runs. Their four-benchmark means are 0.4818 and 0.4823. The corresponding three-pass reference runs reach 0.4861 and 0.4956, giving same-seed gaps of 0.0043 and 0.0133. Averaged over these two seeds, sign routing trails the reference by 0.0088. Thus, routing retains competitive endpoint accuracy while reducing split cost, but it does not reproduce the reference result exactly.

The endpoint comparison does not isolate the cause of this gap. Channel attribution requires the checks described in Appendix B.

Table 6: **Full-run accuracy of one-pass sign routing at step 300.** AIME24/25 use avg@32. Mean gives equal weight to the four displayed benchmark values. Both rows use the Qwen3-8B/GRPO setup.

Training seed	MATH500	AIME24	AIME25	OlympiadBench	Mean
42	0.8000	0.2880	0.2620	0.5770	0.4818
43	0.7980	0.2940	0.2550	0.5820	0.4823
Mean	0.7990	0.2910	0.2585	0.5795	0.4820

Overall optimizer cost. Table 7 reports the two PASP implementations separately. With routing, the actor-update time of PASP matches that of Muon (468 versus 471 seconds), and the remaining overhead is mainly memory (about 35 versus 20 GB per GPU). In our implementation, this memory comes from two gradient-sized buffers that hold G_t^+ and G_t^- during the actor update and from Adagrad-style state that is allocated for every parameter, although only the parameters on the Adagrad fallback path use it; the spectral update itself keeps no persistent state. The peak remains below 40% of the 96 GB device memory. Routing lowers the median step time from 2213 seconds for the three-pass reference to 1367 seconds, compared with 1137 seconds for Muon and 1083 seconds for Pion. Step time also depends on generation length, and PASP responses are about 8% shorter than Muon's on average; the actor-update row is therefore the closer measure of optimizer cost.

D Experimental Details

This appendix separates the protocol shared across optimizers from the optimizer recipes, split implementations, and cross-setting changes.

D.1 Common Training Protocol

The primary Qwen3-8B and Qwen3-1.7B runs start from scale-matched math-supervised checkpoints and train on DAPO-Math-17k (Yu et al., 2025) for 300 steps. All runs use verl (Sheng

Table 7: **Practical cost on the shared 8-GPU Qwen3-8B/GRPO node.** Times are medians over all logged steps of the available seed-42–44 runs; response length is the mean over the same steps, and peak per-GPU memory is taken from `max_memory_allocated`. Actor-update time covers the backward passes and the optimizer step. The optimizers share data and training budget but use their disclosed recipes.

	AdamW	Muon (hidden-only)	Pion	PASP (three-pass)	PASP (routing)
Median step time	1883 s	1137 s	1083 s	2213 s	1367 s
Median actor-update time	589 s	471 s	453 s	1306 s	468 s
Mean response length	6660	5652	5353	5274	5214
Peak memory per GPU	≈28 GB	≈20 GB	≈20 GB	≈34 GB	≈35 GB

et al., 2025) (upstream commit `018f8dc5`), PyTorch 2.7.0a0 (NGC 25.04), and one node with $8\times$ NVIDIA H20 GPUs. Each step samples 256 prompts with 4 responses per prompt and a micro-batch of 1 response/GPU. Prompts are capped at 1024 tokens and generation at 8192 tokens. The shared loss settings are an in-loss KL coefficient of 0.001 with the low-variance estimator, entropy coefficient 0, and a constant learning-rate schedule. Weight decay and the distributed implementation are optimizer-specific and are reported below.

Within each setting, SGD, AdamW, hidden-only Muon, Pion, and PASP share the data, horizon, hardware allocation, and evaluation protocol. Optimizer-specific learning rates, regularization, and distributed implementations are disclosed in the corresponding recipes. Long runs are checkpointed and resumed on the shared cluster. Figure 5 is stitched across resume segments by step index, with every point read from the on-disk training logs.

We use *diverges* for a run that encounters non-finite optimization values before step 300. We use *collapses* for a numerically finite run that loses usable task-solving behavior, either producing a near-zero evaluated endpoint or sustained degenerate generations for which no valid endpoint is reported. A missing value therefore indicates that no endpoint remained suitable for benchmark comparison, not that the run was assigned zero accuracy.

D.2 Optimizer Recipes

The main comparison fixes one prespecified, method-specific recipe per optimizer. PASP, hidden-only Muon, and whole-matrix Pion use 5×10^{-6}; AdamW uses 1×10^{-6}; and momentum SGD uses 1×10^{-3}. PASP uses residual-rank ratio $\alpha = 1$, rank bounds $r_{\min} = 8$ and $r_{\max} = 64$, adaptive RMS-clipping parameters $(\kappa, \beta) = (3, 0.95)$, weight decay 0.01, and FSDP2 (PyTorch Foundation, 2024). Hidden-only Muon retains its native RMS-matched MuonAdamW recipe with weight decay 0.1, update scale 0.2, and FSDP2 (Liu et al., 2025). Whole-matrix Pion uses weight decay 0.01, FSDP2, momentum 0.95, five Newton–Schulz iterations, and scale 2.0 (Fan et al., 2026). AdamW uses weight decay 0.01 and FSDP1; SGD uses weight decay 0.01, momentum 0.9, and FSDP1. The per-head Pion implementations also use FSDP1. These recipes retain the shared data, initialization, global batch size, training horizon, hardware allocation, and evaluation protocol stated above. Because their update rules induce different scales, a common absolute actor learning rate would not provide a matched local comparison. Appendix E.5 first applies the same $\{0.5\times, 1\times, 2\times\}$ relative grid to every central recipe. When a baseline's best result lies at the grid boundary, the check extends once in that direction: to $4\times$ for Muon and $0.25\times$ for Pion. The Pion grid includes 1×10^{-5}, the learning rate used in the official May 2026 release.[2] The resulting grid measures local sensitivity around the reported recipes rather than claiming globally tuned optima.

The per-head Pion comparison uses both ends of the published promotion/suppression family: $(k_p, k_s) = (0, 5)$ and $(2, 3)$, with head dimension 128 and the same rate, protocol, and scale-matched SFT start. At 8B, each variant has three independent runs; all three $k_p = 0$ runs collapse, while the $k_p = 2$ entry aggregates its three runs. At 1.7B, each variant has one run and both collapse (Table 1).

For a parameter tensor j excluded from the spectral path, the signed Adagrad fallback maintains

$$H_{t,j} = H_{t-1,j} + (G^{+}_{t,j})^{\odot 2} + (G^{-}_{t,j})^{\odot 2}, \qquad U_{t,j} = \frac{G^{+}_{t,j} + G^{-}_{t,j}}{\sqrt{H_{t,j}} + \epsilon_{\mathrm{A}}}, \tag{11}$$

[2] Pion official repository, commit `3d228f5`.

with $H_{0,j} = 0$ and $\epsilon_{\mathrm{A}} = 10^{-8}$, where the operations are elementwise. The same joint clip c_t is applied to $U_{t,j}$ before its base-learning-rate update. This fallback retains the signed decomposition in the diagonal accumulator without applying a spectral map.

D.3 Split Implementations and Experimental Assignment

The primary GRPO and cross-model PASP accuracy runs use the reference implementation (Proposition B.2). The DAPO experiment uses the two-pass implementation (Proposition B.3), whose equivalence requires Assumption B.1. The implementation includes the sampled diagnostic described in Appendix B. The wall-clock A/B and routing accuracy runs in Appendix C use single-backward routing. Policy-gradient channel equivalence for those runs requires the distributed condition in Proposition B.4, in addition to locally single-signed micro-batches. Table 13 identifies the implementation used for each primary 8B run.

D.4 Evaluation and Cross-Setting Changes

MATH500 (Lightman et al., 2023; Hendrycks et al., 2021) uses $n = 500$ sampled responses; AIME24 and AIME25 use avg@32 ($n = 960 = 30 \times 32$ each); and OlympiadBench (He et al., 2024) uses $n = 581$ sampled responses. All benchmarks use temperature 0.6, top-p 1.0, and an 8192-token generation budget. The extended benchmarks in Appendix E.4 use the same settings: AMC23-Chain4 uses avg@32 over 40 four-problem chains ($n = 1280 = 40 \times 32$), while Minerva-Math ($n = 272$), the full MATH test set ($n = 5000$), and GSM8K ($n = 1319$) use one sampled response per question.

Table 1 averages each primary benchmark across independent training seeds, while Table 14 provides the complementary run-level view by averaging the four primary benchmarks within each seed. We use the approximately ± 0.005 fluctuation observed under repeated tensor-parallel decoding only as a descriptive estimate of decoding variability; it is not a confidence interval and is not used for significance testing.

The Llama-3.1-8B and DeepSeekMath-7B experiments use the same math-SFT recipe, 300-step GRPO data and budget, and evaluation protocol. For Qwen3-1.7B, Llama-3.1-8B, DeepSeekMath-7B, and Qwen3-8B/DAPO, the PASP, Muon, and Pion comparisons use seeds 42–44; Table 14 reports their per-seed four-benchmark means. DeepSeekMath-7B is evaluated at tensor-parallel 2 with an extended-context override for its native 4096-token position limit. The 1.7B rows in Table 9 apply the same dual-channel probe to the corresponding step-300 checkpoints (ambient dimension 2048) and average over the same seven projection types.

The DAPO experiments retain the same stack and SFT start, use one on-policy optimization epoch per collected batch, and apply clip ratios $0.2/0.28$, dynamic group filtering, and token-mean loss aggregation. GPQA-Diamond (Table 10) uses the same evaluation harness with one sample per question, temperature 0.6, top-p 1.0, seed 0, and the same 8192-token budget.

E Additional Results

Figure 5 reports the shared-seed training dynamics in Section 6.

E.1 Residual-Rank Summary

Among the tested settings, $\alpha = 1$ is the smallest value that completes training and remains close to $\alpha = 2$ and 5 in final accuracy. The sweep supports limited sensitivity within this intermediate range; it does not establish robustness to arbitrary residual ranks. Table 8 reports the benchmark-level results and the independent repeat at the default ratio.

E.2 Signed Rank Asymmetry

The probes cover the seven attention and MLP projection types and exclude embedding and output matrices. Each prompt has four sampled responses. At each probe, the positive and negative policy-gradient channels are computed from the same samples, before any optimizer transformation.

Table 8: **Residual-rank sweep on Qwen3-8B.** Only α varies. Mean accuracy gives equal weight to the four primary benchmarks. The default and repeat use seeds 43 and 44. Dashes indicate no valid endpoint for the collapsed run and unreported benchmark-level scores for the repeat.

α	MATH500	AIME24	AIME25	OlympiadBench	Mean	Outcome
0.5	—	—	—	—	—	collapse (step 90)
1	0.816	0.300	0.260	0.606	0.4956	completes
1 (repeat)	—	—	—	—	0.4852	completes
2	0.824	0.317	0.255	0.585	0.4953	completes
5	0.812	0.312	0.260	0.582	0.4917	completes
10	0.766	0.217	0.193	0.528	0.4259	degraded

Table 9: **Effective-rank asymmetry across estimators, model families, and scales.** Values summarize the per-layer mean and observed range.

Model	Estimator / optimizer	Mean ρ	Layer range
Qwen3-8B	GRPO	35.7	7–73
	Reinforce++	55.1	11–112
	RLOO	88.9	17–183
	PPO	87.1	17–173
	DAPO	32.9	7–67
Llama-3.1-8B	PASP	74.0	14–128
Llama-3.1-8B	Muon	51.2	11–86
Qwen3-1.7B	PASP	48.2	13–106
Qwen3-1.7B	Muon	45.5	12–100

For Figure 2a, let $\mathcal{P} = \{q, k, v, o, \text{gate}, \text{up}, \text{down}\}$ and let L_p be the number of layers with projection type p. We compute effective rank separately for each gradient matrix and summarize each channel as

$$\bar{r}_t^{\pm} = \frac{1}{7} \sum_{p \in \mathcal{P}} \frac{1}{L_p} \sum_{\ell=1}^{L_p} r_{\text{eff}}(G_{t,\ell,p}^{\pm}), \qquad \rho_t = \frac{\bar{r}_t^{-}}{\bar{r}_t^{+}}.$$

Thus, Figure 2a reports a ratio of mean effective ranks, not a mean of matrix-level ratios. It shows steps $30, 60, 90, 120, 150, 180, 210, 240, 270, 300$. At step 30, the channel means are approximately 229 and 10.8, giving $\rho = 21.2$.

Figure 2b shows the layer-mean effective ranks of each projection type at steps 30, 150, and 300. Large points average these three values, and colored spans extend from their minimum to their maximum. The negative-channel value exceeds the positive-channel value in all 21 comparisons.

Table 9 gives the estimator-, model-, and scale-level measurements summarized in Fig. 2c, including both the cross-layer mean and the observed layer range.

E.3 Out-of-Domain Results

Table 10 reports GPQA-Diamond as a secondary evaluation of the model families in Table 1. The Qwen3 entries aggregate the corresponding evaluated training runs, whereas Llama-3.1-8B and DeepSeekMath-7B report one run from their primary three-seed pools. These secondary evaluations did not enter model, hyperparameter, or checkpoint selection. PASP leads on Qwen3-8B and Llama-3.1-8B and is within one question of Pion on Qwen3-1.7B and DeepSeekMath-7B. Every spectral optimizer lifts GPQA-Diamond above its SFT start in all four settings; AdamW remains near SGD on Qwen3-1.7B.

E.4 Extended Benchmark Coverage

AMC23-Chain4. Each chain consists of four AMC23 problems. From the second problem onward, one quantity in the statement is given by the previous answer. The reported protocol scores only the final answer, with an 8192-token budget for the full response. This score does not separately verify

Table 10: **GPQA-Diamond accuracy with one sample per question** ($n = 198$)**.** Qwen3 entries aggregate the corresponding evaluated independent training runs; Llama-3.1-8B and DeepSeekMath-7B report one secondary-evaluation run from their primary three-seed pools. Muon denotes the hidden-only variant and Pion the whole-matrix variant.

Model	Optimizer	GPQA-Diamond
Qwen3-8B	SGD	0.251
	AdamW	0.261
	Muon (hidden-only)	0.397
	Pion	0.449
	Pion per-head	0.414
	PASP	0.480
	SFT start	0.227
Qwen3-1.7B	SGD	0.119
	AdamW	0.129
	Muon (hidden-only)	0.157
	Pion	0.251
	PASP	0.253
	SFT start	0.086
Llama-3.1-8B	SGD	0.152
	AdamW	0.157
	Muon (hidden-only)	0.192
	Pion	0.172
	PASP	0.202
	SFT start	0.136
DeepSeekMath-7B	SGD	0.086
	AdamW	0.101
	Muon (hidden-only)	0.116
	Pion	0.202
	PASP	0.197
	SFT start	0.101

the correctness of every intermediate answer. The benchmark evaluates dependent problem solving, and its scores are not comparable with standard AMC23 results.

The four primary benchmarks of Table 1 were fixed before training, and no additional benchmark entered model, hyperparameter, or checkpoint selection. Table 11 evaluates step-300 checkpoints on AMC23-Chain4 (40 chains, avg@32), Minerva-Math (Lewkowycz et al., 2022), the full MATH test set (Hendrycks et al., 2021), and GSM8K (Cobbe et al., 2021), under the protocol of Appendix D. The Qwen3 entries aggregate the corresponding evaluated runs, whereas Llama-3.1-8B and DeepSeekMath-7B report one secondary-evaluation run from their primary three-seed pools. This selection overlaps with the SimpleRL-Zoo evaluation suite (Zeng et al., 2025), but uses the full MATH test set rather than MATH500. On Qwen3-8B, PASP leads on AMC23-Chain4 and Minerva-Math and matches or slightly exceeds every baseline on the near-saturated MATH and GSM8K. On Qwen3-1.7B, the stability pattern of Table 1 recurs: AdamW remains between SGD and hidden-only Muon, whole-matrix Pion falls below Muon on Minerva-Math, MATH, and GSM8K, and PASP leads on all four columns. In the single secondary-evaluation runs for Llama-3.1-8B and DeepSeekMath-7B, PASP leads on MATH and GSM8K. AMC23-Chain4 accuracy does not exceed 0.010 for any optimizer on these two models, so we do not compare them on this benchmark; on Minerva-Math the optimizers differ by at most a few questions and remain within single-run noise. The near-saturated columns indicate that the gain on competition-level benchmarks is not paid for by regression on easier problems.

E.5 Local Learning-Rate Check

After closing the open learning-rate boundaries, PASP retains the highest best-tested score. Table 12 first tests $0.5\times$, $1\times$, and $2\times$ each optimizer's central actor learning rate at seed 42 while holding the other training settings fixed. Because Muon and Pion attain their best initial-grid scores at $2\times$ and $0.5\times$, respectively, we extend Muon to $4\times$ and Pion to $0.25\times$. Muon collapses at the

Table 11: Extended benchmarks for step-300 checkpoints. Qwen3 entries aggregate the corresponding evaluated independent training runs; Llama-3.1-8B and DeepSeekMath-7B report one secondary-evaluation run from their primary three-seed pools. AMC23-Chain4 chains four AMC23 problems and scores only the final answer, using avg@32; the other columns use one sampled response per question. Muon denotes the hidden-only variant and Pion the whole-matrix variant. None of these benchmarks entered model, hyperparameter, or checkpoint selection.

Model	Optimizer	AMC23-Chain4	Minerva-Math	MATH	GSM8K
Qwen3-8B	SGD	0.008	0.270	0.755	0.920
	AdamW	0.010	0.285	0.773	0.934
	Muon (hidden-only)	0.045	0.309	0.807	0.942
	Pion	0.050	0.304	0.809	0.946
	PASP	0.086	0.328	0.817	0.946
Qwen3-1.7B	SGD	0.012	0.200	0.700	0.850
	AdamW	0.016	0.213	0.718	0.862
	Muon (hidden-only)	0.021	0.224	0.751	0.874
	Pion	0.018	0.157	0.504	0.799
	PASP	0.049	0.281	0.785	0.881
Llama-3.1-8B	SGD	0.001	0.147	0.479	0.870
	AdamW	0.000	0.151	0.489	0.862
	Muon (hidden-only)	0.002	0.202	0.523	0.882
	Pion	0.001	0.173	0.507	0.859
	PASP	0.010	0.180	0.535	0.900
DeepSeekMath-7B	SGD	0.002	0.103	0.391	0.750
	AdamW	0.004	0.103	0.392	0.755
	Muon (hidden-only)	0.001	0.110	0.431	0.772
	Pion	0.004	0.143	0.450	0.775
	PASP	0.006	0.140	0.456	0.800

added point, while Pion decreases to 0.4433; neither extension improves the corresponding baseline. PASP attains 0.4861 at $1\times$, above Pion's best-tested 0.4763 and Muon's best-tested 0.4716 by 0.0098 and 0.0145, respectively. The added points therefore bracket the best-tested PASP, Muon, and Pion rates without changing their ordering. Because this check varies only the seed-42 actor learning rate, it measures local sensitivity rather than fully tuned, multi-seed optimizer optima.

Table 12: **Local actor-learning-rate checks around method-specific central recipes.** The experiment uses Qwen3-8B/GRPO, seed 42, and a 300-step training budget, and changes only the actor learning rate. The initial grid covers $0.5\times$, $1\times$, and $2\times$; a baseline is extended by one point when its best result lies at the grid boundary. Each entry is the equal-weight mean of MATH500, AIME24, AIME25, and OlympiadBench. Bold marks the best tested point for each optimizer, and dashes mark untested settings. For Pion, $2\times$ corresponds to the 1×10^{-5} learning rate used in its official May 2026 release. This check measures local sensitivity; it is not a global hyperparameter search.

Optimizer	$0.25\times$	$0.5\times$	$1\times$	$2\times$	$4\times$
PASP	–	0.4659	0.4861	0.1625	–
Muon (hidden-only)	–	0.4400	0.4555	0.4716	collapse
Pion (whole matrix)	0.4433	0.4763	0.4724	0.0000 (collapse)	–
AdamW	–	0.3980	0.4213	0.4130	–
SGD	–	0.3850	0.4021	0.3920	–

F LIMITATIONS

Scope. PASP targets RLVR regimes with a concentrated positive channel. We have not tested whether the same design benefits pretraining, where Muon's uniform spectral exploration may remain useful.

Cost. The live range projections and second Newton–Schulz pass add runtime. Peak memory rises from approximately 20 to 34–35 GB per GPU because the current implementation keeps two signed-channel buffers and allocates Adagrad-style state for every parameter. Restricting that state

to the fallback parameters, which hold about 15% of the Qwen3-8B weights, would remove most of the second source; we did not apply this optimization. Single-backward routing removes the extra backward passes but not these optimizer-side costs.

Anchor rank and refresh. Anchor construction leaves two design choices insufficiently tested. PASP currently estimates r_t^+ from the stable rank of the positive gradient, then uses a randomized range finder to construct an anchor basis of that dimension. Limited preliminary runs suggest that retaining a modest rank margin, $r_t^+ + \Delta r$, may improve performance, but the current evidence does not establish a reliable effect or an appropriate margin. PASP also recomputes the anchor at every optimizer step. Preliminary observations suggest that the dominant positive subspace may change slowly over short windows, which motivates periodic anchor reuse, including refreshing it every 20 optimizer steps. We have not systematically tested how the refresh interval affects final performance or training stability. Future work should vary the anchor-rank margin and refresh interval jointly and compare final accuracy, training stability, subspace drift, and compute cost across multiple seeds.

Experimental coverage. The primary comparison uses one prespecified recipe per optimizer rather than equal-budget hyperparameter tuning. No optimizer is retuned for Qwen3-1.7B; a retuned Pion may recover at that scale, so the Qwen3-1.7B comparison supports robustness to recipe transfer rather than a ranking of tuned optima. The seed-42 local check in Appendix E.5 uses a common initial three-point grid and one-sided boundary extensions for Muon and Pion. It measures local sensitivity around the reported recipes but does not identify fully tuned optimizer optima. For Qwen3-1.7B, Llama-3.1-8B, DeepSeekMath-7B, and Qwen3-8B/DAPO, PASP, Muon, and Pion each include seeds 42–44, and the component ablation reports means over independent training seeds. The matched-anchor and full-run sign-routing studies use seeds 42 and 43. At 8B, both per-head Pion variants use three seeds, and all three $k_p = 0$ runs collapse; only the 1.7B per-head variants remain single-run comparisons. The residual-rank study, learning-rate endpoints, and some secondary cross-model evaluations also retain single-run cells. The matched-anchor study holds the rank rule fixed to the positive channel; it does not tune a source-specific rank for each alternative anchor. Broader multi-seed validation across learning-rate endpoints, anchor rules, optimizer recipes, and non-mathematical tasks remains necessary.

Scope of the evidence. A two-seed matched-anchor comparison further supports this interpretation: the positive anchor ranks first for both seeds when rank, compute, and all other training choices are fixed. In the local learning-rate check in Appendix E.5, the added boundary points bracket the best-tested PASP, Muon, and Pion rates without changing their ordering; PASP retains the highest best-tested score. Because this check varies only the seed-42 actor learning rate, it characterizes local sensitivity rather than fully tuned optimizer optima. Full-run sign routing reduces gradient-splitting cost but trails the three-pass reference by 0.0088 on average over two matched seeds. These results narrow the remaining claims to the tested models, tasks, learning-rate grid, and seed coverage.

Interpretation. Effective rank summarizes spectral concentration but does not identify policy-relevant directions. The measured asymmetry motivates the positive anchor without establishing a causal explanation for the performance gains. Response counts, token counts, and response lengths may affect the channel spectra, and the probes do not establish whether the asymmetry persists after matching these factors. PASP transforms current gradients, whereas Muon and Pion transform momentum, so cross-method comparisons do not isolate momentum handling. The no-sign-split ablation retains PASP's current-gradient path.

G Per-Run Results

Table 13 reports the independent 8B runs used to compute the primary results in Table 1; the matched-anchor controls are reported separately in Table 3. The three PASP runs span 0.4852–0.4956; seeds 43 and 44 also provide the default and replicate points in the rank-ratio study. Repeated decoding evaluations of the same checkpoint are not counted as independent training runs.

Table 1 averages independent training seeds within each benchmark. To expose the run-to-run variation hidden by that aggregation, Table 14 takes the complementary view and averages the same four benchmarks within each seed.

Table 13: **Mean accuracy across the four reasoning benchmarks for each independent 8B training run.** AIME24/25 use avg@32; the PASP runs use the three-pass reference split.

Optimizer	Implementation	Seed 42	Seed 43	Seed 44
SGD	—	0.4021	0.3907	0.3946
AdamW	—	0.4213	0.4093	0.4132
Muon (hidden-only)	—	0.4555	0.4513	0.4600
Pion (whole-matrix)	—	0.4724	0.4539	0.4828
Pion (per-head, k_p=0)	—	*collapses*	*collapses*	*collapses*
Pion (per-head, k_p=2)	—	0.4619	0.4681	0.4567
PASP	three-pass	0.4861	0.4956	0.4852

Table 14: **Per-seed mean accuracy behind the cross-setting blocks of Table 1.** Each seed column averages MATH500, AIME24, AIME25, and OlympiadBench for one independent training run; Mean averages seeds 42–44.

Setting	Optimizer	Seed 42	Seed 43	Seed 44	Mean
Qwen3-1.7B / GRPO	PASP	0.460	0.470	0.459	0.463
	Muon (hidden-only)	0.398	0.394	0.403	0.398
	Pion	0.211	0.192	0.151	0.184
Llama-3.1-8B / GRPO	PASP	0.211	0.221	0.210	0.214
	Muon (hidden-only)	0.204	0.200	0.208	0.204
	Pion	0.194	0.175	0.204	0.191
DeepSeekMath-7B / GRPO	PASP	0.173	0.183	0.172	0.176
	Muon (hidden-only)	0.161	0.157	0.165	0.161
	Pion	0.169	0.150	0.179	0.166
Qwen3-8B / DAPO	PASP	0.482	0.492	0.481	0.485
	Muon (hidden-only)	0.466	0.462	0.470	0.466
	Pion	0.471	0.452	0.481	0.468

G.1 Paired Differences and Uncertainty

Table 15 compares PASP with the strongest non-PASP optimizer of each setting on the same seeds. With three seeds per setting, the descriptive paired-t intervals are wide, and only the Qwen3-1.7B interval excludes zero. Across the five settings, PASP is higher in 13 of the 15 same-seed comparisons, ties once, and is lower once; excluding the Qwen3-1.7B transfer setting gives 10 wins, one tie, and one loss. Because seeds are nested within settings and the strongest baseline is selected separately for each setting, we do not treat these 15 signs as independent trials or attach a pooled sign-test p-value. The differences of about 0.010 on Llama-3.1-8B and DeepSeekMath-7B are consistent in direction but are not resolved within those settings.

Table 15: **Same-seed differences between PASP and the strongest baseline of each setting.** Each entry is a difference in four-benchmark mean accuracy. The descriptive confidence interval is a paired t interval with two degrees of freedom. Each run is evaluated once, so the differences include decoding variability.

Setting	Baseline	Seed 42	Seed 43	Seed 44	Mean	95% CI
Qwen3-8B / GRPO	Pion	+0.014	+0.042	+0.002	+0.019	[−0.031, +0.070]
Qwen3-1.7B / GRPO	Muon (hidden-only)	+0.062	+0.076	+0.056	+0.065	[+0.039, +0.090]
Llama-3.1-8B / GRPO	Muon (hidden-only)	+0.007	+0.021	+0.002	+0.010	[−0.014, +0.034]
DeepSeekMath-7B / GRPO	Pion	+0.004	+0.033	−0.007	+0.010	[−0.041, +0.061]
Qwen3-8B / DAPO	Pion	+0.011	+0.040	0.000	+0.017	[−0.034, +0.068]